Pontificia Universidad Javeriana
Departamento de Matemáticas
Análisis Numérico

Reto Análisis Numérico

Estefania Aristizabal, Oscar Falla, Victor Barón
aristizabalj-e@javeriana.edu.co falla_o@javeriana.edu.co
victorvc@javeriana.edu.co

23 de Febrero del 2020

1. Evaluación de un Polinomio

1. Implemente en R o Python el método de Horner para evaluar $f'(x_0)$, tenga en cuenta la precisión de la computadora o unidad de redondeo.

   Sea:
   $$P(x) = a_n x^n + a_{n-1} + ... + a_1 x + a_0 \tag{1}$$

   Si $b_n = a_n$ y:
   $$b_k = a_k + b_{k+1}x_0 \qquad \text{para } k = n-1, n-2, ..., 1, 0 \tag{2}$$

   entonces $b_0 = P(x_0)$ más aún, si:
   $$Q(x) = b_n x^{n-1} + b_{n-1}x^{n-2} + ... + b_2 x + b_1 \tag{3}$$

   Por tanto,
   $$P(x) = (x - x_0)Q(x) + b_0 \tag{4}$$

   al derivar respecto a x tenemos
   $$P'(x) = Q(x) + (x - x_0)Q'(x) \qquad y \qquad P'(x_0) = Q(x_0) \tag{5}$$

   Ejemplo con: $P(x) = 2x^4 - 3^2 + 3x - 4$ en $x_0 = -2$:

	de x^4	de x^3	de x^2	de x	constante
$x_0 = -2$	$a_4 = 2$	$a_3 = 0$	$a_2 = -3$	$a_1 = 3$	$a_0 = -4$
	$b_4x_0 = 0$	$b_3x0 = -3$	$b_2x_0 = -10$	$b_1x_0 = 14$	
	$b_4 = 2$	$b_3 = -4$	$b_2 = 5$	$b_1 = -7$	$b_0 = 10$

   En este paso se aplicó el método de Horner al polinomio dado con coeficientes $[2, 0, -3, 3, -4]$ esto es equivalente a aplicar la ecuación (2)

   el polinomio evaluado en x_0 es $P(x_0) = 10$

   De esta manera podemos expresar el polinomio $P(x)$ como:
   $$P(x) = (x+2)(2x^3-4x^2+5x-7)+10 \qquad \text{donde} \qquad Q(x) = 2x^3-4x^2+5x-7 \tag{6}$$

   Derivando tenemos:
   $$P'(x) = (1)(2x^3 - 4x^2 + 5x - 7) + (x + 2)Q'(x) \tag{7}$$

Evaluando en el punto $x_0 = -2$

$$P'(x_0) = (1)(2(-2)^3 - 4(-2)^2 + 5(-2) - 7) + (0)Q'(x) \qquad \text{por lo tanto} \tag{8}$$

$P'(x_0) = Q(x_0)$

El siguiente paso es básicamente repetir el algoritmo de Horner (2) para evaluar polinomios en un punto pero en este caso con $Q(x)$:

$$\begin{array}{r|rrrr}
x_0 = -2 & 2 & -4 & 5 & -7 \\
 & & -4 & 16 & -42 \\
\hline
 & 2 & -8 & 21 & -49
\end{array}$$

De esta forma tenemos:

$$P'(x_0) = Q(x_0) = -49 \tag{9}$$

Algoritmo: Punto 1 Reto .R

2. Implemente el método de Horner si se realiza con números complejos, tenga en cuenta la precisión

   Si $z = \beta + \gamma i$ es un cero complejo de multiplicidad m del polinomio P, entonces $z = \beta - \gamma i$ también es un cero complejo de multiplicidad m del polinomio P, y $(x^2 - 2\beta x + \beta^2 + \gamma^2)^m$ es un factor de P

