Aflevering 6

Lucas Bagge

a)

Dette er en individuel opgave. Opgaveløsning afleveres i blackboard som én pdf fil under "Upload af afleveringsopgaver > Aflevering 4". Afleveringsfristen bestemmes af din instruktor, men ligger i uge 14.

Betragt rummet af differentiable funktioner $f\colon [0, \ \ ] \to \mathbb{R}$ med det L^2-indre produkt

$$\langle f, g \rangle = \int_0 \ \ f(w)g(w)\, d.$$

(a) Vis at samlingen bestående af funktionerne

$$\sin(w), \sin(2w), \ldots, \sin((j-1)w) \qquad \textbf{(8.1)}$$

er ortogonal

I ovenstående opgave beskrivelse ligner det der mangler et tegn i grænserne for integralet. Her følger jeg 13.10 og antager at der skal stå π.

Her kan vi følge ovenstående nævnte eksempel og vise følgende:

$$\int_0^{\pi} sin(m \cdot w)sin(n \cdot w)dx = 0, \text{ hvor } m \neq n$$

Hvor vi kan bruge følgende identitet:

$$sin(\theta)sin(\phi) = \frac{1}{2}[cos(\theta - \phi) - cos(\theta + \phi)]$$

Således kan ovenstående skrives som:

$$\int_0^{\pi} \frac{1}{2}[cos(mw - nw)) - cos(nw + mw)]$$

Smid 1/2 og w ud

$$\frac{1}{2}\int_0^{\pi} cos(m-n)w - cos(n+m)w$$

Hvor vi nu kan integrerer

$$\frac{1}{2}[\frac{1}{m-n}sin(m-n)w - \frac{1}{m+n}sin(m+n)w]_0^{\pi}$$

Her får vi at sin(m-n) bare er nul, så år vi evaluerer udtrykket så får vi nul. Dermed får vi nul og vi har de er ortogonale.

b)

(b) Gør rede for at funktionen $f(w) \equiv 1$ er ikke vinkel ret på $\sin(w)$ eller på $\sin(3w)$, men er vinkelret på $\sin(2w)$ og $\sin(4w)$.

For at vise dette kan vi bare beregne det indre produkt af funktionerne

$$\langle 1, sin(w)\rangle = \int_0^\pi 1 \cdot sin(w)\ dw = [-cos(w)]_0^\pi = 2$$

Ikke vinkelret.

$$\langle 1, sin(3w)\rangle = \int_0^\pi 1 \cdot sin(3w)\ dw = \frac{1}{3}[-cos(w)]_0^{3\pi} = \frac{2}{3}$$

Ikke vinkelret.

$$\langle 1, sin(2w)\rangle = \int_0^\pi 1 \cdot sin(3w)\ dw = \frac{1}{2}[-cos(w)]_0^{2\pi} = 0$$

Vinkelret

$$\langle 1, sin(4w)\rangle = \int_0^\pi 1 \cdot sin(3w)\ dw = \frac{1}{4}[-cos(w)]_0^{4\pi} = 0$$

Den er vinkelret.

Således har jeg vist at f(w)=1 er vinkelret på sin(2w) og sin(4w), men ikke vinkelret for sin(w) og sin(3w).

c)

(c) Bestem projektionen af $f(w) \equiv 1$ langs $\sin(w)$, $\sin(3w)$ og find derfra en funktion af formen $1 + c_1 \sin(w) + c_3 \sin(3w)$, som er vinkelret på både $\sin(w)$ og $\sin(3w)$. Forklar hvorfor denne kombination er vinkelret på alle funktioner i (8.1) for $j = 5$.

Opskriv projektionen:

$$pr_{sin(x),sin(3w)}(1) = \frac{\langle 1, sin(w)\rangle}{||sin(w)||^2} sin(w) + \frac{\langle 1, sin(3w)\rangle}{||sin(3w)||^2} sin(3w)$$

Herefter foretages der en del udregning af det indre produkt og normen, men ender ud med:

$$1 - \frac{4}{\pi} sin(w) - \frac{4}{3\pi} sin(3w)$$

Denne kombination er vinkelret på alle funktioenr i 8.1, da det er et eksempel på en Gram Schmidt process, hvor ideen er at vi kan trække fra vores vektor en projektion på hver vektor (her sin w, sin 2w, sin 3w og sin 4w) så vi får en vekto der er ortogonal til hver vektor. For vores *f*, som er ortogonal til sin (m w) når m er positiv og lige. Derfor skal man trække projektionen med sin w og sin 3w fra.

d)

(d) For $j = 4$, bestem i python den lineære kombination af funktionerne i (8.1), som ligger tættest på funktionen

$$f(w) = 1 - e^{-w}.$$

Plot funktionen og dens tilnærmelse.

I denne opgave skal vi bruge python til at plotte ovenstående funktion og beregne en approximering baseret på vores funktion givet i 8.1.

Allerførst indlæser modulerne numpy og matplotli.

