MASD Assignment 2

Group 56: wgh762, wqm216, hfc635

September 2021

Exercise 2

a)

i)

Udtrykket $\frac{f(t+\Delta t)-f(t))}{\Delta t}$ giver os ikke den afledte af f i tiden t, da Δt ikke bevæger sig mod 0. Dette udtryk ville derfor give en gennemsnitlige funktionstilvækst i intervallet $[t; t+\Delta t]$.

ii)

Udtrykket $\frac{f(t+5h)-f(t))}{5h}$ giver os vores afledte af f ved tiden t ($\frac{df}{dt}$). Da h bevæger sig mod 0 bliver forskellen mellem $t+5h$ og t udtrykt ved 5h uendelig lille når h går mod 0. Dermed ender vi altså med at få funktionstilvæksten i tiden t.

iii)

Udtrykket $\frac{f(t-\Delta t)-f(t))}{-\Delta t}$ giver også den afledte af f i tiden t ($\frac{df}{dt}$). Da forskellen mellem $(t-\Delta t)$ og t bevæger sig mod nul når Δt bevæger sig mod 0 går vi ligesom ii) får vi dermed funktiontilvæksten i tiden t.

b)

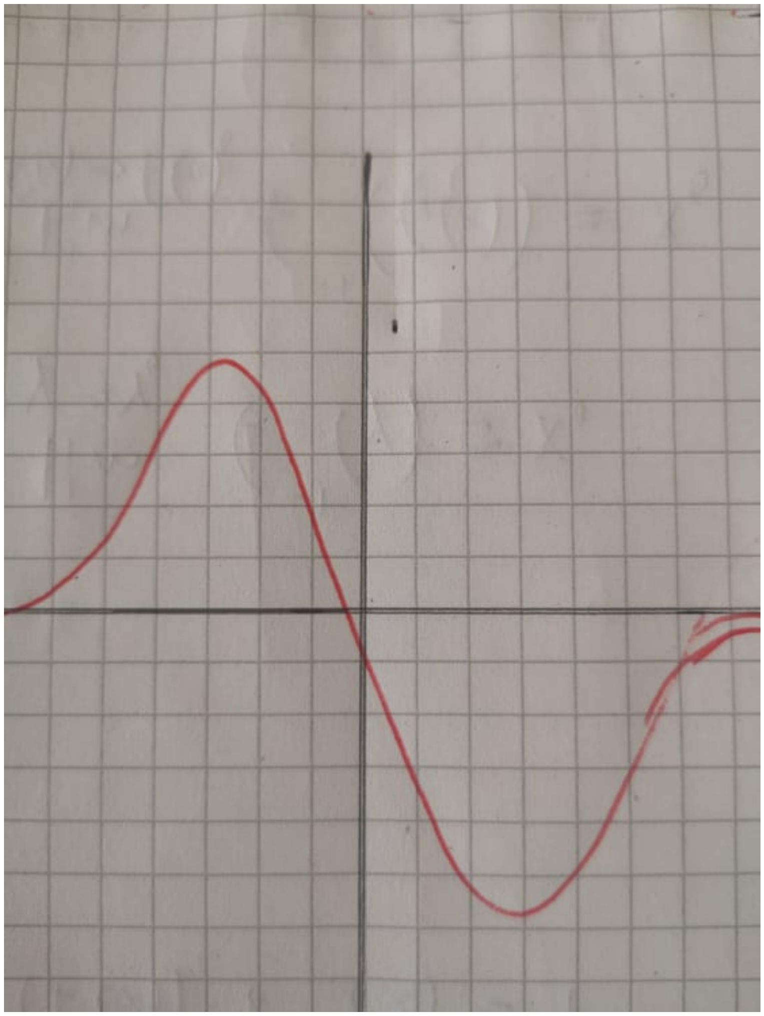

Figure 1: i)

Denne funktion i figur 1 skulle gerne skære origo. Dette var tegneevnerne desværre ikke lige til.

b)

