Reto 1 - Análisis Numérico

Andrés Giraldo Gil
a_giraldo@javeriana.edu.co

Erika Alejandra González
gonzalez_erika@javeriana.edu.co

Leonel Steven Londoño
leonel-londono@javeiana.edu.co

24 de febrero de 2020

1. Evaluación de un Polinomio

1.1. Implemente en 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.

Marco Teórico: El algoritmo de Horner es un algoritmo para evaluar de forma eficiente funciones polínomicas de grado n de una forma monomial. [1] El método de Horner únicamente requiere n multiplicaciones y n sumas para evaluar un polinomio arbitrario de grado n.[2]

Análisis: En este algoritmo inicialmente se le enviaran los coeficientes a la función de Horner, la cual derivará dicha función y devolverá un vector con los nuevos coeficientes. Estos coeficientes seran luego enviados a la funcion Horner la cual aplicará dicho algoritmo con los nuevos coeficientes.

Datos utilizados:
Entradas:

- coef = (2,0,3,3,4), coeficientes que acompañan a x
- $x_i = 3$ valor con el cual se evaluará x

Salidas

- 36, valor de $f'(x)$ en Horner
- Número de multiplicaciones: 3
- Número de sumas: 3

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

Marco Teórico: Los números complejos son una extensión de los números reales. Los números complejos incluyen todas las raíces de los polinomios a diferencia de los reales. Estos son representados con la suma del número real y un número imaginario (que va representado con la letra i). Suelen estar representados de la forma $a + bi$[3]

Análisis: A la función Horner se le envía el número complejo, el cual en este caso debe ser operado con su parte real y su parte imaginaria, para esto en cada iteración se debe realizar su respectiva multiplicación y suma, teniendo en cuenta ambos elementos, esto ocasiona que el número de operaciones sea mayor a que si solo se le aplicara Horner a un número real. Para el manejo de números complejos se utilizó la función as.complex(numero_complejo). Esto permite que R pueda realizar las operaciones necesarias para un número complejo, teniendo en cuenta la parte imaginaria.

Datos utilizados
Entradas

- $x_i = \sqrt{-28}$ valor con el cual se evaluará x
- coef = (2,0,-3,3,4) coeficientes que acompañan x

Salidas

- $1648 + 15.875i$ valor de $f'(x)$ en Horner
- Número de multiplicaciones: 16
- Número de sumas: 12

2. Óptima aproximación polinómica

2.1. Aplique una aproximación de Taylor.

Marco Teórico: En matemáticas, la serie de Taylor es una aproximación de funciones mediante una suma de potencias enteras de polinomios, esta suma se calcula a partir de las derivadas de la función para un determinado valor o punto a. También se le denomina Serie de Maclaurin. [4]

Análisis: Para este punto se utilizó el método de Taylor de la librería Pracma [5] para obtener el polinomio de Taylor con la función trigonométrica $sen(x)$, esta función calcula los primeros 4 coeficientes de la serie de Taylor.

```
seno <- function(x){
    return(sin(x))
}
polinomioDeTaylor = pracma::taylor(seno,0,4)
```

Para calcular la aproximación de Taylor se crea la función *aproxTaylor* que recibe como parametros x (el punto donde se evaluará), el polinomio de Taylor y el grado del polinomio. La función es las siguiente:

```
aproxTaylor <- function(x, polinomio, gradoDelPolinomio){
    seno = 0
    for (i in polinomio) {
        seno = seno + i * (x^gradoDelPolinomio)
        gradoDelPolinomio = gradoDelPolinomio - 1
```

```
        }
    return (seno)
}
```

Para tener una mejor precisión, evaluamos la aproximación de Taylor en el intervalo $[-\pi/64, \pi/64]$ con particiones distintas, las cuales se fueron cambiando dentro de un ciclo con 10 iteraciones donde se cambia la partición, después de obtener los diferentes resultados y compararlos con los datos reales de la evaluación del punto en $sen(x)$ se obtiene el error relativo por cada partición y así mismo la cantidad de operaciones necesarias, estas se pueden evidenciar en la tabla de los datos obtenidos.

