

Foldning

Foldning og Sandsynlighed

Foldning er et centralt matematisk værktøj, der bruges til at analysere og kombinere funktioner. 3Shape bruger foldning til at kombinere data fra billeder, af tænderne, med en såkaldt vægtningsmatrix. Resultatet er et tal som angiver om tænderne er i fokus eller ude af fokus. 3Shape kan bruge denne information til at bestemme afstanden til dele af tænderne og derved lave en 3D model.

Vi vil i dette afsnit beskrive egenskaber for den diskrete foldning og derefter den kontinuerte foldning. I gymnasiet arbejder vi for det meste med kontinuerte funktioner, $f(x)$, men det er lettere at forstå den diskrete foldning.

Vi starter med en diskret sandsynlighedsmodel for summen af to terningkast, og fortsætter med en generel beskrivelse af foldningens matematik og dens praktiske anvendelser.

Diskret Sandsynlighed og Foldning

Terningkast

For en fair terning, vil sandsynligheden for et udfald være $P = \frac{1}{6}$. Hvis vi tager summen af kast med to terninger bliver udfaldsrummet $U = [2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12]$, men hvad er sandsynligheden for de enkelte udfald?

Vi kalder udfaldene for den ene terning X og udfaldene for den anden for Y . Sandsynlighederne skrives som $P_X(m)$ og $P_Y(m)$, hvor m angiver ternindsslaget, eks. $P_X(3) = \frac{1}{6}$, da der er en sjættedel chance for at slå 3 med en terning.

Hvis vi antager at de to terningkast er uafhængige er sandsynligheden for at få en sum på 4 givet ved,

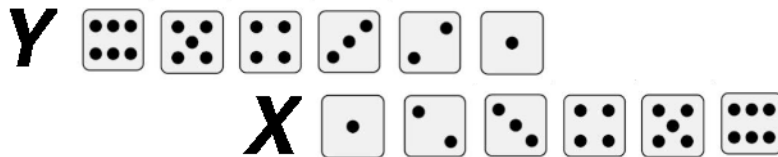
For at regne de samlede sandsynligheder summere man sandsynlighederne for de enkelte udfald. Der er tre måder man kan få fire på, $[1, 3]$, $[2, 2]$, $[3, 1]$ så sandsynligheden kan også beregnes ved

$$P_{X+Y}(4) = 3 \cdot \frac{1}{36} = \frac{1}{12}.$$

Det er relativt nemt at udregne sandsynlighederne for udfaldsrummet for to terningkast ved at tælle kombinationer. En mere generel måde er at bruge foldningen mellem X og Y . Den samlede sandsynlighedsfordeling kan udregnes med proceduren for foldning:

- Flip Y så den kommer til at være $[6,5,4,3,2,1]$
- Forskyd den ind over X .
- For hver forskydning tæl antallet af kombinationer, altså hvor stort overlappet er, ved at forskyde Y .

I figur xx ses X og Y hvor Y er forskudt så den lodrette sum er lig 4.



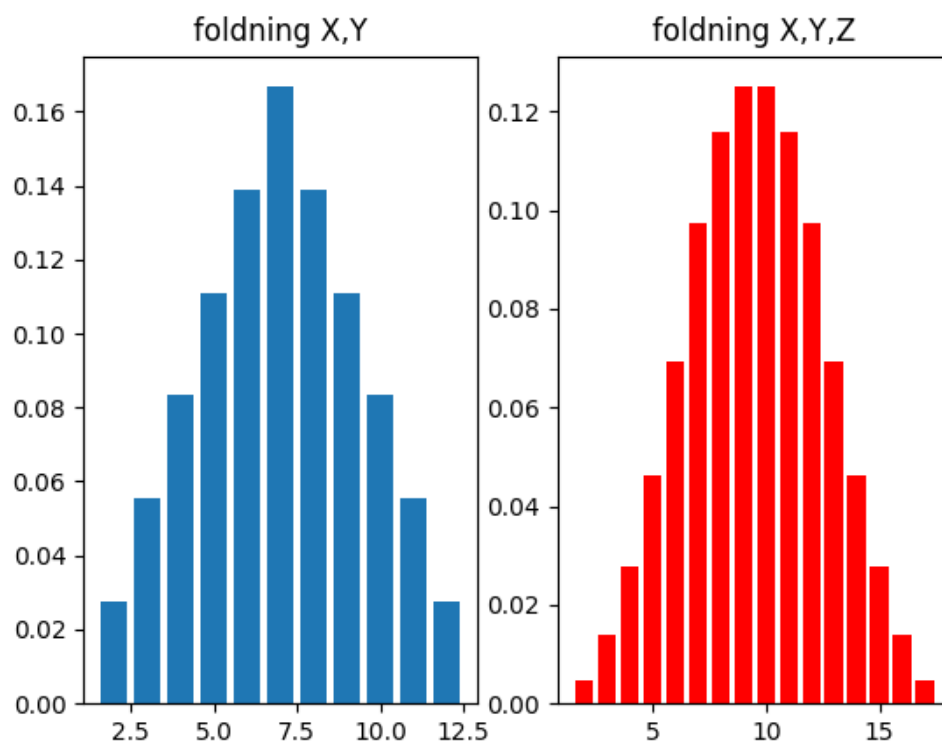
Forskydning af Y så summen bliver 4.

Det ses at ske for 3 kombinationer $[1,3], [2,2], [3,1]$, hvilket giver sandsynlighed $P_{X+Y}(4) = 3 \cdot \frac{1}{36} = \frac{1}{12}$.

Øvelse

- Tegn X og Y som ovenfor men hvor summen bliver til 5 i stede for 4.
- Vis at den samlede foreling bliver $P_{X+Y}(m) = \frac{1}{36} [1,2,3,4,5,6,5,4,3,2,1]$, hvor m angiver indekset i listen.

Foldningen giver altså ikke værdien af summen af terningsslagene men sandsynligheden for at få dem. I figur xx ses foldning af to terningkast og tre terningkast. Udfaldene for hver enkelt terning er ens, hvilket kan skrives som $P_X = P_Y = \frac{1}{6} \cdot [1,1,1,1,1,1]$.



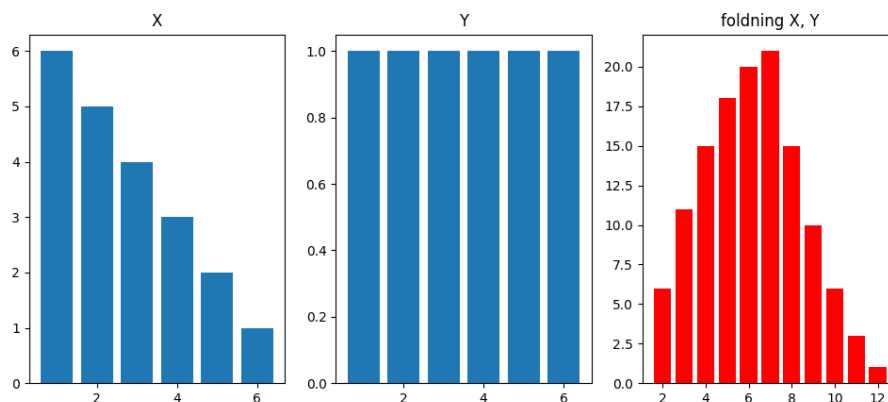
Foldning af to og tre terninger

For to terningkast er x-aksern $[2,12]$ mens den for tre terningkast er $[3,18]$.

