Manual de usuario

1. Introducción

Este documento proporciona las instrucciones de operación para el programa de Métodos Numéricos. El software está diseñado para resolver una variedad de problemas matemáticos comunes en el análisis numérico, de acuerdo con los métodos asignados.

El programa está implementado en Python y utiliza las bibliotecas `numpy` para operaciones matriciales y `sympy` para el manejo de funciones simbólicas.

2. Requisitos Previos

Para la ejecución del programa, el usuario debe tener un entorno de Python 3 instalado, junto con las siguientes bibliotecas:

- `numpy`
- `sympy`

Puede instalarlas usando el comando: `pip install numpy sympy`

3. Instrucciones Generales de Uso

3.1. Formato de Funciones (Sintaxis de SymPy)

Cuando el programa solicite una función (ej. `f(x)` o `f(t,y)`), esta debe ser ingresada como una **cadena de texto** utilizando la sintaxis de la biblioteca `sympy`.

Operación	Sintaxis Correcta	Sintaxis Incorrecta
Potencia	`x**2`, `x**3`	`x^2`, `x^3`
Multiplicación	`5*x`	`5x`
Exponencial (e^x)	`exp(x)`	`e^x`
Logaritmo Natural	`log(x)`	`ln(x)`
Seno, Coseno	`sin(x)`, `cos(x)`	

Pi (π)	`pi`	

3.2. Ingreso de Datos Numéricos

- **Valores individuales:** Ingrese el número. Use el punto (`.`) como separador decimal (ej. `3.14159`).
- **Vectores (Listas):** Para el vector `b` en Gauss-Seidel, ingrese los números separados por un espacio (ej. `6 25 -11`).
- **Matrices:** Para la matriz `A` en Gauss-Seidel, ingrese cada fila en una línea separada, con elementos separados por un espacio (ej. `10 -1 2`).

4. Guía de Funcionalidades (Menú)

El programa se controla mediante un menú numérico.

4.1. Opción 1: Método de Bisección

- **Propósito:** Encontrar una raíz de una ecuación no lineal f(x) = 0 en un intervalo [a,b]
- **Inputs Requeridos:**
  - `f(x)`: La función (ej. `x**3 + 4*x**2 - 10`).
  - `a`: Límite inferior del intervalo (ej. `1`).
  - `b`: Límite superior del intervalo (ej. `2`).
  - `Tolerancia` (Opcional): Criterio de paro (ej. `1e-6`).
  - `Max iteraciones` (Opcional): Límite de iteraciones (ej. `100`).
- **Salida:** Una tabla de iteraciones y la raíz aproximada c.

4.2. Opción 2: Método de Gauss-Seidel

- **Propósito:** Resolver un sistema de ecuaciones lineales $Ax = b$.
- **Inputs Requeridos:**
  - `Tamaño del sistema`: El número de ecuaciones, n.
  - `Matriz A`: n filas, ingresadas una por una (ej. `10 -1 2`).
  - `Vector b`: Una sola línea con n valores (ej. `6 25 -11`).
  - `Vector inicial?`: `s` o `n`. Si es `s`, se solicitará el vector `x0`. Si es `n`, se usará un vector de ceros.
  - `Tolerancia` (Opcional): Criterio de paro (ej. `1e-6`).
  - `Max iteraciones` (Opcional): Límite de iteraciones (ej. `100`).
- **Salida:** Una tabla de iteraciones con la solución x y el error, y la solución final.

4.3. Opción 3: Interpolación de Lagrange

- **Propósito:** Encontrar el polinomio P(x) que pasa por un conjunto de n puntos (x_i, y_i).
- **Inputs Requeridos:**
  - `Número de puntos`: n.
  - `x_i`: Se solicitarán n valores, uno por uno.
  - `y_i`: Se solicitarán n valores, uno por uno.
  - `Evaluar?`: `s` o `n`. Si es `s`, se solicitará un valor x para evaluar P(x).
- **Salida:** El polinomio simbólico P(x) expandido y, si se solicita, el valor de P(x) evaluado.

