

Nama : Namira Nurfaliani

NIM : 21120122140135

Kelas : Metode Numerik/C

Implementasi Interpolasi Menggunakan Metode Polinom Lagrange

https://github.com/iranamira/metnum_Pertemuan10_Namira-Nurfaliani

Penerapan Metode Lagrange terhadap soal adalah menggunakan Bahasa pemrograman python. Alur untuk menemukan hasil dari implementasi interpolasi dengan menggunakan Polinom Lagrange :

1. Mempersiapkan input data dan titik interpolasi.
2. Menghitung nilai interpolasi menggunakan metode Lagrange.
3. Menampilkan hasil interpolasi.
4. Menghitung nilai interpolasi untuk rentang nilai x untuk plotting.
5. Membuat dan menampilkan grafik interpolasi dan titik data asli.

Source Code :

```
import numpy as np
import matplotlib.pyplot as plt

# Data yang diberikan
x = np.array([5, 10, 15, 20, 25, 30, 35, 40])
y = np.array([40, 30, 25, 40, 18, 20, 22, 15])

def lagrange_interpolation(x_values, y_values, x):
    """
    Menghitung nilai interpolasi Lagrange pada titik x.

    Args:
    x_values (numpy array): array dari nilai x yang diketahui
    y_values (numpy array): array dari nilai y yang diketahui
    x (float): nilai x yang ingin diinterpolasi

    Returns:
    float: nilai y hasil interpolasi
    """
    n = len(x_values)
    L = np.zeros(n)
    for i in range(n):
        L[i] = 1
        for j in range(n):
            if i != j:
                L[i] *= (x - x_values[j]) / (x_values[i] - x_values[j])
        return np.sum(L * y_values)

# Testing interpolasi pada beberapa titik
test_points = np.linspace(5, 40, 100)
interpolated_values = [lagrange_interpolation(x, y, point) for point in test_points]
```

```
# Plotting hasil interpolasi
plt.figure(figsize=(10, 6))
plt.plot(test_points, interpolated_values, label='Interpolasi Lagrange',
color='blue')
plt.scatter(x, y, color='red', label='Data asli')
plt.xlabel('Tegangan, x (kg/mm^2)')
plt.ylabel('Waktu patah, y (jam)')
plt.title('Interpolasi Polinom Lagrange')
plt.legend()
plt.grid(True)
plt.show()
```

Penjabaran kode program :

1. Import Library

```
import numpy as np
import matplotlib.pyplot as plt
```

- Numpy : digunakan untuk operasi numerik seperti array dan perhitungan matematis.
- matplotlib.pyplot : digunakan untuk membuat grafik.

2. Data yang diberikan

```
x = np.array([5, 10, 15, 20, 25, 30, 35, 40])
y = np.array([40, 30, 25, 40, 18, 20, 22, 15])
```

- x : adalah array dari nilai tegangan dalam satuan kg/mm²
- y : adalah array dari nilai waktu patah dalam satuan jam yang sesuai dengan nilai x.

3. Fungsi Interpolasi Lagrange

```
def lagrange_interpolation(x_values, y_values, x):
    n = len(x_values)
    L = np.zeros(n)
    for i in range(n):
        L[i] = 1
        for j in range(n):
            if i != j:
                L[i] *= (x - x_values[j]) / (x_values[i] -
x_values[j])
    return np.sum(L * y_values)
```

- def lagrange_interpolation : Fungsi ini menghitung nilai interpolasi Lagrange pada titik x
- x_values dan y_values : adalah array nilai x dan y yang diketahui
- n adalah jumlah titik data
- L : adalah array yang menyimpan nilai basis Lagrange
- for : digunakan untuk menghitung nilai basis Lagrange
- Nilai interpolasi dihitung sebagai jumlah dari produk basis Lagrange dengan nilai y yang sesuai.

4. Menghitung, Testing, Memplot Hasil Interpolasi

```
test_points = np.linspace(5, 40, 100)
interpolated_values = [lagrange_interpolation(x, y, point) for
point in test_points]
```