Vi kan godt lave foldninger med fordelinger som ikke summere til 1, men så kan vi ikke længere fortolke dem som sandsynlighedsfordelinger.

Eksempel, unfair terning

Vi har to terninger og har indsamlet terningslag for dem hver. Hyppigheden af de enkelte slag er vist i figur xx hvor det ses, at terning X langt oftere slår lave slag end høje slag mens terning Y er fair.



unfair terninger

Foldningen er vist med rødt og vi kan igen godt fortolke diagrammet som at det er mest sandsynligt at få en sum på omkring 6 – 7 mens det er ret usandsynligt at få summer omkring 11 og 12.

I Maple kan det laves med

```
with(SignalProcessing);
with(Statistics);
X := [6, 5, 4, 3, 2, 1];
Y := [1, 1, 1, 1, 1, 1];
Z := Convolution(X, Y);
ColumnGraph(Z, offset = 1.5);
```

I Python med

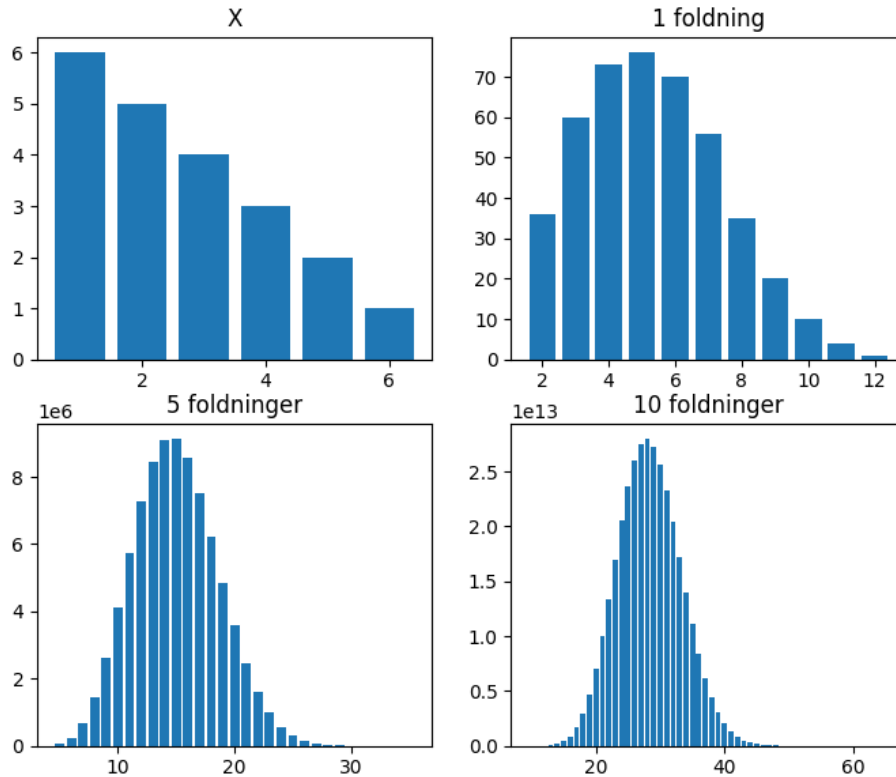
```
import numpy as np
from matplotlib import pyplot as plt
X = [6,5,4,3,2,1]
Y = [1,1,1,1,1,1]
Z = np.convolve(X,Y)
t = np.arange(2,13)
plt.bar(t,Z)
plt.show()
```

Øvelse

For at vise, at det nu er hyppigheder vi arbejder med skrives $H_X = [6,5,4,3,2,1]$ og $H_Y = [1,1,1,1,1,1]$.

- Beregn foldningen mellem H_X og H_Y i hånden.
- Brug computeren til at lave en foldning med resultatet og H_X .
- Tegn et søjlediagram for resultatet.

Figur xx viser fordelingen ved et kast, summen af to, fem og ti kast.



ti foldninger af unfair terning

Det ligner mere og mere en normalfordeling og assymetrien i den oprindelige fordeling er helt forsvundet.

Matematisk notation

Foldning mellem to variable bliver skrevet med en asterix, *. Foldningen mellem X og Y skrives som $X * Y$.

Ved de fair terninger, svarende til en ligefordeling, med $H_X = H_Y = [1,1,1,1,1,1]$ og $X = Y = [1,2,3,4,5,6]$ bliver foldningen som vist tidligere,

$$H_X * H_Y = [1,2,3,4,5,6,5,4,3,2,1].$$

Ovenfor har vi arbejdet med sandsynligheder, P , og hyppigheder, H . Vi vil fremover bruge funktionerne $f[m]$ og $g[m]$, hvor f og g kunne svare til P_X og P_Y eller H_X og H_Y eller noget helt tredje.

De diskrete funktioner kan skrives som lister,

$$f[m] = [x_1, x_2, x_3, x_4, \dots], g[m] = [y_1, y_2, y_3, y_4, \dots].$$

m angiver det element vi kigger på, eks. $f[2] = x_2$.

Proceduren for en foldning er

- Flip $g[m]$.
- Forskyd $g[m]$ ind over $f[m]$.
- Udregn overlap ved at gange værdierne og summere dem.

Vi kigger igen på terningkast, så $g[m] = [y_1, y_2, y_3, y_4, y_5, y_6]$. y -værdierne svarer til sandsynlighederne og indekset til det antal af øjne. Den omvendte rækkefølge kan dannes med $g[7 - m]$, så for $m = 2$ får man $g[7 - 2] = g[5] = y_5$.

Øvelse

- Udfyld tabellen

m	1	2	3	4	5	6
$f[m]$	x_1					
$g[6 - m]$						

Forskydningen sker ved at ændre på n i $g[n - m]$. I ovenstående eksempel var $n = 7$ og Y var forskudt ind over X så den lodrette sum gav 7. Hvis vi vælger et tal forskelligt fra $n = 7$ bliver overlappet ikke fuldt.

Eksempel

Hvis $n = 4$ kan tabellen udfyldes med

m	1	2	3	4	5	6
$f[m]$	x_1	x_2	x_3	x_4	x_5	x_6
$g[4 - m]$	y_3	y_2	y_1	y_0	y_{-1}	y_{-2}

I nederste række har jeg fjernet y_0, y_{-1}, y_{-2} -værdierne da indekstallene ikke svarer til et element i listen.

m	1	2	3	4	5	6
$f[m]$	x_1	x_2	x_3	x_4	x_5	x_6
$g[4 - m]$	y_3	y_2	y_1	0	0	0

Hvis vi addere indekstallet lodret ser vi at summen netop giver 4. Den lodrette sum giver, indeks: $x_1 + y_3 \rightarrow 1 + 3 = 4$, indeks: $x_2 + y_2 \rightarrow 2 + 2 = 4$ og indeks: $x_3 + y_1 \rightarrow 3 + 1 = 4$.

Generelt gælder at indeks: $f[m] + g[n - m] \rightarrow m + n - m = n$, altså netop summen n . Ved terningkast giver indekset n netop antallet af øjne, så vi får kombinationen af terning 1 og terning 2 som giver summen, n .

For at få den endelige foldning skal vi nu gange sandsynlighederne eller hyppigheden når resultatet er n .