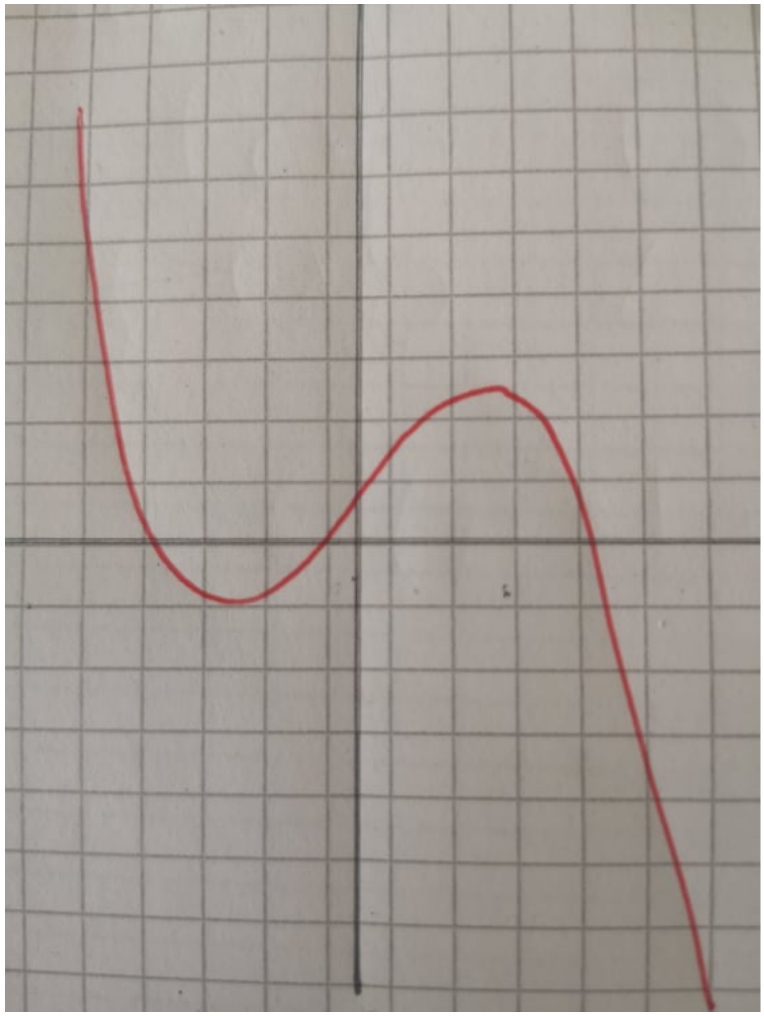

Figure 2: ii)

c)

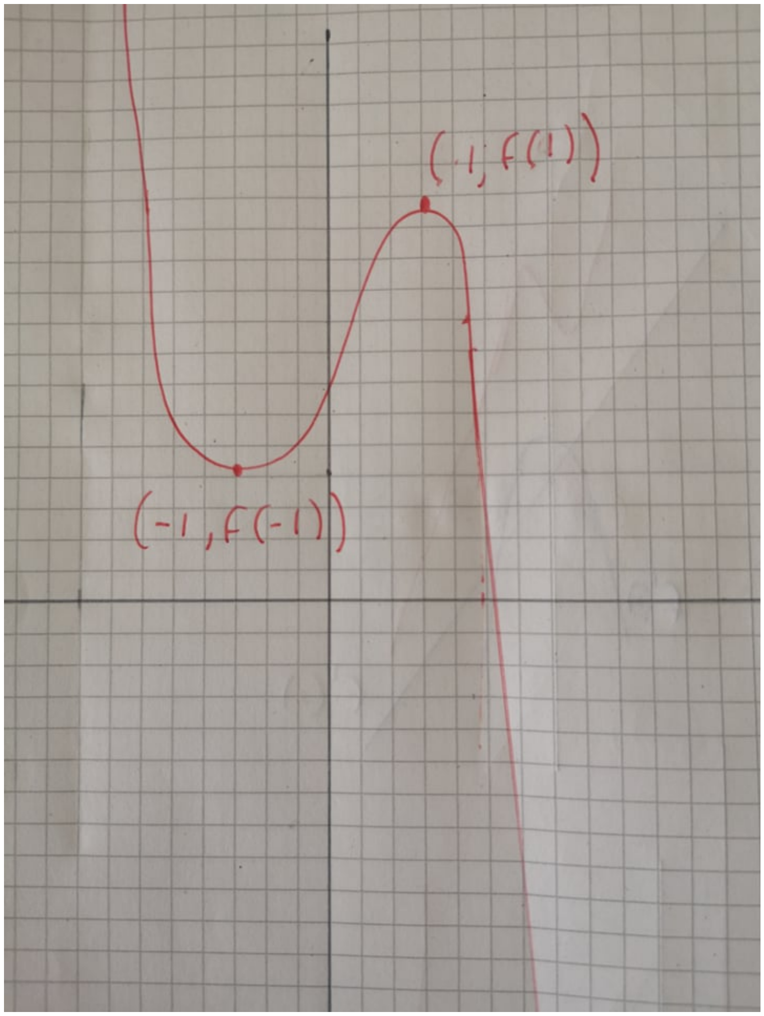

Figure 3: i)