Datos obtenidos

El número de respuetas siempre será el número de iteraciones + 1

Iteraciones	No. Operaciones	Error Relativo
1	8	$4.83515e^{-06}$
2	12	$3.223433e^{-06}$
3	16	$2.447443e^{-06}$
4	20	$2.054725e^{-06}$
5	24	$1.823016e^{-06}$
6	28	$1.671123e^{-06}$
7	32	$1.564501e^{-06}$
8	36	$1.485487e^{-06}$
9	40	$1.424859e^{-06}$
10	44	$1.376723e^{-06}$

Tabla 1: No. de Operaciones y Error Relativo con respecto a las iteraciones

Se puede concluir que a mayor número de iteraciones, se necesita arealizar más opereaciones y el error relativo de cálculo es menor, como se puede evidenciar en la siguiente gráfica:

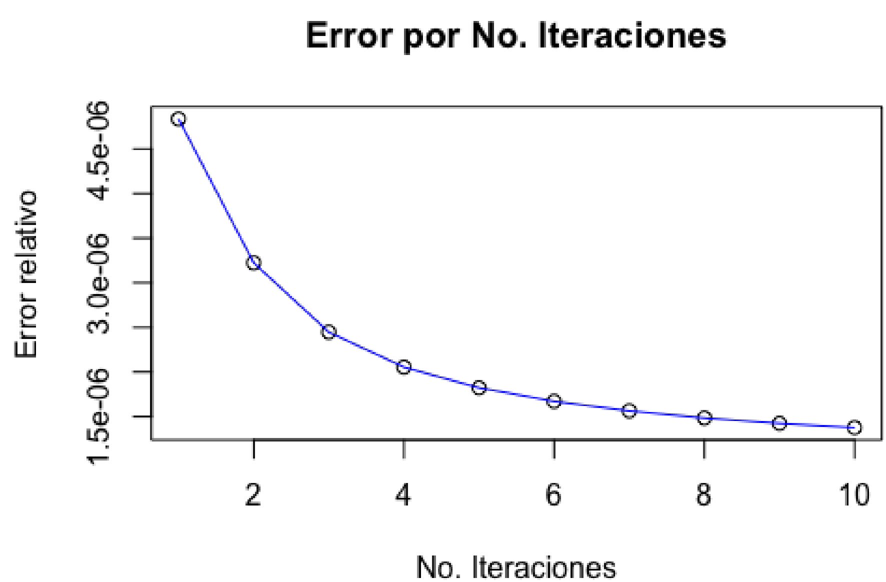

Figura 1.Error vs Iteraciones

2.2. Implemente el método de Remez.

Marco Teórico: El método usado para construir las mejores aproximaciones uniformes de un polinomio se conoce como Remez. La sucesión formada por los polinomios Pn obtenidos por Remez siempre es convergente a la mejor aproximación y ademas esta converge de manera cuadrática, si al aproximar es diferenciable. El algoritmo Remez es una metodología para localizar la aproximación racional minimax a una función. La serie de pasos que se utilizan en este algoritmo los siguientes: [6]

- Seleccionar n+2 puntos x0< x1<···< xn< xn+1 en I= [a, b] arbitrariamente.
- Calcular el polinomiop nEPn y el parametro d tales que

$$f(xi) - pn(xi) = (1)^{(i+1)}d \tag{1}$$

  para i= 0,1,···, n, n+ 1. Si se escoge una base de Pn, las n+ 1 componentes del polinomio en esa base y valor de d son las n+ 2 incognitas de un sistema lineal.
- Determinar los puntos en los que la función |fpn|alcanza su maximo y reemplazar con ellos, todos o parte de los xi utilizados. [6]

Analisis

Datos Utilizados

Funcion a Aproximar:

$$f(x) = sin(x) \tag{2}$$

Intervalo en el cual se hallara la aproximación de la funcion:

[-pi/64,pi/64]

Datos Obtenidos El metodo de Remez implementado se evaluo en tres situaciones diferentes:

Polinomio Aproximado en el intervalo, Grado 3

En esta prueba se evaluo la aproximación usando un grado 3 en el polinomio, como resultado se obtuvo el polinomio:

$$f(x) = 0 + 0.9999999 * x + 0 * x^2 - 0.1666115 * x^3 \tag{3}$$

El polinomio (linea de color rojo) anterior graficado respecto a la funcion sen(x)(linea de color azul) se muestra a continuación:

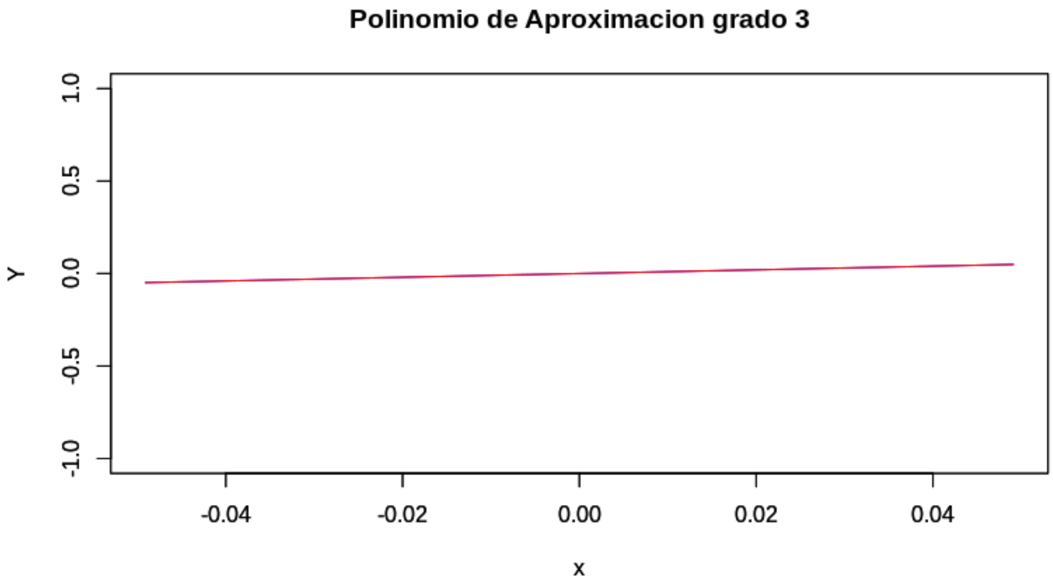

Figura 2.Aproximación del polinomio grado 3 en el intervalo [-pi/64,pi/64]