Hvis vi kigger på to fair terninger igen, bliver tabellen

m	1	2	3	4	5	6
$f[m]$	1/6	1/6	1/6	1/6	1/6	1/6
$g[4 - m]$	1/6	1/6	1/6	0	0	0

Foldningen $(f * g)[4]$ giver

$$(f * g)[4] = \frac{1}{6} \cdot \frac{1}{6} + \frac{1}{6} \cdot \frac{1}{6} + \frac{1}{6} \cdot \frac{1}{6} + \frac{1}{6} \cdot 0 + \frac{1}{6} \cdot 0 + \frac{1}{6} \cdot 0 = 3 \cdot \frac{1}{36} = \frac{1}{12}$$

Vi er nu klar til definitionen af den diskrete foldning.

Definiton - diskret foldning

Den diskrete foldning af funktionerne $f[m]$ og $g[m]$ er givet ved

$$(f * g)[n] = \sum_{m=-\infty}^{\infty} f[m]g[n - m],$$

hvor n og m tilhører heltallene. I definitionen løber m fra minus uendeligt til uendeligt, men som vist ovenfor er det nok at løbe over alle elementer i de to funktioner som bliver foldet.

Eksempel

Vi kigger igen på to fair terninger med $f = g = \frac{1}{6}[1,1,1,1,1,1]$ og $X = Y = [1,2,3,4,5,6]$.

Foldningen $(f * g)[4]$ giver altså sandsynligheden for at få tallet 4 med to terninger. Det giver

Det føles nok som en ret besværlig måde at udfregne det på, men den generelle definition kan heldigvis bruges i rigtig mange situationer.

Øvelse

- Udregn $(f * g)[2]$ og $(f * g)[6]$ på samme måde og tjek ovenfor om I får det samme?

Anvendelse af diskret foldning

Gennemsnit

Et simpelt gennemsnit kan skrives som

$$\bar{f} = \frac{1}{M} \sum_1^M f = \frac{1}{M} (x_1 + x_2 + \dots + x_M),$$

hvilket er foldningen mellem $f = [x_1, x_2, \dots, x_M]$ og $g = [1/M, 1/M, \dots, 1/M] = \frac{1}{M} [1, 1, \dots, 1]$.

Foldningen bliver

$$(f * g) = \frac{1}{M} \sum_1^M f \cdot [1, 1, \dots, 1] = \frac{1}{M} (x_1 + x_2 + \dots + x_M).$$

Eksempel

Figur xx viser sinuskurven og et eksempel på 11 målinger, de røde.

Vi kan nu opskrive de to datasæt, $f =$

$[0., 0.588, 0.951, 0.951, 0.588, 0., -0.588, -0.951, -0.951, -0.588, 0.]$ og $g =$

$\frac{1}{11} [1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1]$.

Øvelse

- Opskriv den diskrete foldning og udregn gennemsnittet.

Glidende gennemsnit

Ved et glidende gennemsnit tages gennemsnittet af M punkter rundt om punktet n . For at det skal være symmetrisk omkring n skal M være et ulige heltal. Det giver

$$\bar{f}[n] = \frac{1}{M} \sum_{m=n-\frac{1}{2}(M-1)}^{m=n+\frac{1}{2}(M-1)} f[m].$$

Øvelse

- Overvej hvorfor M skal være ulige.

Eksempel

Vi lader $M = 3$ hvilket betyder, at vi tager et gennemsnit af punkterne før og efter samt midterpunktet. Vi lader igen data komme fra en sinussvingning, $f =$

$[0., 0.588, 0.951, 0.951, 0.588, 0., -0.588, -0.951, -0.951, -0.588, 0.]$ og foldningen bliver

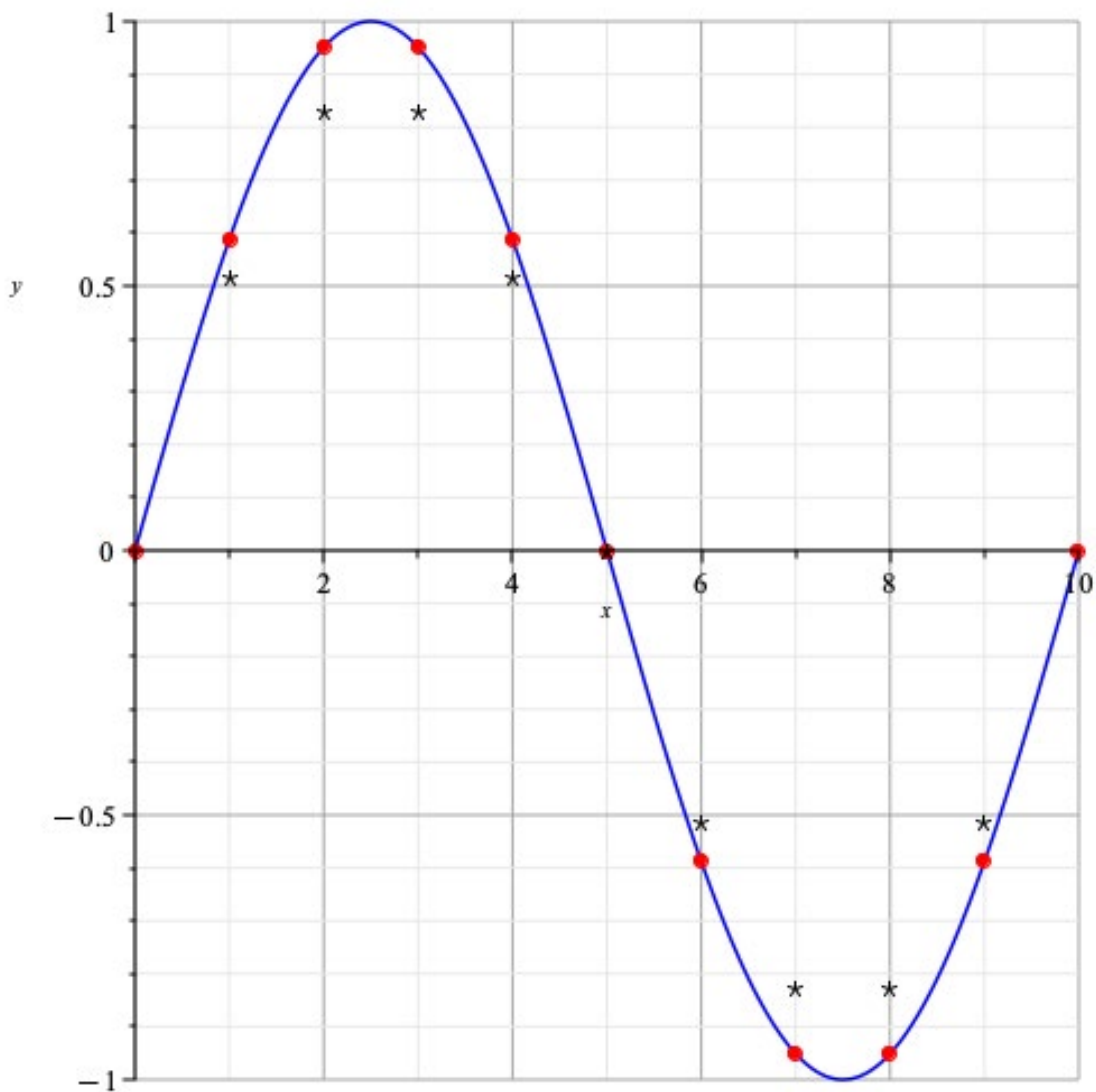
$$\bar{f}[n] = \frac{1}{3} \sum_{m=n-\frac{1}{2}(3-1)}^{m=n+\frac{1}{2}(3-1)} f[m] = \frac{1}{3} \sum_{m=n-1}^{m=n+1} f[m] = \frac{1}{3}(f[n-1] + f[n] + f[n+1]).$$

n	1	2	3	4	5	6	7	8	9	10	11
$f[n]$	0	0.588	0.951	0.951	0.588	0	-0.588	-0.951	-0.951	-0.588	0
$\bar{f}[n]$	—	0.513	0.830	0.830	0.513	0	-0.513	-0.830	-0.830	-0.513	—

hvor eks. $\bar{f}[3] = \frac{1}{3}(0.588 + 0.951 + 0.951) = 0.830$. Der er forskellige måder at klare enderne på, her har jeg forkorte listen så der kun er værdier hvor hele gennemsnittet kan beregnes.