   Si deseamos evaluar un polinomio $P(x)$ en un valor complejo del argumento $x = \beta + i\gamma$, donde los coeficientes $a_k = b_k + ic_k$ son complejos. Poniendo $d_k = Q_k + iR_k$ obtenemos:

$$Q_n = b_0, R_0 = c_n \tag{10}$$
$$Q_k = Q_{k-1}\beta - R_{6k-1}\gamma + b_k, k = 1, 2, ..., N, \tag{11}$$
$$R_k = R_{k-1}\beta + Q_{k-1}\gamma + c_k, k = 1, 2, ..., N, \tag{12}$$

   Entonces, la división compleja funciona de la siguiente manera:

$\beta + 1\gamma$	Coef. x^N	Coef. x^{N-1}	...	Coef. x	Término constante
	b_0, c_0	b_1, c_1	...	b_{N-1}, c_{N-1}	b_N, c_N
		$Q_0\beta - R_0\gamma$	...	$QN - 2\beta - R_{N-2}\gamma$	$Q_{N-1}\beta - R_{N-1}\gamma$
		$Q_0\gamma + R_0\beta$	...	$Q_{N-2}\gamma + R_{N-2}\beta$	$Q_{N-1}\gamma + R_{N-1}\beta$
	Q_0, R_0	Q_1, R_1	...	$Q_{N-1}, RN - 1$	Q_N, R_N

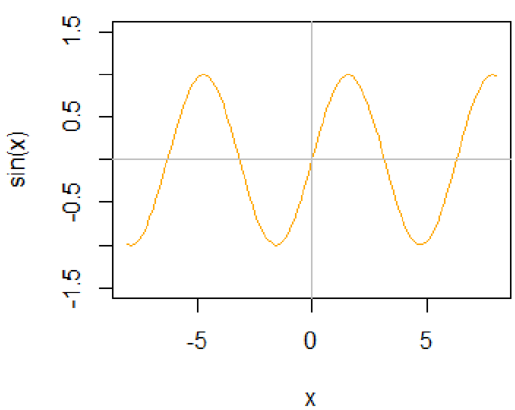
Usando el método de Horner, al evaluar el polinomio $P(x) = (1+i)x^3 + 2$ en $x_0 = (1-i)$, se obtienen los siguientes resultados:

$1-i$				
	$1, 1$	$0, 0$	$0, 0$	$2, 0$
		$2, 0$	$2, -2$	$0, -4$
	$1, 1$	$2, 0$	$2, -2$	$2, -4$

Básicamente, cuando se tratan de números complejos la máquina divide el número en dos partes en su parte real y en su parte compleja, de este modo el número de operaciones se multiplica a razón de dos sin mencionar la operación para dividir dicho número

hace cada una de las operaciones por aparte y después los une nuevamente para dar el resultado.

Algoritmo: Horner.h



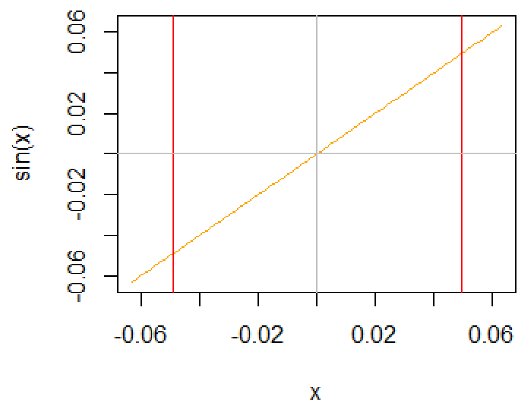
Gráfica del $sen(x)$

2. Óptima Aproximación Polinómica

1. Aplique una aproximación de Taylor

   Sustituir la función original por $T_n^{[}f(x)]$ en un punto de la función

   Cuanto más nos alejamos de este punto inicial de pegado, más error se cometerá en la estimación. Pero, el error se minimiza aumentando el grado del polinomio