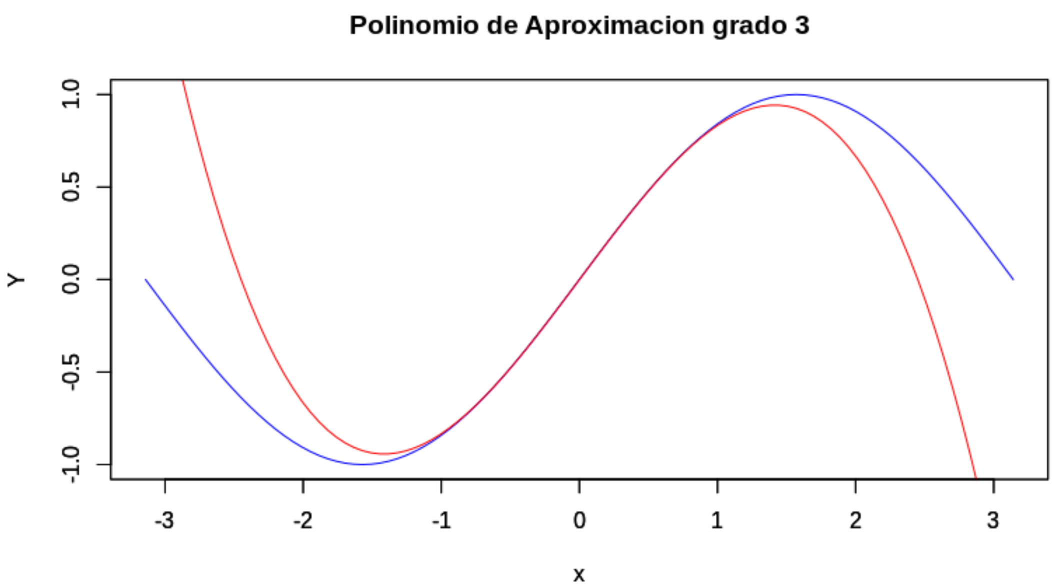

Figura 3.Aproximación del polinomio grado 3 en el intervalo [-pi,pi]

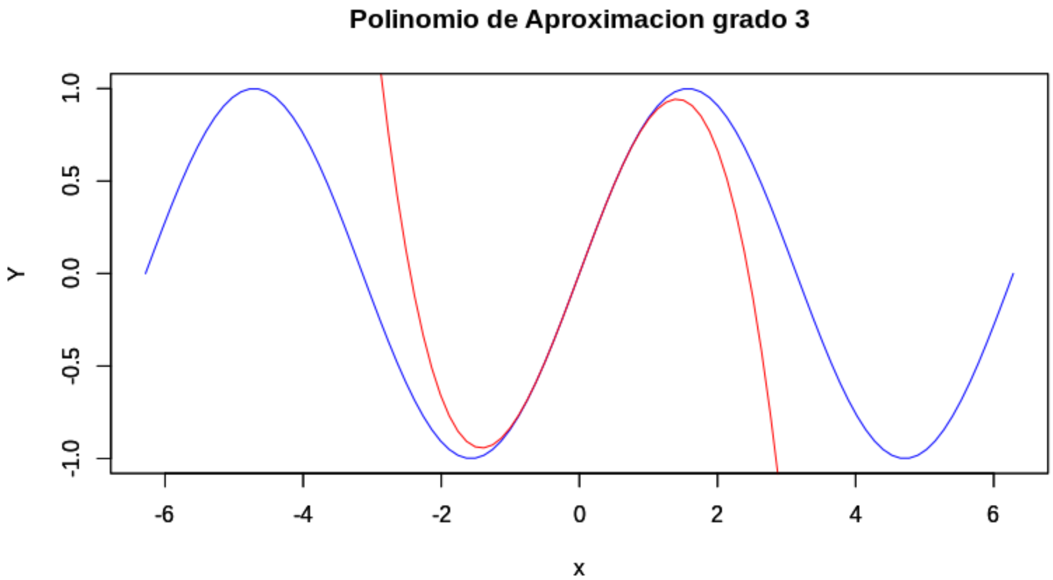

Figura 4.Aproximación del polinomio grado 3 en el intervalo [-2pi,2pi]

La aproximación de un polinomio de grado 3,como se pudo observar en las figuras anteriores, varia dependiendo del intervalo que se tome. La figura 2, la exactitud del polinomio a simple vista es muy parecida, pero a medida que se aumenta el intervalo de grafica, el polinomio se empieza a diferenciar como en la figura 3, el polinomio finalmente tiene la forma del funcion seno hasta cierto punto y despues crece infinitamente en el eje y como se muestra en la figura 4.

Dado el punto en el cual se evaluara el polinomio y la función seno: pi/200

El Error Relativo en ese punto respecto al valor en la función seno en el mismo punto es de : 7.155069e-18

El Error Absoluto en ese punto respecto al valor en la función seno en el mismo punto es de : 1.123869e-15

Polinomio Aproximado en el intervalo, Grado 4

En esta prueba se evaluo la aproximación usando un grado 4 en el polinomio, como resultado se obtuvo el polinomio:

$$f(x) = 1.769123e-10+1.000000e+00*x-7.342065e-07*x^2-1.666444e-01*x^3+2.742338e-04*x^4 \quad (4)$$

