① La $f, g, h : \mathbb{R}^n \to \mathbb{R}$ være gitt ved

I $f(x,y) := xy + x + 2y^2$

II $g(x,y) := e^{-2xy} - 4$

III $h(x,y,z) := \arctan(x) \cdot \cos(y) - 3z\sin(x)$

IV $p(x,y,z) := \ln(x+y^3) - (x^2y + y^2) \cdot \ln(2z^2)$

I $\frac{\partial f}{\partial x^2} = \frac{\partial}{\partial x}(y+1) \underline{\underline{= 0}}$

$\frac{\partial f}{\partial y^2} = \frac{\partial}{\partial y}(x+4y) \underline{\underline{= 4}}$

$\frac{\partial^2 f}{\partial x \partial y} = \frac{\partial}{\partial y}(y+1) \underline{\underline{= 1}}$

II $\frac{\partial g}{\partial x^2} = \frac{\partial}{\partial x}(-2ye^{-2xy}) \underline{\underline{= 4y^2e^{-2xy}}}$

$\frac{\partial g}{\partial y^2} = \frac{\partial}{\partial y}(-2xe^{-2xy}) \underline{\underline{= 4x^2e^{-2xy}}}$

$\frac{\partial g}{\partial x \partial y} = \frac{\partial}{\partial y}(-2ye^{-2xy}) \underline{\underline{= 4xye^{-2xy}}}$

III $$\frac{\partial h}{\partial x} = \frac{\cos(y)}{1+x^2} - 3z\cos(x)$$

$$\frac{\partial h}{\partial y} = -\arctan(x)\cdot\sin(y)$$

$$\frac{\partial h}{\partial z} = -3\sin(x)$$

$$\frac{\partial^2 h}{\partial x\partial y} = \frac{\partial}{\partial y}\left(\frac{\cos(y)}{1+x^2} - 3z\cos(x)\right) = -\frac{\sin(y)}{1+x^2}$$

IV $$\frac{\partial p}{\partial x} = \frac{1}{x+y^3} - 2xy\ln(2z^2)$$

$$\frac{\partial p}{\partial y} = \frac{1}{x+y^3}\cdot 3y^2 - (x^2+2y)\ln(2z^2)$$

$$= \frac{3y^2}{x+y^3} - (x^2+2y)\ln(2z^2)$$

$$\frac{\partial p}{\partial z} = -\frac{1}{\cancel{2}z^{\cancel{2}}}\cdot\overset{2}{\cancel{4}}\cancel{z}\cdot(x^2y+y^2) = -2\cdot\frac{x^2y+y^2}{z}$$

$$\frac{\partial^3 \varphi}{\partial x \partial y \partial z} = \frac{\partial^2}{\partial y \partial z}\left(\frac{1}{x+y^3} - 2xy\ln(2z^2)\right)$$

Teorem sier at $\frac{\partial^2 i}{\partial y \partial z} = \frac{\partial^2 i}{\partial z \partial y}$. Deriverer altså nå først på z, for å få lettere utregning.

$$\Rightarrow \frac{\partial}{\partial y}\left(-2xy \cdot \frac{1}{\cancel{2}z^{\cancel{2}}} \cdot \cancel{4}^{2}\cancel{z}\right):$$

$$= \frac{\partial}{\partial y}\left(-4 \cdot \frac{xy}{z}\right) = \underline{\underline{-\frac{4x}{z}}}$$

(2) Gitt $\{(x_i, y_i)\}_{i=0}^{n} = \{(3,2), (4,6), (5,12)\}$
der $y_i = f(x_i)$

a) Lagrange:

Vet at $f(x) = \sum_{i=0}^{n} y_i l_i(x) = \sum_{i=0}^{n} \frac{y_i}{q_i(x_i)} q_i(x)$

der $q_i(x) = \prod_{\substack{j=0,\\ j\neq i}}^{n} (x - x_j)$

Vi er gitt y_i-ene, og må da finne $q_i(x_i)$ og $q_i(x)$.