Gráfica del $sen(x)$ en el intervalo $[-\pi/64, \pi/64]$

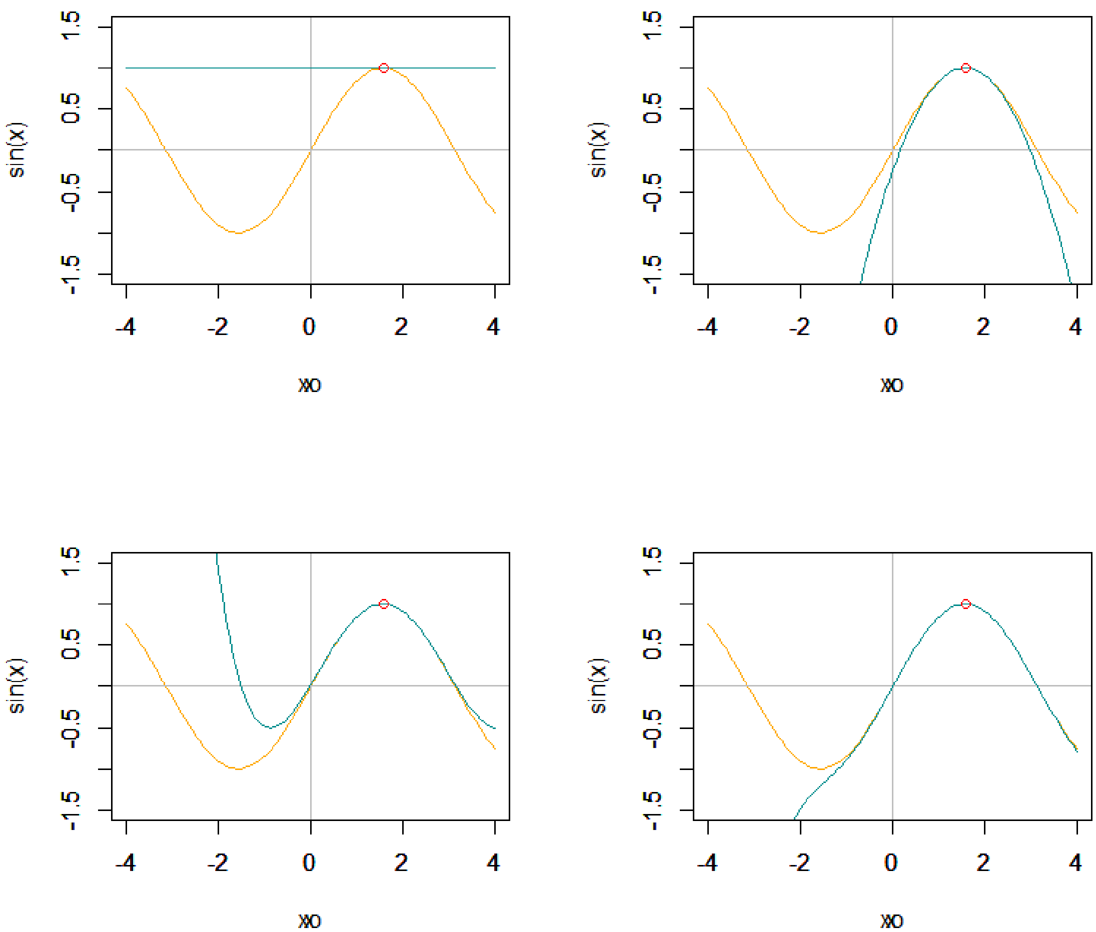
- Paso 1: Elegir el punto del polinomio de pegado original x_0
- Paso 2: Aplicar la expresión del polinomio de Taylor

$$T[f(x), x_0] = \sum_{i=0}^{\infty} \frac{f^{(i)}(x_0)}{i!}(x - x_0)^i \tag{13}$$

$$P_n(x) = \sum_{i=0}^{n} \frac{f^{(i)}(x_0)}{i!}(x - x_0)^i \tag{14}$$

$$P_n(x) = f(x_0) + f^{(1)}(x - x_0) + \frac{f^{(2)}(x_0)}{2!}(x - x_0)^{(2)} + ... + \frac{f^{(n)}(x_0)}{n!}(x - x_0)^{(n)} \tag{15}$$

Entre más términos tenga el polinomio, más se ajustará a la función original



Aproximaciones de Taylor en el punto $\pi/2$ con grado = (1,2,4,6)

Una de las grandes ventajas de evaluar el seno, es que sus derivadas se repiten después de cierto punto, esto hace el código más sencillo de implementar.