Ved denne version af et glidende gennemsnit reduceres antallet af datapunkter med $M - 1$.

Nedenfor ses sammen sinuskurve med et glidende gennemsnit som *.



Sinus og diskret glidenden gennemsnit

Øvelse

Her er 101 datapunkter og det skal vi ikke regne i hånden.

$X = [0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23, 24, 25, 26, 27, 28, 29, 30, 31, 32, 33, 34, 35, 36, 37, 38, 39, 40, 41, 42, 43, 44, 45, 46, 47, 48, 49, 50, 51, 52, 53, 54, 55, 56, 57, 58, 59, 60, 61, 62, 63, 64, 65, 66, 67, 68, 69, 70, 71, 72, 73, 74, 75, 76, 77, 78, 79, 80, 81, 82, 83, 84, 85, 86, 87, 88, 89, 90, 91, 92, 93, 94, 95, 96, 97, 98, 99, 100]$

$Y = [0.58, 0.49, 0.45, 0.58, 0.38, 0.92, 0.45, 0.99, 0.68, 0.90, 1.00, 0.64, 1.00, 0.96, 1.00, 0.93, 0.96, 0.95, 0.99, 1.00, 0.99, 0.99, 0.70, 0.98, 0.93, 0.93, 0.55, 0.84, 0.56, 0.92, 0.48, 0.92, 0.13, 0.23, 0.33, 0.79, 0.65, -0.08, 0.26, 0.57, -0.17, 0.04, 0.11, -0.17, -0.19, 0.03, -0.51, 0.17, -0.54, -0.72, -0.29, -$

0.77, -0.26, -0.19, -0.89, -0.31, -0.96, -0.55, -0.96, -0.58, -0.81, -0.80, -0.92, -0.81, -0.95, -1.00, -0.99, -0.99, -0.98, -1.00, -1.00, -0.98, -0.99, -0.99, -0.77, -0.83, -0.93, -0.77, -0.48, -0.71, -0.93, -0.40, -0.46, -0.38, -0.25, -0.68, -0.02, -0.18, -0.10, -0.23, 0.07, -0.29, -0.13, 0.18, -0.24, 0.57, 0.25, 0.42, 0.64, 0.68, 0.62]

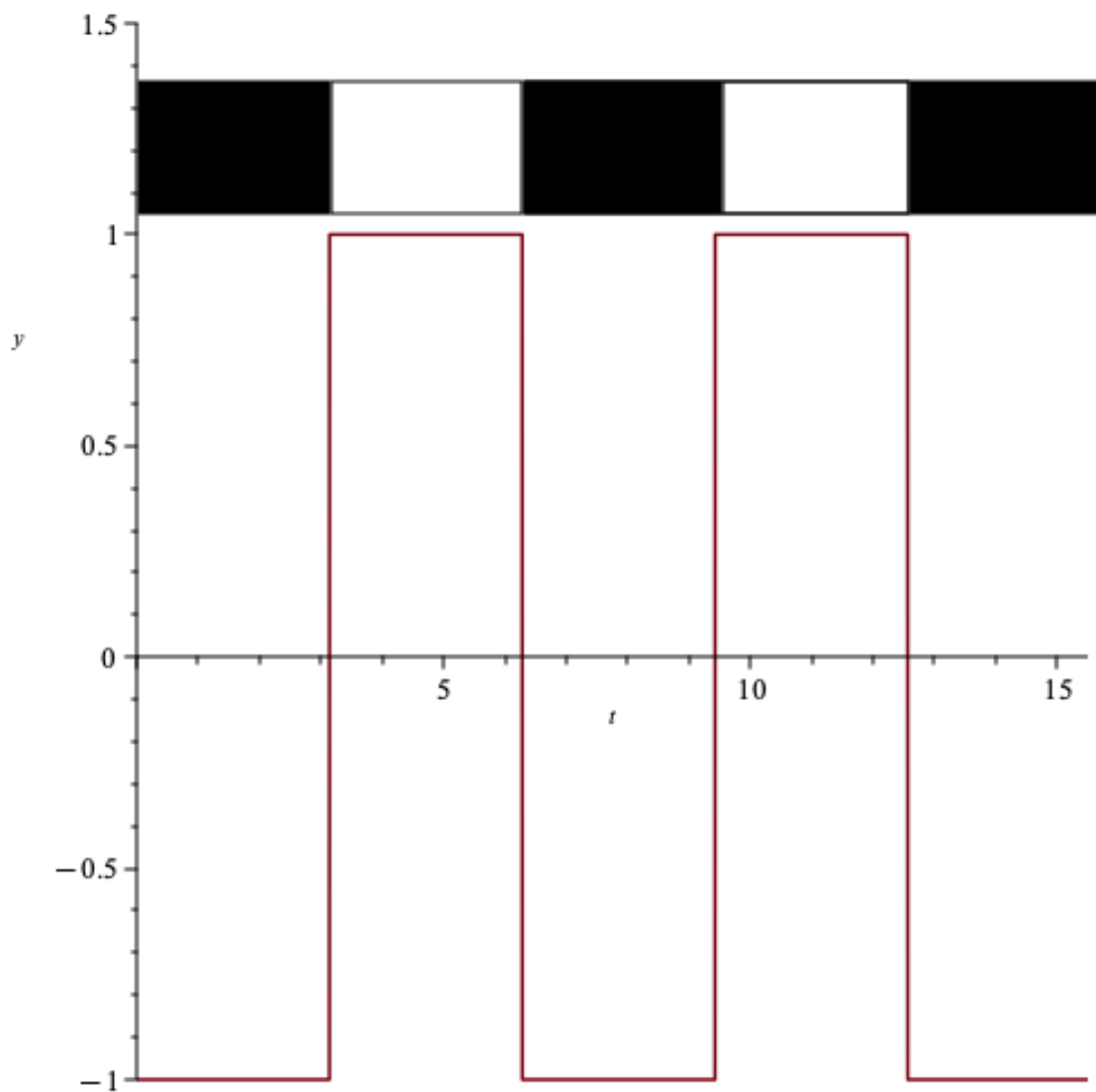
- Lav et punktplot af datasættet. De fleste matematikprogrammer kan lave glidende gennemsnit, moving average på engelsk. I Maple skal man brug *with(Statistics)* og *MovingAverage(Y, M)* for at beregne det glidende gennemsnit.
- Lav et glidende gennemsnit med $M = 5$ over data og lav et punktplot for resultatet.
- Prøv at forøg M og argumenter for hvad der sker med resultatet.
- Lav en liste med $g = [1,1,1,1,1] \cdot 1/5$ og lav foldningen med Y , det skulle gerne give det samme som det glidende gennemsnit.

Ovenstående er et eksempel på et glidende gennemsnit til at reducere støj i et datasæt, hvilket bruges ved databehandling. Når det bliver brugt i elektronik kalder man det et low pass filter.