$$q_0(x) = (x-x_1)(x-x_2) = \underline{(x-4)(x-5)}$$

$$q_0(x_0) = (3-4)(3-5) = (-1)\cdot(-2) \underline{= 2}$$

$$q_1(x) = (x-x_0)(x-x_2) = \underline{(x-3)(x-5)}$$

$$q_1(x_1) = (4-3)(4-5) = 1\cdot(-1) \underline{= -1}$$

$$q_2(x) = (x-x_0)(x-x_1) = \underline{(x-3)(x-4)}$$

$$q_2(x_2) = (5-3)(5-4) \underline{= 2}$$

Da er $f(x)$:

$$f(x) = \sum_{i=0}^{2} \frac{y_i}{q_i(x_i)} \cdot q_i(x):$$

$$= \frac{2}{2}(x-4)(x-5) - \frac{6}{1}(x-3)(x-5) + \frac{\cancel{12}^{6}}{\cancel{2}}(x-3)(x-4)$$

$$= x^2 - 9x + 20 - \cancel{6x^2} + 48x - 90 + \cancel{6x^2} - 42x + 72$$

$$\underline{\underline{= x^2 - 3x + 2}}$$

Newton:

Ønsker at $f(x) = \sum_{i=0}^{n} c_i w_i(x)$,

der $w_i(x) = \prod_{k=0}^{i-1} x - x_k$.

Vi må altså finne c_0, c_1 og c_2

$$y_0 = f(x_0) = \sum_{i=0}^{n} c_i w_i(x_0) = \underline{c_0 = 2}$$

$$y_1 = f(x_1) = \sum_{i=0}^{n} c_i w_i(x_0) = c_0 + c_1(x_1 - x_0)$$

$$\Rightarrow \quad c_1 = \frac{y_1 - y_0}{x_1 - x_0} = \frac{6-2}{4-3} = \underline{4}$$

$$y_2 = f(x_2) = \sum_{i=1}^{n} c_i w_i(x_2) = c_0 + c_1(x_2 - x_0) + c_2(x_2 - x_0)(x_2 - x_1)$$

$$\Rightarrow \quad c_2 = \frac{y_2 - y_0 - \frac{y_1 - y_0}{x_1 - x_0}(x_2 - x_0)}{(x_2 - x_0)(x_2 - x_1)}$$

For min egen del vil jeg gå fra uttrykket over til

$$c_2 = \frac{\frac{y_2 - y_1}{x_2 - x_1} - \frac{y_1 - y_0}{x_1 - x_0}}{x_2 - x_0}$$

$$\frac{y_2 - y_0 - \frac{y_1 - y_0}{x_1 - x_0}(x_2 - x_0)}{(x_2 - x_0)(x_2 - x_1)} = \frac{y_2 - y_0 - y_1 \frac{x_2 - x_0}{x_1 - x_0} + y_0 \frac{x_2 - x_0}{x_1 - x_0}}{(x_2 - x_0)(x_2 - x_1)}$$

$$= \frac{y_2 + y_0\left(\frac{x_2 - x_0}{x_1 - x_0} - 1\right) - y_1 \frac{x_2 - x_0}{x_1 - x_0}}{(x_2 - x_0)(x_2 - x_1)}$$

$$\frac{x_2 - x_0}{x_1 - x_0} - 1 = \frac{x_2 - x_0 - x_1 + x_0}{x_1 - x_0} = \frac{x_2 - x_1}{x_1 - x_0}$$

