
Matematikk 1000

Øvingsoppgaver i numerikk – leksjon 5

Litt oppsummering undervegs

Løsningsforslag

Oppgave 1 – Et skjæringspunkt

$$\begin{aligned}f(x) &= \sqrt{x} - e^{-x} \\g(x) &= 1 + \arctan \frac{x}{5} .\end{aligned}$$

a) Vi kan lage plottet slik i kommando-vinduet:

```
>> x=0:1e-2:5;
>> f=sqrt(x)-exp(-x);
>> g=1+atan(x/5);
>> plot(x,f,'linewidth',2)
>> hold on
>> plot(x,g,'r','linewidth',2)
>> hold off
>> set(gca,'fontsize',15)
```

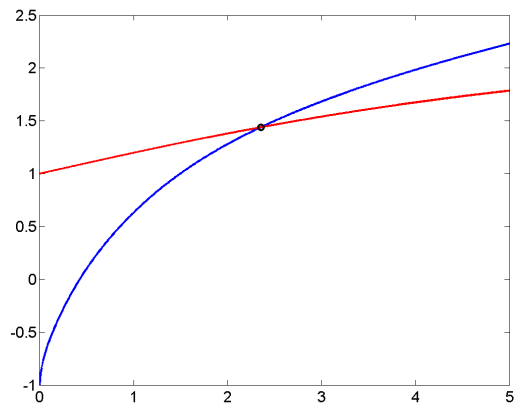
Her har vi valgt å la x gå fra 0 til 5 i steg på $10^{-2} = 0.01$. Vi har plotta f blå og g rød, og vi har valgt å gjøre tallene på aksene litt større. Vi har vist plottet i figur 1.

b) Vi skal løse likninga $f(x) = g(x)$, eller $h(x) = 0$ der

$$h(x) = f(x) - g(x) = \sqrt{x} - e^{-x} - 1 - \arctan \frac{x}{5} .$$

Vi ser tydelig at $h(0)$ negativ og $h(5)$ er positiv:

$$\begin{aligned}h(0) &= \sqrt{0} - e^0 - 1 - \arctan 0 = -2 < 0 \\h(5) &= \sqrt{5} - e^{-5} - 1 - \arctan 1 \approx 0.4439 > 0 .\end{aligned}$$



Figur 1: Svaret i oppgave 1 a) og c).

Videre er h en kontinuerlig funksjon. Vi kan derfor bruke intervallhalveringsmetoden med start-intervallet $[0, 5]$. (Selsvagt kunne vi ha starta med andre intervall også – så lenge det inneholder nullpunktet.) Vi tar utgangspunkt i skriptet vi lagde i leksjon 4 og gjør de nødvendige justeringene:

```

1  % Implementering av intervallhalveringsmetoden for likninga
2  % sqrt(x)-cos(x)=0 med a=0 og b=pi/2 som start-grenser
3
4  % Grenser
5  a=0;
6  b=5;
7
8  % Presisjon
9  Pres=1e-5;
10
11 % Funksjonsverdier
12 Fa=FunkTilIntHalv(a);
13 Fb=FunkTilIntHalv(b);
14
15 % Starter while-løkke som kjører til ønsket presisjon
16 while abs(b-a)>Pres
17     c=(a+b)/2;          % Midtpunktet
18     Fc=FunkTilIntHalv(c); % Funksjonsverdi i midtpunktet
19     if Fa*Fc<0
20         b=c;            % Setter ny b til c
21     else
22         a=c;            % Setter ny a til c
23     end
24 end
25

```

```

26 % Regner ut nytt midtpunkt og skriver svaret til skjerm
27 x=(a+b)/2

```

Vi har fiksert grensene i linje 5 og 6. Presisjonen har vi satt i linje 9. (Denne er egentlig satt litt for strengt, for vi deler det endelige intervallet i to enda en gang – i linje 27.) I tillegg må vi oppdatere funksjonsfila, som blir kalt i linje 12, 13 og 18:

```

1 function f=FunkTilIntHalv(x)
2
3 % Funksjon som vi skal finne nullpunktet til
4 % Blir brukt av skriptet IntervallHalvering
5
6 f=sqrt(x)-exp(-x)-1-atan(x/5);