Diskret foldning og billedbehandling

Billeder kan beskrives ved deres pixelværdier for hver (x, y) koordinat, hvor hjørnet øverst til venstre har værdien $(0,0)$. Farvebilleder kan beskrives ud fra mængden af rød, grøn og blå, kaldet RGB værdier. Her er det standard at arbejde med værdier fra 0 – 255, hvor 0 svarer til ingen farve mens 255 svarer til fuld farve. Hvis man vil have en rød farve skrives det som $(255,0,0)$, altså intet grøn eller blå farve. Man kan danne andre farver ved at blande primærfarverne eks. har gul farve en RGB værdi på $(255,255,0)$. Når vi skal lave foldning med billeder bliver vi nødt til at give hver pixel en værdi. Vi vil holde os til gråtoner for at holde det simple.

Nedenfor er billedet at sorte og hvide felter baskrevet med de diskrete værdier $g = [-1,1, -1,1, -1]$, hvor -1 svarer til sort og 1 hvid og g er 5 lang.



Sort hvid

Hvis vi gerne vil gøre billedet mere sløret kan vi eks. lave en foldning med $f = [0.1, 0.8, 0.1]$, hvor vi lader hver pixel være et vægtet gennemsnit af sig selv og naboernes værdier.

Foldningen er

$$(f * g)[n] = \sum_{m=1}^{m=3} f[m]g[n-m]$$

skrevet ud bliver det.

$$(f * g)[2] = 0.1 \cdot (-1) + 0.8 \cdot 1 + 0.1 \cdot (-1) = 0.6$$

$$(f * g)[3] = 0.1 \cdot 1 + 0.8 \cdot (-1) + 0.1 \cdot 1 = -0.6$$

$$(f * g)[4] = 0.1 \cdot (-1) + 0.8 \cdot 1 + 0.1 \cdot (-1) = 0.6$$

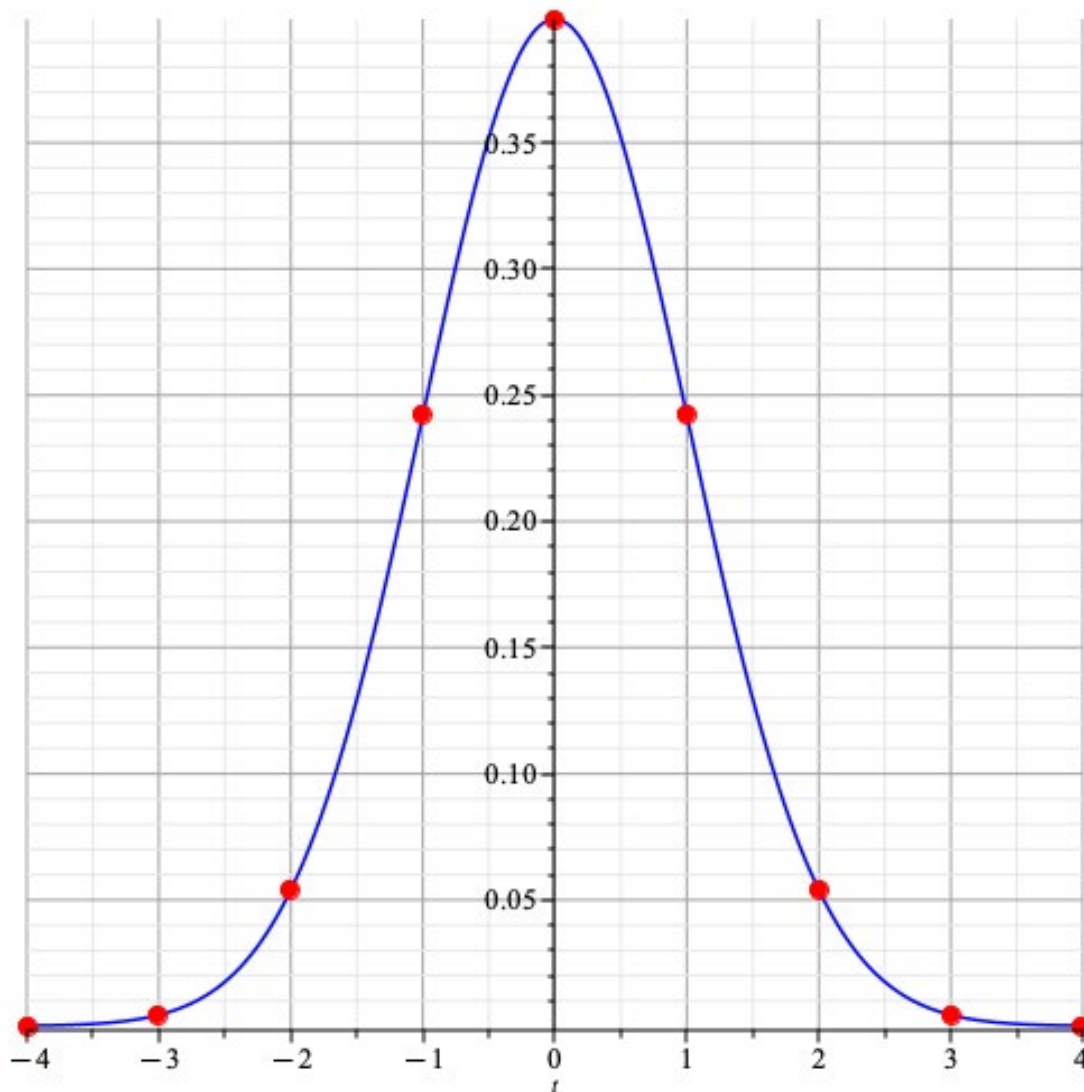
Vores output at foldningen bliver listen, $h = [0.6, -0.6, 0.6, -0.6]$. I forhold til $g = [-1, 1, -1, 1]$ er kontrasten mindre.

Øvelse

- Vælg en anden vægtfordeling og lav en foldning med firkant-funktionen, g .
- Overvej hvad der sker hvis summen af vægtene ikke er 1.

Gauss foldning

Vi viser nedenfor at gauss-fordelingen er brugbar når billeder skal sløres.