Da er $\frac{x_2 - x_0}{x_1 - x_0} = \frac{x_2 - x_1}{x_1 - x_0} + 1$

$$\Rightarrow \frac{y_2 + y_0 \frac{x_2 - x_1}{x_1 - x_0} - y_1 \cdot \frac{x_2 - x_1}{x_1 - x_0} - y_1}{(x_2 - x_0)(x_2 - x_1)}$$

$$= \frac{y_2 - y_1 - \left(\frac{x_2 - x_1}{x_1 - x_0}\right)(y_1 - y_0)}{(x_2 - x_0)(x_2 - x_1)}$$

$$c_2 = \frac{\frac{y_2 - y_1}{x_2 - x_1} - \frac{y_1 - y_0}{x_1 - x_0}}{x_2 - x_0}$$

Da har vi at

$$c_2 = \frac{\frac{12-6}{5-4} - \frac{6-2}{4-3}}{5-3} = \frac{6-4}{2} = \underline{1}$$

Vi kan da lage polynomet

$$f(x) = \sum_{i=0}^{2} c_i w_i(x) = 2 + 4\cdot(x-3) + (x-3)(x-4)$$

$$= 2 + 4x - 12 + x^2 - 7x + 12$$

$$\underline{\underline{= x^2 - 3x + 2}}$$

som stemmer med det vi fant da vi brukte Lagrange-metoden

Metoden viser også at at for $m+1$ verdier $y_{i_0}, y_{i_1}, \ldots, y_{i_m}$ så er

$$\underline{c_k = [y_0, y_1, \ldots, y_k]} = \frac{[y_0, \ldots, y_k] - [y_0, \ldots, y_{k-1}]}{x_k - x_0}$$

b) Siden Newtons ide bruker foregående verdier for å regne ut neste koeffisient bruker vi denne metoden når vi legger til et datapunkt.

La $f(6) = 26$, dvs $x_3 = 6$ og $y_3 = 26$

Da er

$$c_3 = [y_0, y_1, y_2, y_3] = \frac{[y_1, y_2, y_3] - [y_0, y_1, y_2]}{x_3 - x_0}$$

Vi har $c_2 = [y_0, y_1, y_2] = 1$

$$[y_1, y_2, y_3] = \frac{[y_2, y_3] - [y_1, y_2]}{x_3 - x_1} = \frac{\frac{y_3 - y_2}{x_3 - x_2} - \frac{y_2 - y_1}{x_2 - x_1}}{x_3 - x_1}$$

$$= \frac{\frac{26-12}{6-5} - \frac{12-6}{5-4}}{6-4} = \frac{14-6}{2} = \underline{4}$$

Da er $c_3 = \frac{4-1}{6-3} = \frac{3}{3} = \underline{1}$

Så trenger vi $c_3 w_3(x) = w_3(x)$

$= (x - x_0)(x - x_1)(x - x_2) = (x-3)(x-4)(x-5)$

$= (x^2 - 7x + 12)(x - 5) = x^3 - 12x^2 + 47x - 60$

og så legger vi dette sammen med resultatet vårt fra a:

$$f(x) = x^3 - 12x^2 + 47x - 60 + x^2 - 3x + 2 = \underline{\underline{x^3 - 11x^2 + 44x - 58}}$$

3 Gjør oppgaven i jupyter. Brukte en kanskje litt lite effektiv måte å regne divided difference, men det funker.

4) a) Vet at

$$x_i = \frac{b-a}{2}\tilde{x}_i + \frac{a+b}{2}$$, der $\tilde{x}_i$ er

Chebyshev-nodene $\tilde{x}_i \in [-1, 1]$

Samtidig er

$$\tilde{x}_i = \cos\left(\frac{(2i+1)\pi}{2(n+1)}\right)$$, $i = 0, 1, \dots, n$

