Introduction to Computation (CS2201)

Lecture 8

Kripabandhu Ghosh

CDS, IISER Kolkata

DIFFERENTIAL EQUATIONS

Euler method

Premise

Given a differential equation $\frac{dy}{dx} = f(x, y)$, with initial condition $y(x_0) = y_0$

Euler method

A successive approximation of this equation can be given by:
$y_{n+1} = y_n + h * f(x_n, y_n)$ where $h = \frac{(x_n - x_0)}{n}$

Euler method (example)

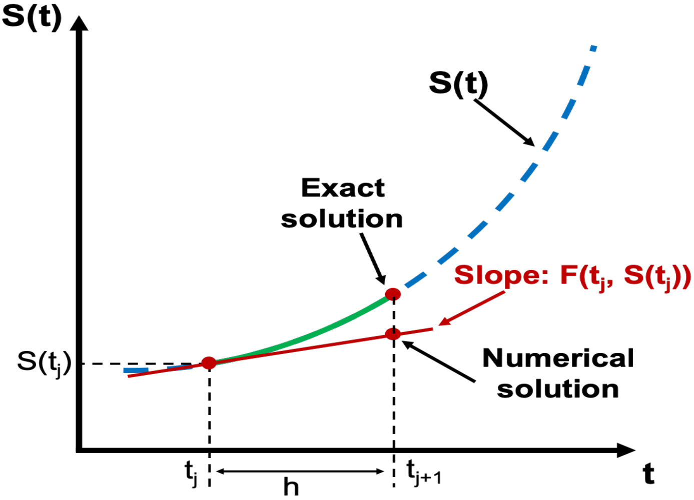
Equation

Given a differential equation $\frac{dy}{dx} = e^{-x}$, with initial condition $y_0 = 1$ has the exact solution $y =$ -e^{-x} and $f(x, y) = e^{-x}$

Some steps

n	y_n	x_n	$f(x_n, y_n)$	h	y_{n+1}	Actual (-e^{-x})
0	-1	0	1	0.25	-0.75	-1.0
1	-0.75	0.25	0.77880	0.25	-0.55530	-0.7788
2	-0.7788	0.5	0.60653	0.25	-0.40367	-0.60653
3	-0.40367	0.75	0.47237	0.25	-0.28557	-0.47237

Euler ODE[1]

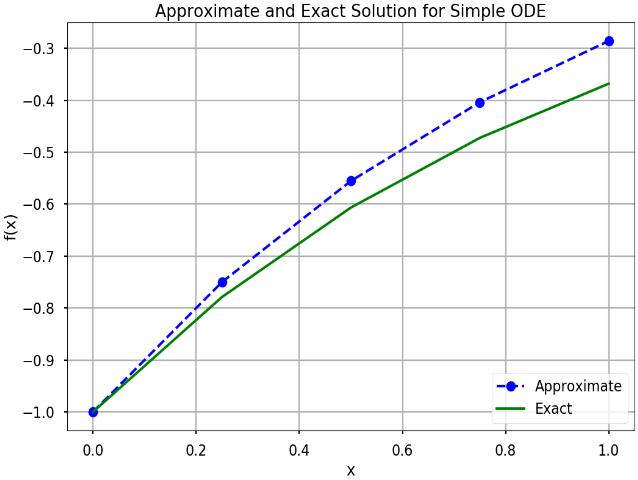


[1] http://tinyurl.com/eulerODE

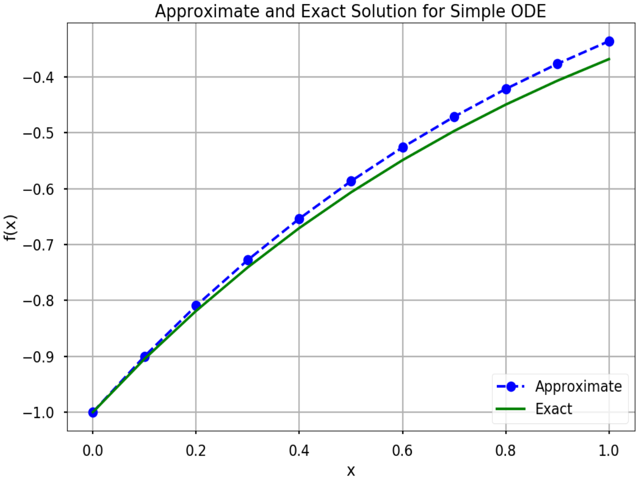
Euler ODE

```
import numpy as np
import matplotlib.pyplot as plt
plt.style.use('seaborn-poster')

#Define parameters
def f(t, s):
    return np.exp(-t)

h = 0.1 # Step size
t = np.arange(0, 1 + h, h) # Numerical grid
s0 = -1 # Initial Condition

# Explicit Euler Method
s = np.zeros(len(t))
s[0] = s0
for i in range(0, len(t) - 1):
    s[i + 1] = s[i] + h*f(t[i], s[i])

plt.figure(figsize = (12, 8))
plt.plot(t, s, 'bo–', label='Approximate')
plt.plot(t, -np.exp(-t), 'g', label='Exact')
plt.title('Approximate and Exact Solution
for Simple ODE')
plt.xlabel('x')
plt.ylabel('f(x)')
plt.grid()
plt.legend(loc='lower right')
plt.show()
```

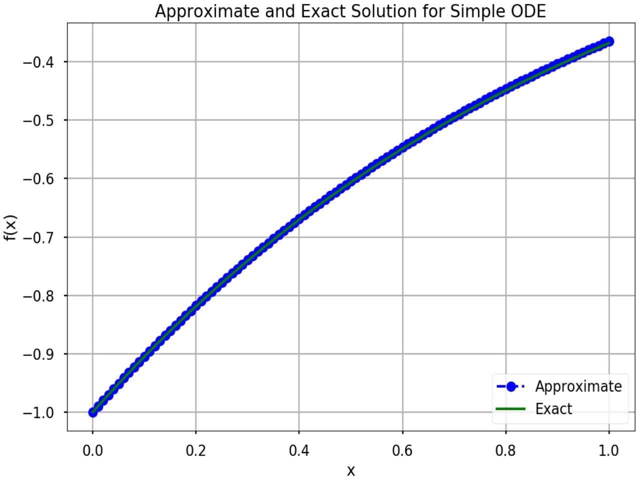
Euler ODE (h = 0.25)



Euler ODE (h = 0.1)



Euler ODE (h = 0.01)



THANK YOU !!!