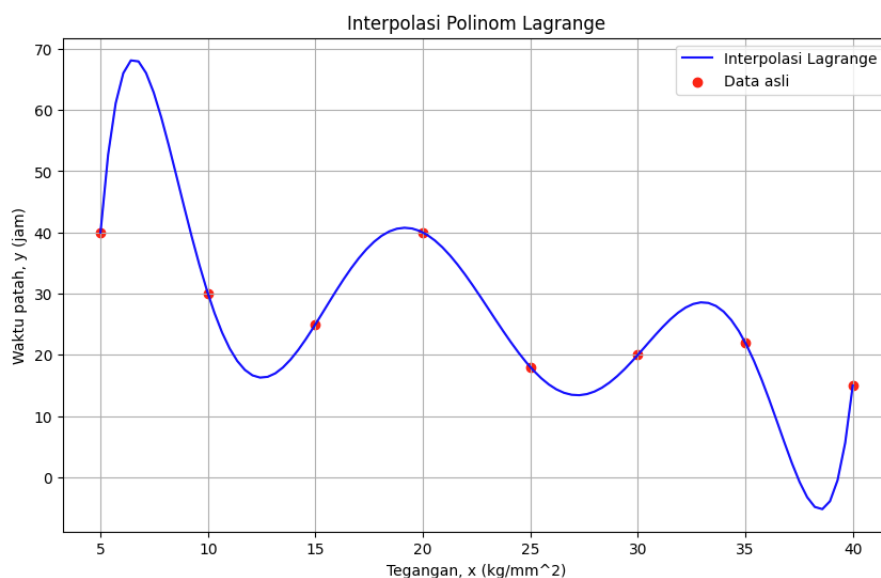
- `test_points` : adalah array 100 titik yang dipilih secara linier dari 5 hingga 40 untuk mengevaluasi interpolasi.
- `interpolated_values` : menyimpan hasil interpolasi untuk setiap titik dalam `test_points`.

5. Plotting

```
plt.figure(figsize=(10, 6))
plt.plot(test_points, interpolated_values, label='Interpolasi
Lagrange', color='blue')
plt.scatter(x, y, color='red', label='Data asli')
plt.xlabel('Tegangan, x (kg/mm^2)')
plt.ylabel('Waktu patah, y (jam)')
plt.title('Interpolasi Polinom Lagrange')
plt.legend()
plt.grid(True)
plt.show()
```

- Membuat figure baru dengan ukuran 10x6
- `plt.plot` : Memplot garis hasil interpolasi Lagrange
- `plt.scatter` : Memplot titik data asli
- Menambahkan label sumbu x dan y, serta judul grafik.
- Menambahkan legenda untuk membedakan antara interpolasi dan data asli.
- `plt.grid(True)` : Menambahkan grid pada grafik untuk memudahkan interpretasi.
- `plt.show()` : Menampilkan grafik

Analisi Hasil dari menggunakan metode Polinom Lagrange



Gambar Hasil dari Polinom Lagrange

1. Data Asli (Red Dots) : Titik-titik merah pada grafik menunjukkan titik-titik data asli yang digunakan untuk interpolasi. Dalam kasus ini, ada 8 titik data yang tersebar antara $x=5$ dan $x=40$. Titik-titik ini menjadi dasar bagi polinomial interpolasi yang dihasilkan oleh metode Lagrange.
2. Kurva Interpolasi Lagrange (Blue Line): Garis biru yang mulus menunjukkan hasil interpolasi menggunakan metode Lagrange. Interpolasi Lagrange memberikan polinom yang melewati semua titik data yang diberikan. Oleh karena itu, kurva ini tepat melalui setiap titik merah, menghasilkan kurva kontinu yang berosilasi di antara titik-titik data tersebut.
3. Polinomial Lagrange: Interpolasi Lagrange menghasilkan polinom tingkat tinggi yang mencoba melewati semua titik data. Hal ini sering menyebabkan osilasi yang tidak realistis di antara titik-titik data, terutama jika ada banyak titik data atau titik data tersebut tidak merata. Kurva yang dihasilkan dalam grafik ini menunjukkan osilasi yang signifikan antara beberapa titik data, yang merupakan karakteristik umum dari polinom interpolasi tingkat tinggi.

Kesimpulan

- Interpolasi Lagrange efektif dalam menghasilkan polinomial yang melewati semua titik data asli, memastikan bahwa semua informasi data terwakili dalam model.
- Interpolasi sangat spesifik terhadap data yang ada, sehingga dapat terjadinya overfitting. Hal ini membuat polinomial mungkin kurang baik dalam melakukan prediksi atau ekstrapolasi di luar rentang data yang diberikan.
- Interpolasi Lagrange memastikan kurva melewati semua titik data, kurva interpolasi dapat menunjukkan osilasi yang berlebihan, terutama jika derajat polinomial tinggi atau jarak antar titik data besar. Hal ini bisa terlihat dari beberapa titik puncak dan lembah yang tajam pada kurva.