Vi finner da nodene på $[-1, 1]$:

$$\tilde{x}_0 = \cos\left(\frac{\pi}{10}\right) \approx 0,951$$

$$\tilde{x}_1 = \cos\left(\frac{3\pi}{10}\right) \approx 0,588$$

$$\tilde{x}_2 = \cos\left(\frac{5\pi}{10}\right) \approx 0$$

$$\tilde{x}_3 = \cos\left(\frac{7\pi}{10}\right) \approx -0,588$$

$$\tilde{x}_4 = \cos\left(\frac{9\pi}{10}\right) \approx -0,951$$

Vi kan deretter finne nodene på $[2,5]$

$$x_0 = \frac{5-2}{2} \cdot 0,951 + \frac{5+2}{2} = \underline{4,927}$$

$$x_1 = \frac{3}{2} \cdot 0,588 + \frac{7}{2} = \underline{4,382}$$

$$x_2 = \underline{3,500}$$

$$x_3 = \frac{3}{2} \cdot (-0,588) + \frac{7}{2} = \underline{2,618}$$

$$x_4 = \frac{3}{2}(-0,951) + \frac{7}{2} = \underline{2,074}$$

Chebyshev-nodene er altså

$$\underline{\{4.927,\ 4.382,\ 3.500,\ 2.618,\ 2.074\}}$$

b) $f(x) = 2^{x^2 - 6x + 9}$

$f(x_0) = 2^{4.927^2 - 6 \cdot 4,927 + 9} = 13,12$

$f(x_1) = 2^{4,382^2 - 6 \cdot 4,382 + 9} = 3,758$

$f(x_2) = 2^{3,500^2 - 6 \cdot 3,500 + 9} = 1,189$

$f(x_3) = 2^{2,616^2 - 6 \cdot 2.618 + 9} = 1,106$

$f(x_4) = 2^{2,074^2 - 6 \cdot 2,074 + 9} = 1.812$

Interpolerer oppg. b og c i jupyter, samt plotte forskjellen (d)

c) Siden vi har funnet 5 Chebyshev noder, antar jeg vi skal interpolere f nå de 5 likedistanserte nodene, ikke 4 som det står i oppg.teksten.

$\bar{x}_i = a + i \cdot \frac{b-a}{n-1} = 2 + i \cdot \frac{5-2}{5} = 2 + i \cdot \frac{3}{4}$

$\bar{x}_0 = 2 + 0 \cdot 0,75 = 2$

$\bar{x}_1 = 2 + 1 \cdot 0,75 = 2,75$

$\bar{x}_2 = 2 + 2 \cdot 0,75 = 3,50$

$\bar{x}_3 = 2 + 3 \cdot 0,75 = 4,25$

$\bar{x}_4 = 2 + 4 \cdot 0,75 = 5$

$$f(\overline{x}_0) = 2^{2^2 - 6\cdot 2 + 9} = 2,000$$
$$f(\overline{x}_1) = f(2,75) = 1,044$$
$$f(\overline{x}_2) = f(3,50) = 1,189$$
$$f(\overline{x}_3) = f(4,25) = 2,454$$
$$f(\overline{x}_4) = f(5) = 16,00$$

d) Gjøres i python sammen med resten av oppg. 4

oppgave3_oving1

September 8, 2021

[7]:
```
# importing useful packages
import numpy as np
import sympy as sp
import re

# setting x equal to the symbol x
x = sp.symbols("x")
```

Given a set of $n+1$ datapoints (x_i, y_i), the interpolate-function returns a polynom p of $deg(p) = n$ that interpolates all of the $n+1$ datapoints. This function uses Newtons idea in which the polynom p is defined as $p(x) = \sum_{i=0}^{n} c_i w_i(x)$ where w_i is the omega function and c_i is a constant and equal to the divided difference of $i+1$ values $[y_0, y_1, ..., y_{i+1}]$.