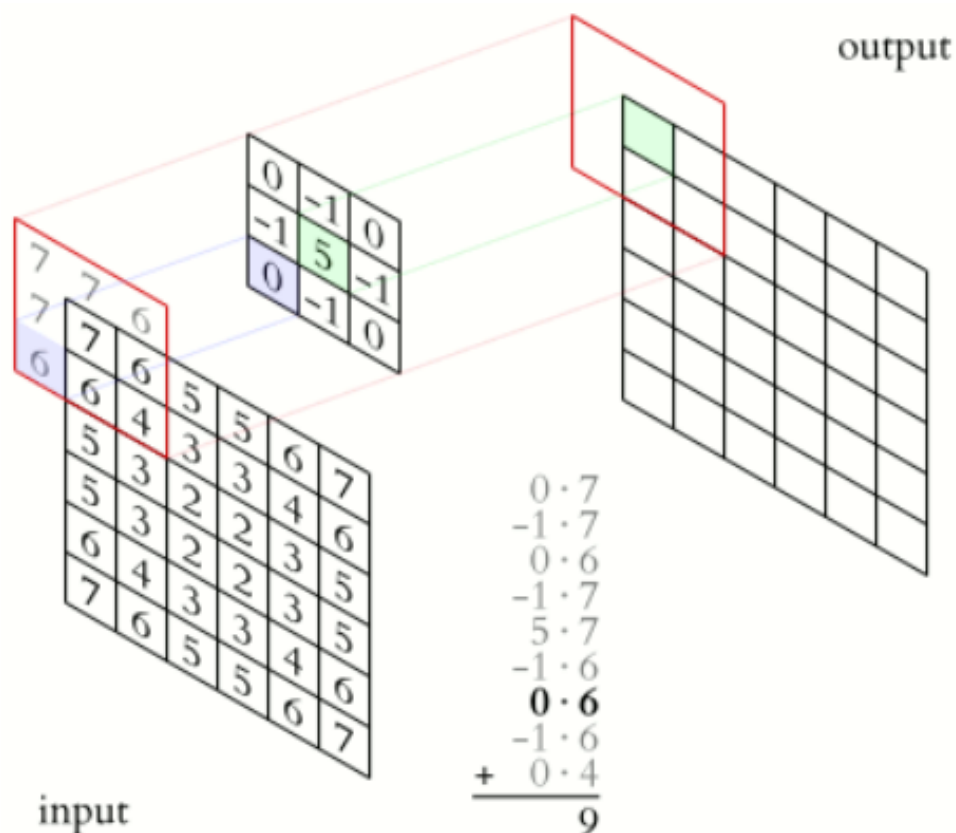
normal_diskret

Figuren viser den kontinuerte gauss-fordeling med middelværdi $\mu = 0$ og spredning $\sigma = 1$. De røde punkter giver værdierne til en diskret foldning, $f = [0.000, 0.004, 0.054, 0.242, 0.399, 0.242, 0.054, 0.004, 0.000]$. Med tre decimaler afrundes fire gange spredning til nul og pixelværdierne vil være et vægtet gennemsnit af de 7 omkringliggende pixels. Når billedredigeringsprogrammer laver Gauss-blur, sløring, er den parameter man indstiller netop spredningen σ . Større σ gør billedet mere sløret.

2D foldning

Det man skal folde med er nu også to-dimensionel og beskrives med en N, M foldningsmatrix.

Figuren viser en animation af processen, kan findes her, [wikipedia convolution](#).



2d foldning.

I det her tilfælde bliver den midterste pixelværdi ganget med 5 og der bliver fratrasket værdien af de fire omkringliggende pixelværdier fra. Dette giver et tal som overskriver den midterste pixelværdi. Derefter flytter filtret en plads og det gentages. Den vægtmatrix der er blevet brugt er,

$$w = \begin{bmatrix} 0 & -1 & 0 \\ -1 & 5 & -1 \\ 0 & -1 & 0 \end{bmatrix}.$$

Foldningen kan skrives op som

$$(f * w)[x, y] = \sum_{i=-[N/2]}^{[N/2]} \sum_{j=-[M/2]}^{[M/2]} f(x + i, y + j) \cdot w(i, j).$$

Værdien i venstre hjørne af illustrationen ovenfor beregnes som

$$(f * w)[0, 0] = f(-1, 0) \cdot w(-1, 0) + f(0, -1) \cdot w(0, -1) + f(0, 0) \cdot w(0, 0) + f(0, 1) \cdot w(0, 1) + f(1, 0) \cdot w(1, 0)$$

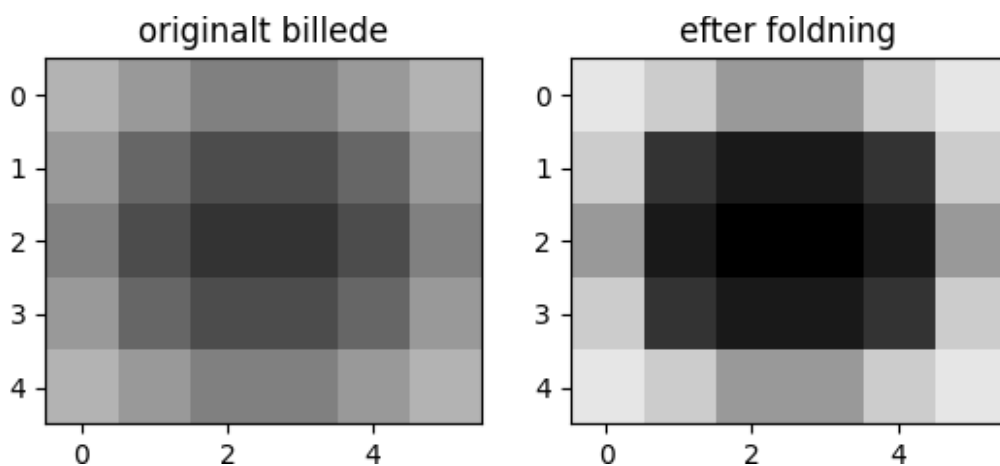
$$(f * w)[0,0] = 7 \cdot (-1) + 7 \cdot (-1) + 7 \cdot 5 + 6 \cdot (-1) + 6 \cdot (-1) = 9$$

På animationen kan man se at de yderste pixel-værdier bliver beregnet ved at kopiere kendte værdier ind der hvor der mangler i foldningen. Der er mange metoder til at behandle siderne og hjørnerne, men for normale billeder hvor der er mange pixels er det simplest at smide de yderste væk.

Øvelse

- Beregn værdien for $(f * w)[1,0]$

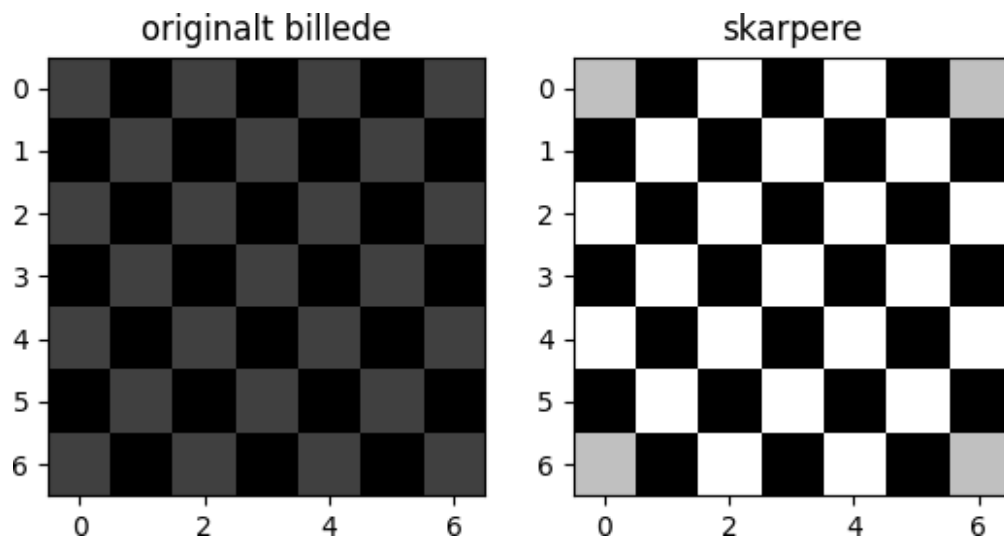
Nedenfor er lavet en foldning med et gråtonebillede. Her er lave værdier mørke og høje værdier lyse. Billedet er efter foldning blevet skarpere i den forstand, at mørke regioner er blevet mørkere mens lyse er blevet lysere.