```

Vi kjører skriptet og plotter løsninga (deloppg. c):

```
>> IntervallHalvering
```

```
x =
```

```
2.3568
```

```

>> f=sqrt(x)-exp(-x);
>> hold on
>> plot(x,f,'ko','linewidth',2)
>> hold off

```

Merk at vektoren **x** har blitt skrevet over av skriptet **IntervallHalvering**; etter at vi har kjørt det, er **x** bare et tall.

Oppgave 2 – Noen summer

- a) Vi så hvordan vi kunne regne ut summer i leksjon 3. Den første summen kan regnes ut med dette skriptet:

```
1  % Skript som regner ut en sum
2
3  % Setter summen til 0
4  S=0;
5
6  for i=1:100
7      S=S+i;          % Legger til et ledd
8  end
9
10 % Skriver summen til skjerm:
11 S
```

I kommandovinduet får vi

```
>> SummeSkript
```

```
S =
```

```
5050
```

```
>> n=100;
```

```
>> n*(n+1)/2
```

```
ans =
```

```
5050
```

Her har vi også kontrollert at vi får det samme svaret med den gitte formelen.

For å regne ut neste sum, justerer vi linje 6 (grensene) og linje 7 (leddet) til

```
for i=5:500
    S=S+1/i-1/(i+1);      % Legger til et ledd
end
```

Dette gir svaret 0.1980, eller med litt flere desimaler: 0.198003992. Dette kunne vi også ha funnet eksakt i dette tilfellet; summen er slik at leddene kansellerer hverandre:

$$\sum_{i=5}^n \left(\frac{1}{i} - \frac{1}{i+1} \right) = \left(\frac{1}{5} - \frac{1}{6} \right) + \left(\frac{1}{6} - \frac{1}{7} \right) + \left(\frac{1}{7} - \frac{1}{8} \right) + \dots$$
$$+ \left(\frac{1}{n-1} - \frac{1}{n} \right) + \left(\frac{1}{n} - \frac{1}{n+1} \right) = \frac{1}{5} - \frac{1}{n+1}$$

Det eksakte svaret er altså $1/5 - 1/501$, noe som gir akkurat det samme som over skrevet med 9 desimaler.

- b) Vi tar utgangspunkt i det samme skriptet igjen. Siden vi skal jusetere den øvre grensa flere ganger, kan det være en fordel å la denne grensa være en variabel som vi bestemmer i starten (se linje 4 og 9):

```

1  % Skript som regner ut en sum
2
3  % Bestemmer den øvre grensa
4  N=10;
5
6  % Setter summen til 0
7  S=0;
8
9  for i=-5:N
10     S=S+1/sqrt(i^2+1);          % Legger til et ledd
11 end
12
13 % Skriver summen til skjerm:
14 S

```

Vi får svaret 5.4582 når vi kjører det. Det at vi kaller indeksen i i skriptet, ikke k slik som i oppgaveteksten, spiller ingen rolle. Når vi så skal la n ta andre verdier, endrer vi bare linje 4 i skriptet, lagrer og kjører det på nytt. For $n = 100$, $n = 1000$ og $n = 10000$ får vi svarene 7.7144, 10.0124 og 12.3146. Om vi tillater oss å fortsette med 100000 og 1000000, får vi svarene 14.6171 og 16.9197. Selv om summen øker veldig sakte når n øker, er det lite som tyder på at summen går mot noen bestemt verdi.

Det er forresten interessant å legge merke til hvor lite tid maskina bruker på å regne ut summen – selv med en million ledd. Min kontor-pc brukte 3 hundredels sekund på jobben.