[8]:
```
def interpolate(data_x, data_y):
    constants = np.array([])
    c_0 = data_y[0]
    c_1 = (data_y[1] - c_0)/(data_x[1] - data_x[0])
    constants = np.append(constants, [c_0, c_1])
    omegas = np.array([1])
    omega_1 = x - data_x[0]
    omegas = np.append(omegas, omega_1)

    if (len(data_x) <= 2):
        return;

    for i in range(2, len(data_x)):
        if (i == len(data_x) - 1):
            c_i = divided_diff(data_x[0:], data_y[0:])
        else:
            c_i = divided_diff(data_x[0:i+1], data_y[0:i+1])

        constants = np.append(constants, c_i)
        omega_i = omegas[i-1] * (x-data_x[i-1])
        omegas = np.append(omegas, omega_i)

    return sp.expand(np.dot(constants, omegas))
```

[9]:
```
def divided_diff(data_x, values):
    if (len(values) == 1):
        return values[0]
    else:
        return ((divided_diff(data_x[1:], values[1:])
                 - divided_diff(data_x[0:-1],values[0:-1]))/(data_x[-1] -␣
↪data_x[0]))
```

[15]:
```
# main program

if __name__ == "__main__":
    data_x = [1976, 1981, 1986, 1991, 1996, 2001]
    data_y = [4017101, 4092340, 4159187, 4249830, 4369957, 4503436]

    """
    num_inp = int(input("Number of data points (int): "))
    for i in range(1, num_inp+1):
        inp = input(f"Data point {i} (format: (x,y)): ")
        inp = re.sub("[(|)]", "", inp)
        data_point = inp.split(",")
        for i in range(2):
            data_point[i] = data_point[i].strip()
        data_x.append(float(data_point[0]))
        data_y.append(float(data_point[1]))
    """
    print(interpolate(data_x, data_y))
```

```
0.0124799999999998*x**5 - 125.693066666665*x**4 + 506293.75146666*x**3 -
1019529128.85412*x**2 + 1026369657797.55*x - 413243740768195.0
```

[21]:
```
def f(x):
    return (0.0124799999999998*x**5
            - 125.693066666665*x**4
            + 506293.75146666*x**3
            - 1019529128.85412*x**2
            + 1026369657797.55*x
            - 413243740768195.0)

print("Population in 1983:", f(1983), "\nActual population: 4122511\n")
print("Population in 1999:", f(1999), "\nActual population: 4445329\n")
print("Population in 2010:", f(2010), "\nActual population: 4858199\n")
print("Population in 2020:", f(2020), "\nActual population: 5367580\n")
```

```
Population in 1983: 4117633.75
Actual population: 4122511

Population in 1999: 4450240.5
```

```
Actual population: 4445329

Population in 2010: 4663065.0
Actual population: 4858199

Population in 2020: 4412780.5
Actual population: 5367580
```

oppgave4_oving1

September 12, 2021

[1]:
```python
# importing useful packages
import numpy as np
import sympy as sp
import re

# setting x equal to the symbol x
x = sp.symbols("x")
```

[3]:
```python
def divided_diff(data_x, values):
    if (len(values) == 1):
        return values[0]
    else:
        return ((divided_diff(data_x[1:], values[1:])
                 - divided_diff(data_x[0:-1],values[0:-1]))/(data_x[-1] -␣
 ↪data_x[0]))
```

[4]:
```python
def interpolate(data_x, data_y):
    constants = np.array([])
    c_0 = data_y[0]
    c_1 = (data_y[1] - c_0)/(data_x[1] - data_x[0])
    constants = np.append(constants, [c_0, c_1])
    omegas = np.array([1])
    omega_1 = x - data_x[0]
    omegas = np.append(omegas, omega_1)

    if (len(data_x) <= 2):
        return;

    for i in range(2, len(data_x)):
        if (i == len(data_x) - 1):
            c_i = divided_diff(data_x[0:], data_y[0:])
        else:
            c_i = divided_diff(data_x[0:i+1], data_y[0:i+1])

        constants = np.append(constants, c_i)
        omega_i = omegas[i-1] * (x-data_x[i-1])
        omegas = np.append(omegas, omega_i)
```

```
    return sp.expand(np.dot(constants, omegas))
```