4.4. Opción 4: Regla de Simpson (1/3)

- **Propósito:** Calcular el valor de una integral definida $\int_a^b f(x)dx$
- **Inputs Requeridos:**
  - `f(x)`: La función a integrar (ej. `exp(x)`).
  - a: Límite inferior de integración.
  - b: Límite superior de integración.
  - `Subintervalos` (Opcional): El número n de divisiones (ej. `100`). (Nota: si n es impar, el programa lo ajustará a n+1$).
- **Salida:** El valor aproximado de la integral y una comparación con el valor exacto (si `sympy` puede calcularlo).

4.5. Opción 5: Derivada Numérica (2do Orden)

- **Propósito:** Calcular la primera derivada $f'(x_0)$ usando una fórmula de diferencia central de segundo orden $(O(h^2))$
- **Inputs Requeridos:**
  - `f(x)`: La función a derivar (ej. `sin(x)`).
  - `x0`: El punto donde se desea evaluar la derivada.
  - h (Opcional): El tamaño de paso (ej. `0.01`).
- **Salida:** El valor de $f'(x_0)$ aproximado y una comparación con la derivada analítica exacta.

4.6. Opción 6: Método de Euler

- **Propósito:** Resolver una Ecuación Diferencial Ordinaria (EDO) de primer orden dy/dt = f(t,y) con una condición inicial $y(t_0) = y_0$.
- **Inputs Requeridos:**
  - `f(t,y)`: La función de la EDO. **Debe usar las variables `t` y `y`** (ej. `t + y`).

- `t0`: Condición inicial para t.
- `y0`: Condición inicial para y.
- `tf`: Punto final de t para la solución.
- h: Tamaño de paso.

- **Salida:** Una tabla de los pasos $(t_i,\ y_i)$ y la solución final $y(tf)$

4.7. Opción 7: Sistema EDO (RK4)

- **Propósito:** Resolver un sistema de n EDOs de primer orden usando el método de Runge-Kutta de 4to orden.
- **Inputs Requeridos:**
  - `Número de ecuaciones`: `n`.
  - `dy{i}/dt`: Se solicitarán n funciones. **Debe usar `t` y las variables `y0`, `y1`, ... `y(n-1)`**.
    - (Ej. para un sistema de 2 EDOs: `dy0/dt = y1` y `dy1/dt = -y0`).
  - `t0`: Condición inicial para t.
  - `y{i}({t0})`: Se solicitarán n valores iniciales, uno por uno.
  - `tf`: Punto final de t para la solución.
  - h: Tamaño de paso.
- **Salida:** Una tabla de los pasos $(t_i, [y0, y1, \ldots])$ y la solución final en $t = tf$

4.8. Opción 0: Salir

- **Propósito:** Termina la ejecución del programa.

5. Manejo de Errores

El programa está diseñado para ser robusto y no fallará de forma catastrófica.

- **Error de Convergencia:** Si un método iterativo (Bisección, Gauss-Seidel) no alcanza la tolerancia en el número máximo de iteraciones, el programa lo notificará con el mensaje: `Maximo de iteraciones alcanzado`.
- **Entrada Inválida:** Si el usuario ingresa un dato de tipo incorrecto (ej. texto donde se espera un número, o una sintaxis de función inválida como `x^2`), el programa no se detendrá.
- **Error Matemático:** Si ocurre una operación indefinida (ej. división por cero al evaluar `1/x` en x=0, o `log(-5)`), el programa capturará la excepción.

En todos los casos de **Entrada Inválida** o **Error Matemático**, el programa mostrará un mensaje `ERROR: [descripción del error]` y permitirá al usuario presionar `[Enter para continuar]` para regresar al menú principal e intentarlo de nuevo.