- c) Vi beholder N som variabel og justerer **for**-løkka:

```

for i=0:N
    S=S+i^2*2^(-i/2);          % Legger til et ledd
end

```

Med de samme n -verdiene som over, får vi svarene 33.8443, 48.0416, 48.0416 og 48.0416. Med fire desimaler ser vi ingen forskjell på de tre siste svarene. Om vi tar med litt flere desimaler (**> format long**), får vi svarene 33.844261793466430, 48.041630560319675, 48.041630560342625 og 48.041630560342625. Selv ikke med 15 desimaler ser vi forskjell på de to siste summene. Mye tyder på at denne summen går mot noe endelig når den øvre grensa n går mot uendelig.

- d) Her er det snakk om såpass mange justeringar at vi har valgt å lage eit nytt skript:

```

1  % Skript som regner ut summen av f(x_i) der x_i=(i-1)/2
2  % og f(x) = x^(3/2) for i fra og med 1 til og med 8.
3
4  % Setter summen til 0
5  S=0;
6
7  for i=1:8
8      xi=(i-1)/2;
9      S=S+xi^(3/2);
10 end
11
12 % Ganger summen med 1/2 og skriver til skjerm
13 Svar=0.5*S

```

Når vi kjører skriptet, får vi svaret 10.8580.

- e) Vi gjør de tilsvarende generaliseringene i skriptet (inkludert kommentarene):

```

1  % Skript som regner ut summen av f(x_i) der x_i=(i-1)*Delta x,
2  % med Delta x=4/n og f(x) = x^(3/2) for i fra og med 1 til og med n.
3  % n blir fiksert i skriptet
4
5  % Fikserer øvre summasjonsgrense
6  n=10;
7
8  % Bestemmer Delta x
9  DeltaX=4/n;
10 % Setter summen til 0
11 S=0;
12
13 for i=1:n
14     xi=(i-1)*DeltaX;
15     S=S+xi^(3/2);
16 end
17
18 % Ganger summen med Delta x og skriver til skjerm
19 Svar=DeltaX*S

```

Med n lik 10, 50 og 100 får vi svarene 11.2374, 12.4816 og 12.6404. Om vi tillater oss å ta med 500, 1000 og 10000 også, får vi 12.7680, 12.7840 og 12.7984. Det ser ut til å gå mot en bestemt verdi når n øker. Vi regner ut integralet:

$$\int_0^4 x^{3/2} dx = \left[\frac{1}{\frac{3}{2} + 1} x^{3/2+1} \right]_0^4 = \frac{2}{5} \left(4^{5/2} - 0^{5/2} \right) = \frac{64}{5} = 12.8 .$$

Integralet ligger ganske nær det vi regna ut over. Dette er selvsagt ikke tilfeldig.

Ekstra-oppgave – Mer om bursdager

Ut fra hintet vi får, kan vi se at sannsynligheten for at n elever har bursdag på ulike datoer blir

$$\frac{365}{365} \cdot \frac{364}{365} \cdot \frac{363}{365} \cdot \dots \cdot \frac{365 - n + 1}{365} .$$

Spørsmålet er: Hva må n være for at dette produktet skal bli mindre enn $1/2$?

Det kan vi finne ut på flere måter. En måte kan være å lage et lite skript med ei `while`-løkke der vi for hver iterasjon multipliserer med neste faktor helt til produktet bikker $1/2$. Dette kan gjøres slik:

```
1 % Skript som regner ut hvor mange elever det må være i en
2 % skoleklasse for at der mest sannsynlig vil være minst to
3 % elever med bursdag på samme dag
4
5 % Setter sannsynlighente P til én
6 P=1;
7 % Setter antall elever til null
8 elever=0;
9
10 while P>1/2
11     elever=elev+1;           % Øker antall elever med 1
12     P=P*(365-elev+1)/365;    % Oppdaterer sannsynlighet
13 end
14
15 % Skriver resultatet til skjerm
16 elever
```

Vi kjører skriptet, som vi har kalt `SammeBursdag.m`, i kommandovinduet:

```
>> SammeBursdag
```

```
elev =
```

```
23
```

Med andre ord må vi regne med det at i en gjennomsnittlig skoleklasse vil være elever med samme fødselsdag.