[6]:
```
# main program

def main():
    data_x = []
    data_y = []

    num_inp = int(input("Number of data points (int): "))
    for i in range(1, num_inp+1):
        inp = input(f"Data point {i} (format: (x,y)): ")
        inp = re.sub("[(|)]", "", inp)
        data_point = inp.split(",")
        for i in range(2):
            data_point[i] = data_point[i].strip()
        data_x.append(float(data_point[0]))
        data_y.append(float(data_point[1]))
    print(interpolate(data_x, data_y))
```

[7]:
```
# 4b, interpolating f(x) on the chebyshev nodes
main()
```

```
Number of data points (int): 5
Data point 1 (format: (x,y)): (2.074, 1.812)
Data point 2 (format: (x,y)): (2.618, 1.106)
Data point 3 (format: (x,y)): (3.5, 1.189)
Data point 4 (format: (x,y)): (4.382, 3.758)
Data point 5 (format: (x,y)): (4.927, 13.12)
1.18052907259596*x**4 - 14.5745806134148*x**3 + 66.9436901695421*x**2 -
135.466478760135*x + 102.993470927311
```

Kommentar Det vil si at polynomet i oppgave 4b er $p_c(x) = 1.181 * x^4 - 14.57 * x^3 + 66.94 * x^2 - 135.5 * x + 103.0$

[8]:
```
# 4c, interpolating f(x) on the equidistributed nodes
main()
```

```
Number of data points (int): 5
Data point 1 (format: (x,y)): (2,2)
Data point 2 (format: (x,y)): (2.75, 1.044)
Data point 3 (format: (x,y)): (3.5, 1.189)
Data point 4 (format: (x,y)): (4.25, 2.954)
Data point 5 (format: (x,y)): (5, 16)
1.20388477366255*x**4 - 14.8435226337449*x**3 + 68.1342716049383*x**2 -
137.76446090535*x + 104.477860082305
```

Kommentar Det vil si at polynomet i oppgave 4c er $p_{eq}(x) = 1.204 * x^4 - 14.84 * x^3 + 68.13 * x^2 - 137.8 * x + 104.5$

Vi vil nå plotte to grafer. Den ene grafen inneholder $d_c(x) = p_c(x) - f(x)$, mens den andre inneholder $d_{eq}(x) = p_{eq}(x) - f(x)$. Samtidig vil også største forskjell bli skrevet ut (globalt toppunkt).

[18]:
```python
# importerer matplotlib
import matplotlib.pyplot as plt

def f(x):
    return 2**(x**2-6*x+9)

def p_c(x):
    return 1.181*x**4 - 14.57*x**3 + 66.94*x**2 - 135.5*x + 103

def p_eq(x):
    return 1.204*x**4 - 14.84*x**3 + 68.13*x**2 - 137.76*x + 104.5

def d_c(x):
    return p_c(x) - f(x)

def d_eq(x):
    return p_eq(x) - f(x)


x = np.linspace(2, 5, 100)

y_c = np.array([])
y_eq = np.array([])
max_diff_c = 0
max_diff_eq = 0

for num in x:
    num_c = d_c(num)
    num_eq = d_eq(num)

    if (num_c > max_diff_c):
        max_diff_c = num_c
    if (num_eq > max_diff_eq):
        max_diff_eq = num_eq

    y_c = np.append(y_c, num_c)
    y_eq = np.append(y_eq, num_eq)


fig, (ax1, ax2) = plt.subplots(1, 2)
fig.subplots_adjust(right=2)
# plot 1
```

```python
ax1.plot(x, y_c)
ax1.set_xlabel("x")
ax1.set_ylabel("d_c(x)")
ax1.set_title("p_c(x) - f(x)")
ax1.axhline()


#plot 2
ax2.plot(x, y_eq)
ax2.set_xlabel("x")
ax2.set_ylabel("d_eq(x)")
ax2.set_title("p_eq(x) - f(x)")
ax2.axhline()

print("Største error for p_c:", max_diff_c)
print("Største error for p_eq:", max_diff_eq)
```

```
Største error for p_c: 1.0688990920123072
Største error for p_eq: 1.4701654998293758
```