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

Efterfølgende bruger jeg samme metode som Andrew vise i hans undervisning, men hvor han viste en fourier cosinus udvikling.

```
n = 100
x, h = np.linspace(0, np.pi, n, retstep = True)
print("De først 10 obervationer i x: \n",x[:10])
```

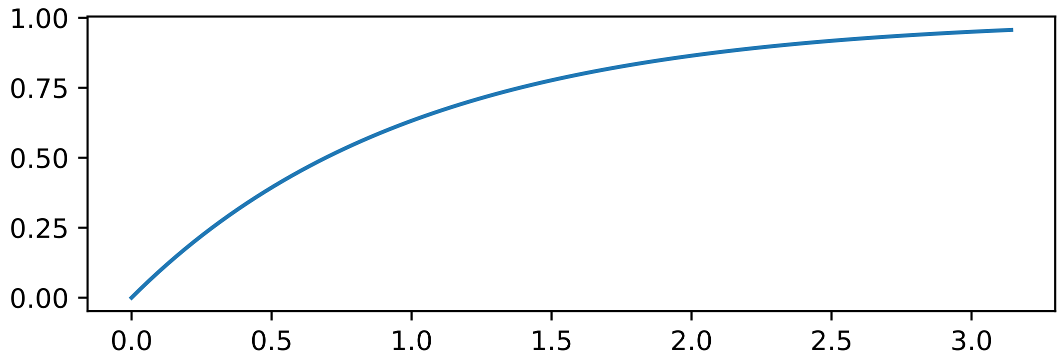
```
## De først 10 obervationer i x:
##  [0.         0.03173326 0.06346652 0.09519978 0.12693304 0.1586663
##  0.19039955 0.22213281 0.25386607 0.28559933]
```

```
print("Vores h observation, som er pi: \n", h)
```

```
## Vores h observation, som er pi:
##  0.031733259127 16963
```

Nu vil jeg plotte vores eksponential funktion.

```
fig, ax = plt.subplots()
ax.set_aspect("equal")
ax.plot(x, 1-np.exp(-x))
plt.show()
```



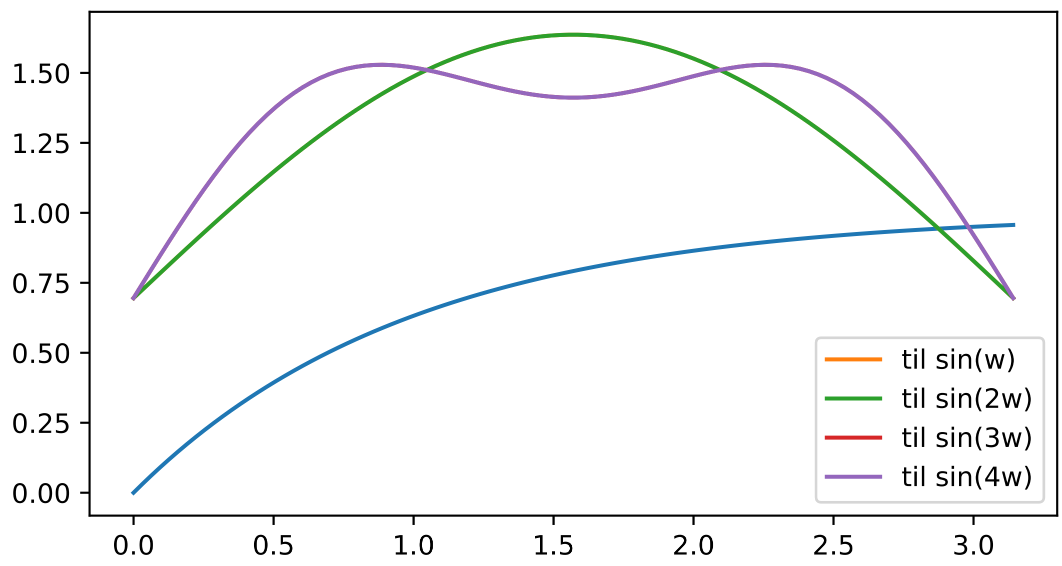
Herefter benytter jeg mig ad de tre funktioner vi så i undervisning:

- `indre_produkt` den beregner det indre produkt ved hjælp af **Trapezreglen**.
- ´nor_sq´ beregner længden.
- `proj` beregner projektionen for os.

```
def indre_produkt(f, g, h):
  return np.trapz(f * g, dx = h)

def nor_sq(f, h):
  return indre_produkt(f, f, h)

def proj(f, k, x, h):
  konstant = np.ones_like(x)
  out = indre_produkt(f, konstant, h) / nor_sq(konstant, h) * konstant
  for m in range(1, k, 2):
    out += indre_produkt(f, np.sin(m * x), h) / nor_sq(np.sin(m * x), h) * np.sin(m * x)
  return out
```

```
f = 1 - np.exp(-x)

fig, ax = plt.subplots()
ax.set_aspect('equal')
ax.plot(x, f)
ax.plot(x, proj(f, 2, x, h), label = "til sin(w)")
ax.plot(x, proj(f, 3, x, h), label = "til sin(2w)")
```

```
ax.plot(x, proj(f, 4, x, h), label = "til sin(3w)")
ax.plot(x, proj(f, 5, x, h), label = "til sin(4w)")
plt.legend(loc = "lower right")
plt.show()
```



Her ser vi at funktionen sin(w), sin(2w) ligger oveni hinanden og det samme gælder for sin(3w) og sin(4w). Vores tilnærmelse er ikke særlig god og vi kommer ikke rigtig tilnærmelsesvis tæt på.