LAGRANGE’S INTERPOLATION IMPLEMENTATION IN PYTHON

SUBMITTED BY: UMANG KANCHAN

SUBMITTED TO: Dr. AYESHA CHOUDHARY

ASSIGNMENT 5

NUMERICAL METHODS (CS-406)

2023

import numpy as np

# Function to calculate Lagrange basis polynomial

def lagrange\_basis(*x*, *i*, *x\_values*):

    basis = 1

    for j in range(len(x\_values)):

        if i != j:

            basis \*= (x - x\_values[j]) / (x\_values[i] - x\_values[j])

    return basis

# Function to perform Lagrange interpolation

def lagrange\_interpolation(*x*, *x\_values*, *y\_values*):

    n = len(x\_values)

    lagrange\_poly = 0

    for i in range(n):

        lagrange\_poly += y\_values[i] \* lagrange\_basis(x, i, x\_values)

    return lagrange\_poly

# Get input from user

n = *int*(input("Enter number of data points: "))

x\_values = np.zeros(n)

y\_values = np.zeros(n)

print("Enter data points:")

for i in range(n):

    x\_values[i] = *float*(input("x[" + *str*(i) + "] = "))

    y\_values[i] = *float*(input("y[" + *str*(i) + "] = "))

x = *float*(input("Enter interpolation point: "))

# Perform Lagrange interpolation

y = lagrange\_interpolation(x, x\_values, y\_values)

# Print solution

print("Interpolated value at x = " + *str*(x) + " is y = " + *str*(y))

SAMPLE OUTPUT

Enter number of data points: 3

Enter data points:

x[0] = 1

y[0] = 0

x[1] = 2

y[1] = 1

x[2] = 3

y[2] = 4

Enter interpolation point: 4

Interpolated value at x = 4.0 is y = 9.0