Pontificia Universidad JAVERIANA Bogotá

Facultad de Ciencias
Departamento de Matemáticas
eherrera@javeriana.edu.co

Análisis Numérico
Primer parcial
Agosto de 2021

Recomendaciones

- No se resuelven preguntas del contenido a evaluar.
- Cada estudiante debe subir a su repositorio un documento en pdf con la solución, incluidas gráficas y tablas (recuerde que debe aparecer la fecha y hora en los pantallazos o ImpPt) y debe estar los archivos: R y/o .py (código).
- Tiene plazo para subir su solución en la carpeta parciales/parcial1 hasta 9:10 am (hora local)-agosto 26 de 2021

**Inmediatamente que suba su solución, debe enviar un correo desde su cuenta de la javeriana a: eherrera@javeriana.edu.co con el link del repositorio (esto es obligatorio)**

1. En cada uno de los siguientes ejercicios debe implementar en R y/o Python el algoritmo asignado para resolver el problema (incluya en el código comentarios), generar una tabla con los errores relativos, determinar el número de iteraciones realizadas; una gráfica que evidencie el tipo de convergencia del método utilizado y si esta no es cuadrática aplique un método para acelerar la convergencia, para resolver el problema: Demuestre numéricamente que $f(x) = x^3 + 2x + k$ cruza el eje $x$ exactamente una vez, independientemente del valor de la constante $k$. Realice varias pruebas tomando como valor de k: último digito de su documento de identificación+$\sqrt{k+2}$; último digito de su documento de identificación-1/3.

a. Método de Müller; $TOL < 10^{-11}$
b. Método de la secante; $TOL < 10^{-11}$
c. Método de la posición falsa; $TOL < 10^{-12}$
d. Método del punto fijo; $TOL < 10^{-12}$
e. Método de Newton-Raphson; $TOL < 10^{-12}$

2. En los siguientes ejercicios aplicar Taylor para aproximar $f(x)$, con un polinomio $P_i$ alrededor de $x_0$; calcular el error hacia adelante y hacia atrás para cada $x^*$; encuentre un límite superior para el error $|f(x^*) - P_i(x^*)|$ y compárela con el error real; realizar una gráfica que muestre el polinomio de aproximación y la función. Implemente en R y/o Python, con una precisión de $10^{-8}$.

a. $f(x) = e^x cosx; x_0 = 0; P_3(0.5)$
b. $f(x) = (x-1)lnx; x_0 = 1; P_3(0.5)$
c. $f(x) = xe^{x^2}; x_0 = 0; P_3(0.4)$
d. $f(x) = 2cos2x - (x-2)^2; x_0 = 0; P_3(0.4)$

3. Para cada una de las siguientes ecuaciones, determine un intervalo [a, b] en el que la iteración de punto fijo converge. Estime el número de iteraciones necesarias para obtener aproximaciones precisas dentro de $10^{-5}$ y realice los cálculos. Implemente en R y/o Python

a. $2 + sinx - x = 0$
b. $x - cosx = 0$
c. $x^3 - 2x - 5$
d. $3x + e^x - x^2 - 2$.

4. Sea $f(x) = e^x - x - 1$ (a) Muestre numéricamente que f tiene un cero de multiplicidad 2. (b) Utilizando el método de Newton con $x_0 = 1$demuestre numéricamente que converge a la raíz, pero no cuadráticamente. Implemente en R y/o Python
5. Sea $f(x) = e^x - x - 1$ realice una modificación del método de Newton con $x_0 = 1$para que converja a la raíz cuadráticamente. Implemente en R y/o Python