Error:

el error en el punto $\pi/4$ fue igual a 0.0003224255

el error en el punto 0 fue igual a 0.01996896

Esto demuestra que entre más lejos nos encontremos del punto de pegado en este caso $\pi/2$ mayor va a ser el error de aproximación.

Algoritmo Taylor.R

2. Implemente el método de Remez

Algoritmo de Remes: Es un algoritmo iterativo usado para encontrar simples aproximaciones

$$b_0 + b_1 x_i + ... + b_n x_i^n + (-1)^i E = f(x_i)(\text{donde} i = 0, 1, ...n + 1), \tag{16}$$

para las incógnitas $b_0, b_1, ...b_n$ y E.

- Paso 1: según el grado del polinomio elegir los puntos dentro del intervalo dado en este caso $[-\pi/64, \pi/64]$

  para esto tenemos dos alternativas

  - división regular:

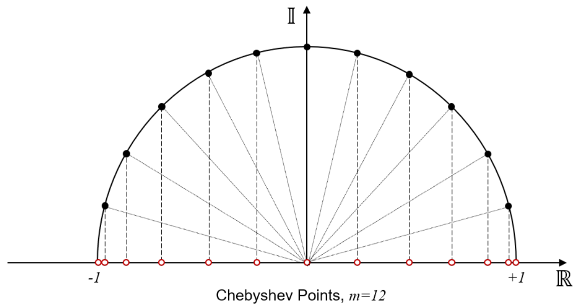
    La división regular consta en dividir el intervalo en $n + 2$ puntos o nodos, cuyas divisiones tienen la misma distancia.

  - division especial por nodos de Chebyshev:

    Son las raíces de los polinomios de Chebyshev del primer tipo. Se usan como nodos en la interpolación polinómica porque el polinomio de interpolación resultante minimiza el efecto del fenómeno de Runge.

    Los puntos de un intervalo de [a,b] estan dadas por:

$$x_k = \frac{1}{2}(a + b) + \frac{1}{2}(b - a)cos(\frac{2k - 1}{2n}\pi), \qquad k = 1, 2, ..., n \tag{17}$$



Particiones por Chebyshev en el intervalo $[-1, 1]$

Paso 2: resolver el sistema de ecuaciones dada por la expresión (18). Vamos a encontrar un polinomio de grado 1 que se aproxime a $sin(x)$ en el intervalo $[-\pi/64, \pi/64]$.

$$\begin{bmatrix} b_0 & b_1 & E \\ 1 & -\pi/64 & 1 \\ 1 & 0 & -1 \\ 1 & -\pi/64 & 1 \end{bmatrix} \begin{bmatrix} f(x) \\ 0 \\ 0{,}9995985 \\ 0 \end{bmatrix}$$

Obtenemos como resultado:

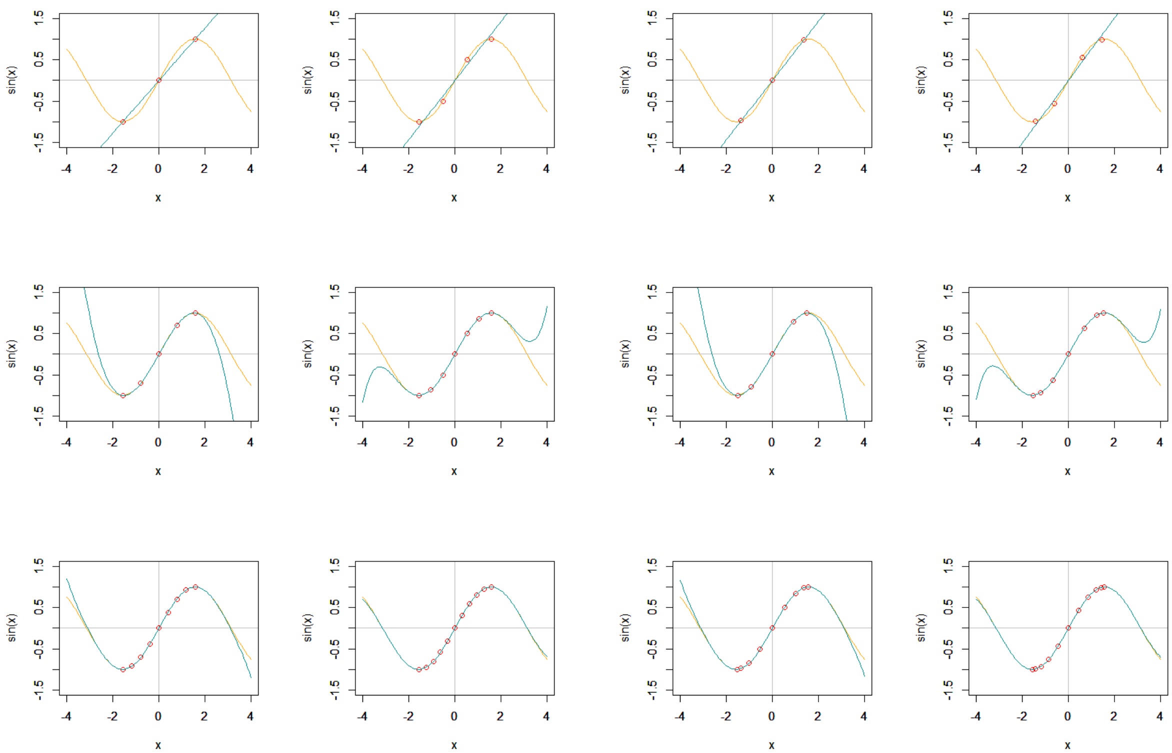
$$b_0 = 0 \qquad b_1 = 0{,}9995985 \qquad E = 0$$

$$P(x) = 0{,}9995985x \qquad E = 0 \tag{18}$$

Nota:

Como la función $sin(x)$ es relativamente sencilla vamos a hacer el análisis en un intervalo más amplio en este caso de $[-\pi/2, \pi/2]$.

Division Regular   División Especial Chebyshev



Comparación entre los tipos de particiones grado (1,2,3,4,5,7,9)

El error se calculó con la sumatoria de los errores absolutos en los puntos medios de cada partición, podemos observar que las particiones especiales de Chebyshev nos retornan un valor más preciso y minimizan el error ya que reducen la oscilación en los bordes que es el mismo fenómeno de Runge.

Algoritmo: Remez.R

Conclusiones:

Grado	Division Regular	División Especial Chebyshev
1	0.4142136	0.2799201
2	0.2320508	0.1522195
3	0.02116119	0.0176458
4	0.01782185	0.0101784
5	0.000460794	0.000398267
7	6.071399e-06	4.628377e-06
9	5.583387e-08	3.284476e-08

Error total

- se puede encontrar $P^n(x_0)$ si se sigue profundizando el algorito de Horner (2) por ejemplo:

$$P''(x_0) = Q'(x_0) = H(x_0) \tag{19}$$

  Donde $H(x) = 2x^2 - 8x + 21$ tomado de la tabla2

- Con los resultados del método de Horner se puede aproximar a uno de los ceros de la función aplicando el método de Newton:

$$x_1 = x_0 - \frac{P(x_0)}{P'(x_0)} \tag{20}$$

- si el punto de pegado en la aproximación de Taylor es 0, entonces es el caso especial llamado Aproximación de Maclaurin

  dado por la expresión:

$$P_n(x) = \sum i = 0^n \frac{f^i(0)!}{i!} \tag{21}$$

- El error de truncamiento esta dado por la omisión de cifras significativas en las opereciones matemáticas esto se debe a la limitación de la máquina para almacenar dichas cifras.