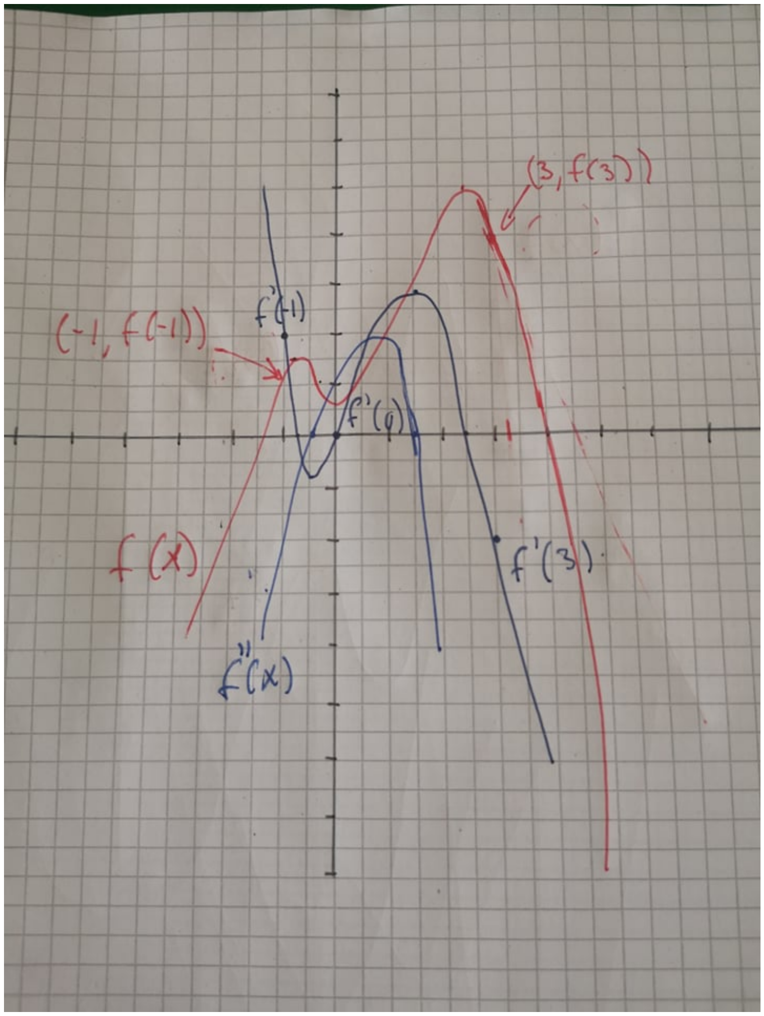

Figure 4: ii)

Exercise 3

a)

$$
\begin{aligned}
\frac{d}{dx}(x^3 + e^{2x}) &= \frac{d}{dx}x^3 + \frac{d}{dx}e^{2x} && \text{Gennem sumreglen} \\
&= 3x^2 + \frac{d}{dx}(2x)e^{2x} && \text{Gennem kædereglen} \\
&= 3x^2 + 2e^{2x}
\end{aligned}
$$

b)

$$\begin{aligned}
\frac{d}{dx}(e^{x^2+3x^3}) &= e^{x^2+3x^3}\frac{d}{dx}(x^2+3x^3) && \text{Gennem kædereglen}\\
&= e^{x^2+3x^3}(\frac{d}{dx}x^2+\frac{d}{dx}3x^3) && \text{Gennem sumreglen}\\
&= e^{x^2+3x^3}(2x+9x^2)
\end{aligned}$$

c)

$$\begin{aligned}
\frac{d}{dx}\frac{lnx}{x^2} &= \frac{2x^2\frac{1}{x}-lnx(2x)}{x^4} && \text{Gennem kvotientreglen}\\
&= \frac{1}{x^3}-2\frac{lnx}{x^3}
\end{aligned}$$

d)

$$\begin{aligned}
\frac{\partial}{\partial x}e^{x^2+3xy+2y^3} &= e^{x^2+3xy+2y^3}\frac{\partial}{\partial x}(x^2+3xy+2y^3) && \text{Gennem kædereglen}\\
&= e^{x^2+3xy+2y^3}(2x+3y)
\end{aligned}$$

e)

$$\begin{aligned}
\frac{\partial}{\partial y}(e^{xy}ln(x^2+y^3)) &= e^{xy}\frac{\partial}{\partial y}ln(x^2+y^3)+ln(x^2+y^3)\frac{\partial}{\partial y}e^{xy} && \text{Gennem produktreglen}\\
&= e^{xy}\frac{1}{x^2+y^3}\frac{\partial}{\partial y}(x^2+y^3)+ln(x^2+y^3)e^{xy}\frac{\partial}{\partial y}(xy)\\
& && \text{Gennem dobbelt brug af kædereglen}\\
&= \frac{3y^2e^{xy}}{x^2+y^3}+ln(x^2+y^3)e^{xy}x
\end{aligned}$$

f)

til at starte med:

$$\begin{aligned}
x^TAx &= x^T(\sum_{k=1}^{n}a_{1k}x_k \ldots \sum_{k=1}^{n}a_{nk}x_k)^T\\
&= \sum_{h=1}^{n}x_h\sum_{k=1}^{n}a_{hk}x_k\\
&= \sum_{h=1}^{n}\sum_{k=1}^{n}x_h a_{hk}x_k
\end{aligned}$$

