5η Άσκηση

Τα σημεία της συνάρτησης είναι $0.25, 0.5, 0.75$, άρα ισαπέχοντα. Για την προσέγγιση των παραγώγων πρώτης και δεύτερης τάξης, μπορεί να εφαρμοστούν οι τύποι διαφορών (1) και (2) για $n = 2$. Εν προκειμένω είναι $h = 0.25$ το βήμα και $x = x_1$ το σημείο υπολογισμού των παραγώγων.

$$f'(x) \approx \frac{1}{h}\left[\Delta f_0 + \frac{1}{2}(2\theta - 1)\Delta^2 f_0 + \cdots + \frac{d}{d\theta}\binom{\theta}{n}\Delta^n f_0\right] \tag{1}$$

$$f''(x) \approx \frac{1}{h^2}\left[\Delta^2 f_0 + (\theta - 1)\Delta^3 f_0 + \cdots + \frac{d^2}{d\theta^2}\binom{\theta}{n}\Delta^n f_0\right] \tag{2}$$

Τελικά οι τύποι της πρώτης και δεύτερης παραγώγου δίνονται από τις εξισώσεις (3) και μπορούν εύκολα να υλοποιηθούν σε πρόγραμμα.

$$\begin{aligned} f'(x) &\approx \frac{1}{h}\left(\Delta f_0 + \frac{1}{2}(2\theta - 1)\Delta^2 f_0\right) \\ f''(x) &\approx \frac{1}{h^2}\Delta^2 f_0 \end{aligned} \tag{3}$$

Μετά από εκτέλεση του προγράμματος προκύπτουν τα νούμερα του Πίνακα 1. Καταρχάς, οι πραγματικές τιμές διαφέρουν μόνο ως προς το πρόσημο, λόγω του ότι η $f(x)$ είναι η εκθετική συνάρτηση. Το σφάλμα είναι γενικά μικρό αλλά μπορεί να εκτιμηθεί και το άνω φράγμα του.

	Πραγματική	Αριθμητική
$f'(0.5)$	-0.6065	-0.6128
$f''(0.5)$	0.6065	0.6096

Πίνακας 1: Σύγκριση τιμών παραγώγων

Από θεωρία, είναι γνωστή η σχέση (4), που προέρχεται από την αντίστοιχη σχέση σφάλματος στην παρεμβολή πολυωνύμων και μπορεί να ληφθεί η απόλυτη τιμή της.

$$E_n = \prod_{i=0, i\neq k}^{n} (x_k - x_i)\frac{f^{(n+1)\xi}}{(n+1)!} \tag{4}$$

Ο τελικός υπολογισμός για το φράγμα της πρώτης παραγώγου δίνεται από την (5).

$$
\begin{aligned}
|E_2| &= |x_1 - x_0||x_1 - x_2|\frac{|e^{-\xi}|}{3!} \rightarrow \\
&\leq \frac{e^{-0.25}|x_1 - x_0||x_1 - x_2|}{6} \rightarrow \\
&\approx 0.00811250
\end{aligned}
\tag{5}
$$

Παρακάτω ακολουθεί ο κώδικας που γράφτηκε σε Python και έγινε χρήση της βιβλιοθήκης Numpy. Οι ρουτίνες υπολογισμού εμπρός διαφορών και παραγώγων δίνονται στο Παράρτημα.

ex5.py

```python
import numpy as np
import interpolation


def f(x):
    return np.exp(-x)


def df(x):
    return -np.exp(-x)


def d2f(x):
    return f(x)


xValues = np.array([0.25, 0.5, 0.75])
fValues = f(xValues)
dValues = interpolation.differenceTable(fValues)


print('Actual df(0.5)=', df(0.5))
print('Numeric df(0.5)', interpolation.df(0.5, xValues, dValues)
    )
print('Actual d2f(0.5)=', d2f(0.5))
```

```
print('Numeric d2f(0.5)', interpolation.d2f(0.5, xValues, dValues))

error = f(xValues[0]) / 6 * \
        np.abs((xValues[1]-xValues[0]) * (xValues[1] - xValues[2]))
print('Max error is', error)
```