2dfoldning_sharp

Nedenfor er et billede af et skakternet mønster som er foldet med samme vægt-matrice, w . Her er det tydeligt at foldningen gør billedet skarpere.

$$\begin{bmatrix} 1 & 0 & 1 & 0 & 1 & 0 & 1 \\ 0 & 1 & 0 & 1 & 0 & 1 & 0 \\ 1 & 0 & 1 & 0 & 1 & 0 & 1 \\ 0 & 1 & 0 & 1 & 0 & 1 & 0 \\ 1 & 0 & 1 & 0 & 1 & 0 & 1 \\ 0 & 1 & 0 & 1 & 0 & 1 & 0 \\ 1 & 0 & 1 & 0 & 1 & 0 & 1 \end{bmatrix} * \begin{bmatrix} 0 & -1 & 0 \\ -1 & 5 & -1 \\ 0 & -1 & 0 \end{bmatrix} \Rightarrow \begin{bmatrix} 3 & -3 & 4 & -3 & 4 & -3 & 3 \\ -3 & 5 & -4 & 5 & -4 & 5 & -3 \\ 4 & -4 & 5 & -4 & 5 & -4 & 4 \\ -3 & 5 & -4 & 5 & -4 & 5 & -3 \\ 4 & -4 & 5 & -4 & 5 & -4 & 4 \\ -3 & 5 & -4 & 5 & -4 & 5 & -3 \\ 3 & -3 & 4 & -3 & 4 & -3 & 3 \end{bmatrix}$$



Foldning af skaktern

Igen ses uønskede effekter ved hjørnerne.

Øvelser

- Udfør beregningen af pixelværdierne de fire midterste pixels.
- Overvej hvad der vil ske hvis vægtmatricen ikke summere til 1.

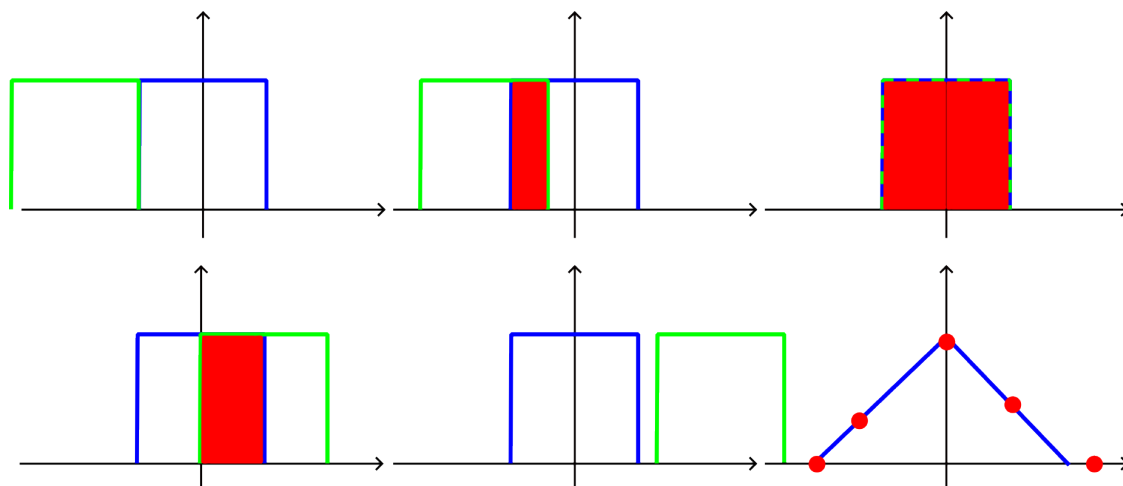
Foldning af kontinuerte funktioner

Det er også muligt at lave foldning med kontinuerte funktioner. Vi vil i det næste se på egenskaberne af foldning og igen kigge på anvendelse af foldning for kontinuerte funktioner. Det er ikke nødvendigt at have gennemgået den diskrete foldning for at forstå dette afsnit.

Foldning giver et mål for hvor meget to funktioner overlapper som funktion af en variabel. Det kan illustreres som fællesmængden af arealerne under kurverne. Det skal bemærkes at denne måde at se foldning på kun er illustrativ og ikke den præcise måde det forholder sig.

Eksempel

Figur xx viser et eksempel hvor foldningen netop svarer til arealet. Her er $f(x) = 1$ i intervallet $[-0.5, 0.5]$, og $g(x) = 1$ i intervallet $[t - 0.5, t + 0.5]$ (og nul ellers). Ved at ændre parameteren t forskyder vi funktionen $g(x)$ i forhold til $f(x)$. Fordelingen svarer til sandsynligheden ved en fair terning hvor hvert slag har samme sandsynlighed. I det diskrete tilfælde skulle man bare summere når de to funktioner overlappede, det samme gør man her. Overlappet mellem de to funktioner ændrer sig afhængigt af t -værdien.



Foldning

Foldningen svarer til det røde areal mellem kurverne. I første koordinatsystem er $t = -1$ og de to funktioner har lige præcist ikke noget overlap. Den grønne figur forskydes nu og overlappet bliver større indtil $t = 0$ hvor de overlapper helt og arealet er én, figur xx. Derefter bliver arealet mindre igen indtil der ikke er noget overlap. Foldningen mellem f og g kan ses nederst til højre som en hat. På akserne har vi nu t langs 1. akse og arealet mellem figurerne op af 2. akse. Foldningen mellem to funktioner giver altså ikke et tal som et bestemt integral, men en ny funktion.

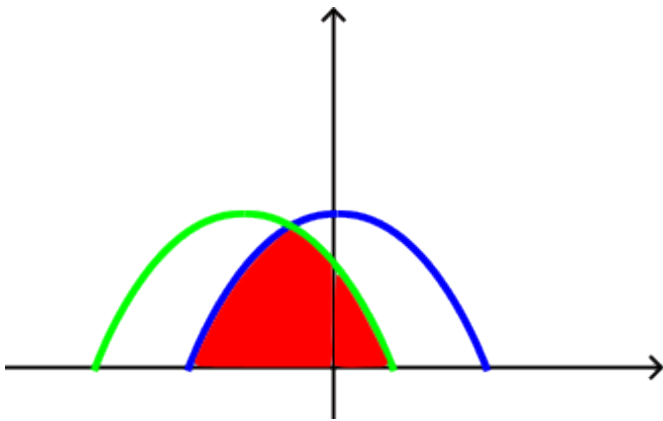
Resultater

- Ved foldning med to funktioner $f(x)$, $g(x)$ dannes der en ny funktion $h(t)$.
- Den nye funktion er en funktion af variabelen t .
- $h(t)$ er nul når $f(x)$ og $g(x)$ ikke har noget overlap.
- $h(t)$ har maksimum når $f(x)$ og $g(x)$ har maksimalt overlap.

Bemærk at fortolkningen af en foldning som arealet under kurverne, kun fælder hvis vi har konstante funktioner. Som i det diskrete tilfælde skal udfaldet vægtes med sandsynligheden, når fordelingerne ikke er konstante.