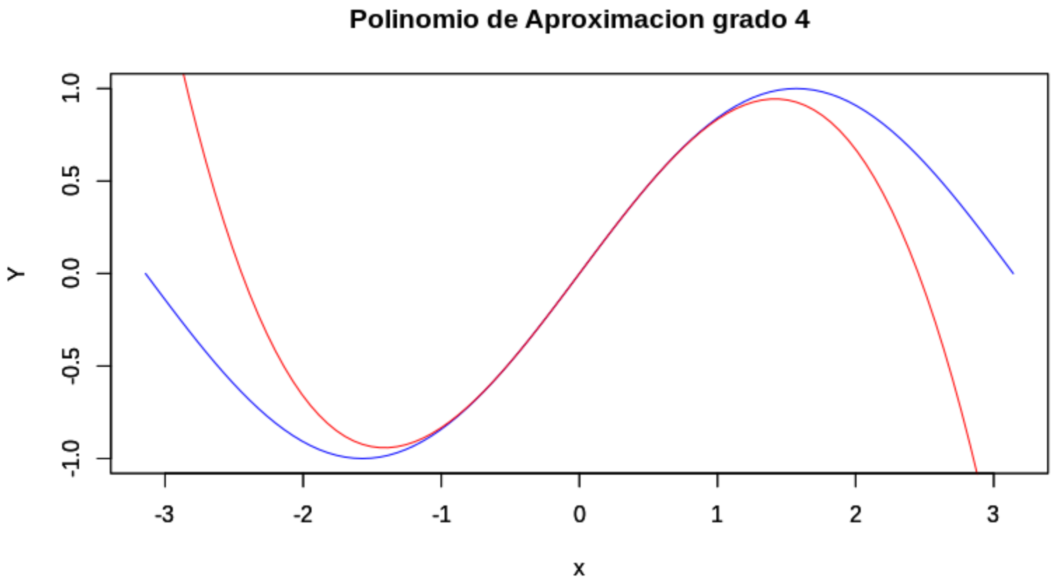

Figura 5.Aproximación del polinomio grado 4 en el intervalo [-pi,pi]

Como se pudo observar en la figura 4 la aproximación de un polinomio de grado 4 en el intervalo [-pi, pi] es muy parecida a la del grado 3, la unica diferencia radica en que el porcentaje de los errores es mas reducido, esto quiere decir que la aproximación es mas acertada:

Dado el punto en el cual se evaluara el polinomio y la función seno: pi/200

El Error Relativo en ese punto respecto al valor en la función seno en el mismo punto es de : 4.142958e-20

El Error Absoluto en ese punto respecto al valor en la función seno en el mismo punto es de : 6.507475e-18

Polinomio Aproximado en el intervalo, Grado 5 en adelante

Para los polinomios con un grado igual a 5 o mayor, la aproximación es muy similar a la de un polinomio de grado 4, los errores asociados a dichos polinomios empiezan a hacer constantes, tiendiendo al error del grado 4.

Analsis final

Al observar los datos obtenidos se determina que para el ejercicio en especifico usado con el metodo de Remez, se dice que la mejor aproximación es la del polinomio de grado 4, ya que el porcentaje de error respecto a la del 3 es mejor, para realizar una aproximación mas exacta se necesita un polinomio de un grado muy grande. Pero como el intervalo es pequeño, el grado 3 y 4 se ajustan bastante a la aproximación y el porcentaje de error evaluado en un punto dado es muy bajo en ambos casos.

Referencias

[1] A new method of solving numerical equations of all orders, by continuous approximation. Horner, W.G. 1819.

[2] Burden, R. Numerical Analisys. CENCAGE Learning. 2010

[3] Conway, John B. Functions of One Complex Variable I. Springe. 1986.

[4] Weisstein, Eric W. Series de Taylor. MathWorld

[5] Libreria Pracma para R. Disponible en: https://www.rdocumentation.org/packages/pracma/versions/1.9.9/topics/taylor

[6] Moreno G. Carlos. Introduccion al calculo numerico. Universidad Nacional de Educacion a Distancia. Madrid. 2014. Disponible en: http://www.cartagena99.com/recursos/alumnos/apuntes/8436262573calc.pdf.pdf