Fra dette udregnes:

$$\frac{\partial}{\partial x_i}(x^T Ax) = \sum_{h=1}^{n}\sum_{k=1}^{n}(x_h a_{hk} x_k)$$
$$= \sum_{h \neq i} x_h a_{hi} + \sum_{k \neq i} a_{ik} x_k + 2a_{ii}x_i$$

Hvor $2a_{ii}x_i$ kommer fra udtrykket $a_{ii}x_i + x_i a_{ii}$

$$= \sum_{h=1}^{n} x_h a_{hi} + \sum_{k=1}^{n} a_{ik} x_k$$

Dette sættes så i noget der refererer til matricerne:
Det skal dog først noteres at følgende udtryk er sandt:

$$\sum_{h=1}^{n} x_h a_{hi} = x^T a_{.,i}$$

Hvor $a_{(.,i)}$ er den i'te kolonne af A. På samme måde så:

$$\sum_{k=1}^{n} a_{ik} x_k = a_{(i,.)}^T x$$

Hvor $a_{(i,.)}$ er en kolonnevektor der svarer til den i'te række af A. Med disse notationer, findes:

$$\frac{\partial}{\partial x_i}(x^T Ax) = x^T a_{(.,i)} + a_{(i,.)}^T x = a_{(.,i)}^T x + a_{(i,.)}^T x$$

Den sidste lighed er sand da:

$$u^T v = u_1 v_1 + u_2 v_2 + ... + u_n v_n = v_1 u_1 + v_2 u_2 + ... + v_n u_n = v^T u$$

g)

Det vides at:

$$\nabla x(x^T Ax) = (\frac{\partial}{\partial x_1}(x^T Ax), ..., \frac{\partial}{\partial x_n}(x^T Ax))$$

Og at:

$$\frac{\partial}{\partial x_i}(x^T Ax) = x^T a_{(.,i)} + a_{(i,.)}^T x = a_{(.,i)}^T x + a_{(i,.)}^T x$$

Samlet bliver det:

$$\nabla x(x^T Ax) = (A^T + A)x$$

Exercise 4

See the attached .ipynb for the jupyter notebook. Below is a transscript of the code.

a

```
def badness(a, b, cho, nob):
    # A compact implemntation of the formular from the assigment.
    # The inner part of the formular is first calculated, then summed
    # together and then timed with 1/20
    return (1/20)*sum((nob-a*cho-b)**2)
```

b

```
    def badnessgradient (a,b,cho,nob) :
    # A compact implementation of the formular from the assigment.
    # First the inner is differentialet accoring to a,b.
    # The results are then summed up and timed with 1/20
    return (1/20 * sum(np.diff((nob-a*cho-b)**2,1)), 1/20 * sum (np.diff((nob-a*cho-b)**2,2)))
```

c

```
# Starting values of a = 1 and b = 0
ab = (1,0)

# Number for our "steps"
steps =0.0001

# a while loop that checks if badness is under 50
while 110 > badness(ab[0],ab[1],cho,nob) > 50 :

    # A implementation of the formular form the assigmemt
    ab = (ab[0] + steps * badness(ab[0],ab[1],cho,nob) , ab[1] -
    steps * badness(ab[0],ab[1],cho,nob))

# printing and plotting the results

print (ab)
print (badness(ab[0],ab[1],cho,nob))
plotdatafit(ab[0],ab[1],cho,nob,countries)
```

Our values for a, b and badness is:

- a = 2.359541676999574
- b = -1.3595416769995745
- badness = 49.98669001213736

Our plot looks like this

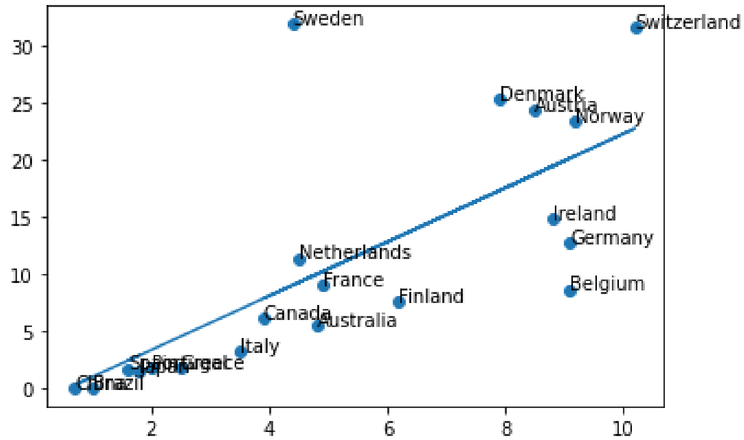

Figure 5: Our plot