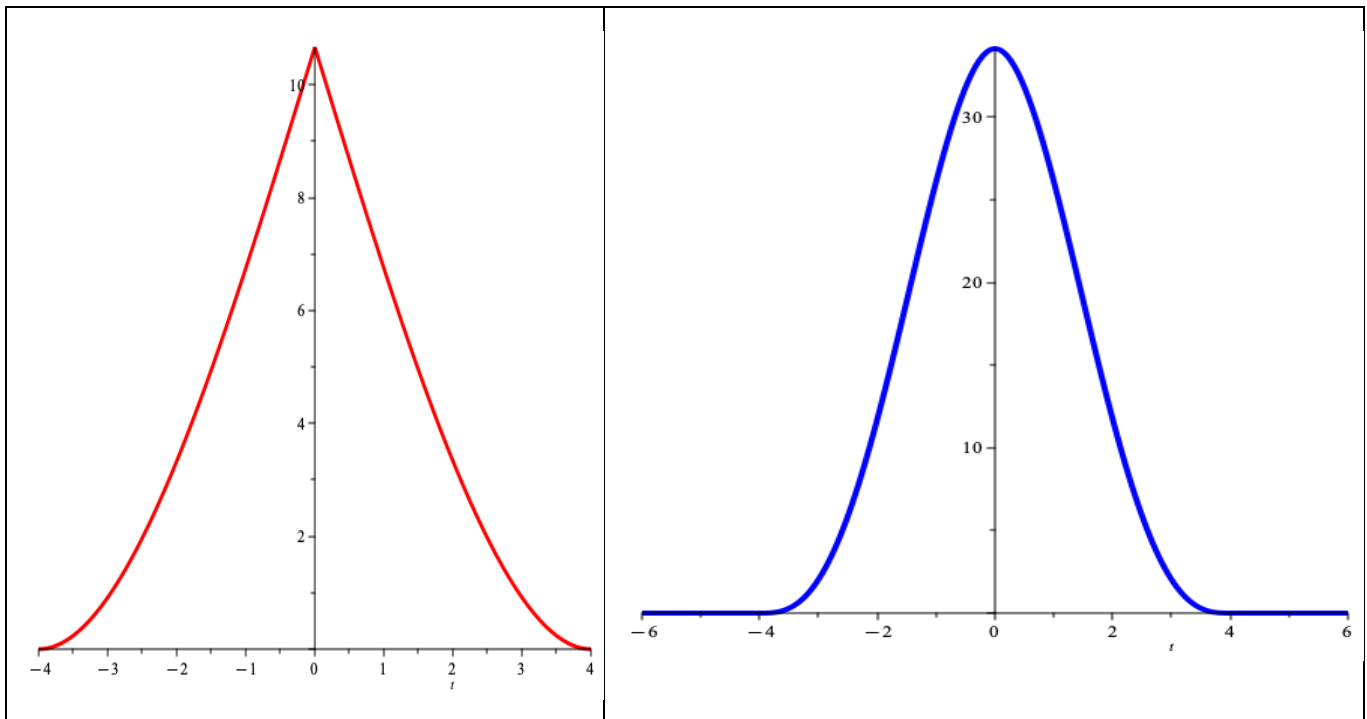
Eksempel

Figuren viser to parabler med forskriften $f(x) = -x^2 + 4$ og $g(t - x) = -(t - x)^2 + 4$, i 1. og 2. kvadrant.



Foldning

Arealet er farvet rødt og når t bliver større forskyder den grønne parabel sig mod højre. I figur xx ses arealet som den røde graf mens foldningen giver den blå.



I dette tilfælde er foldningen altså ikke bare arealet under graferne. Vi har brugt for en definition for at kunne beregne foldningen.

Definition - Kontinuert foldning

Givet to kontinuerte funktioner $f(x)$ og $g(x)$ betegnes foldningen typisk med asterisk-symbolet $*$, altså $f(x) * g(x)$. Dette udregnes med integralet af produktet af de to funktioner,

$$(f * g)(t) = \int_{-\infty}^{\infty} f(x)g(t-x)dx.$$

Lad os kigge på funktionen $g(t-x)$ hvor x er den uafhængige variabel mens t giver forskydningen langs x-aksen. Hvis vi sætter $t = 0$ ses det at funktionen bliver $g(-x)$. Dette svarer til at funktionen g er spejlet i y-aksen.

Øvelse

- Tegn grafen for $g(t-x)$ for forskellige værdier af t og se hvordan funktionen flytter sig.
- Brug definitionen til at lave foldningen mellem f og g ovenfor.

Øvelse

Givet to funktioner $f(x) = \sin(x)$ og $g(x) = 2 \cdot xe^{-x^2}$.

- Undersøg de to grafer ved at tegne graferne $f(x)$, $g(t-x)$ hvor t kan variere.
- Udregn foldningen $(f * g)(t)$ og tegn grafen for foldningen i intervallet $[-4; 4]$.
- Undersøg igen med skyder hvordan graferne $f(x)$, $g(t-x)$ ligger i forhold til hinanden når foldningen giver nul.
- Prøv at forklar hvorfor foldningen giver noget positivt og noget negativt (husk at minus gange minus giver plus).

Matematiske egenskaber ved foldning.

Der gælder samme regler for foldning som for addition, multiplikation og division. Foldning er,

- Distributiv.
- Associativ.
- Kommutativ.

Distributiv egenskab

Den distributive egenskab ved multiplikation siger at $x \cdot (a + b) = x \cdot a + x \cdot b$. For en foldning med funktionerne f, g, h vil det sige at $(f * (g + h))(t) = (f * g + f * h)(t)$.

Vi kan altså vælge at addere funktioner før foldning eller lave foldningen først og så addere.

Associative egenskab

Den associative egenskab giver, at hvad man ganger først ved multiplikation, er ligegyldig da, $a \cdot (b \cdot c) = (a \cdot b) \cdot c$. Ved foldning giver det at $(f * (g * h))(t) = ((f * g) * h)(t)$

Det viser, at rækkefølgen af foldning ikke betyder noget.

Kommutative egenskab

Den kommutative egenskab gør at rækkefølgen ved multiplikation er ligegyldig da, $a \cdot b = b \cdot a$. For en foldning gælder at $(f * g)(t) = (g * f)(t)$ hvilket skrevet med integraler giver $\int_{-\infty}^{\infty} f(x)g(t-x)dx = \int_{-\infty}^{\infty} f(t-x)g(x)dx$.

Der er altså ikke noget specielt ved den funktion vi vælger at forskyde langs x-aksen.

Øvelse

- Vis at foldning er distributiv og associativ ved brug af integralregnereglerne.
- Vis at foldning er kommutativ ved at lave substitution, $x = t - \tilde{x}$. Overvej om det ændre integrationsgrænserne.

Andre definitioner

Foldningens mange anvendelsesmuligheder gør, at der er mange forskellige definitioner af en foldning. Vi skal vise at de forskellige definitioner kan udledes ud fra vores definition, med valg af funktionen $g(x)$.

Ved signalbehandling bruger man ofte definitionen

$$(f * g)(t) = \int_0^{\infty} f(x)g(t-x)dx$$

hvor $t = 0$ er der hvor signalet starter.

Øvelse

Vis, ved at opdele integralet, at funktionen

$$g(x) = \begin{cases} 0 & x < 0 \\ h(x) & x \geq 0 \end{cases}$$

medfører at

$$\int_0^{\infty} f(x)h(x)dx = \int_{-\infty}^{\infty} f(x)g(t-x)dx$$

Vi kan altså, ved at indføre stykvise funktioner ændre integrationsgrænser. Dette kan være nyttigt hvis integralet for eksempel ikke konvergerer.

Anvendelse af foldning

Gennemsnit

Gennemsnit over et interval a kan laves med følgende funktion,

$$g(t-x) = \begin{cases} \frac{1}{a} & 0 < t \leq a \\ 0 & \text{ellers.} \end{cases}$$

Lad os udnytte at vi kan ændre integrationsgrænser til at omskrive udtrykket

$$(f * g)(t) = \int_{-\infty}^{\infty} f(x)g(x-t)dx = \frac{1}{a} \int_0^a f(x)dx.$$

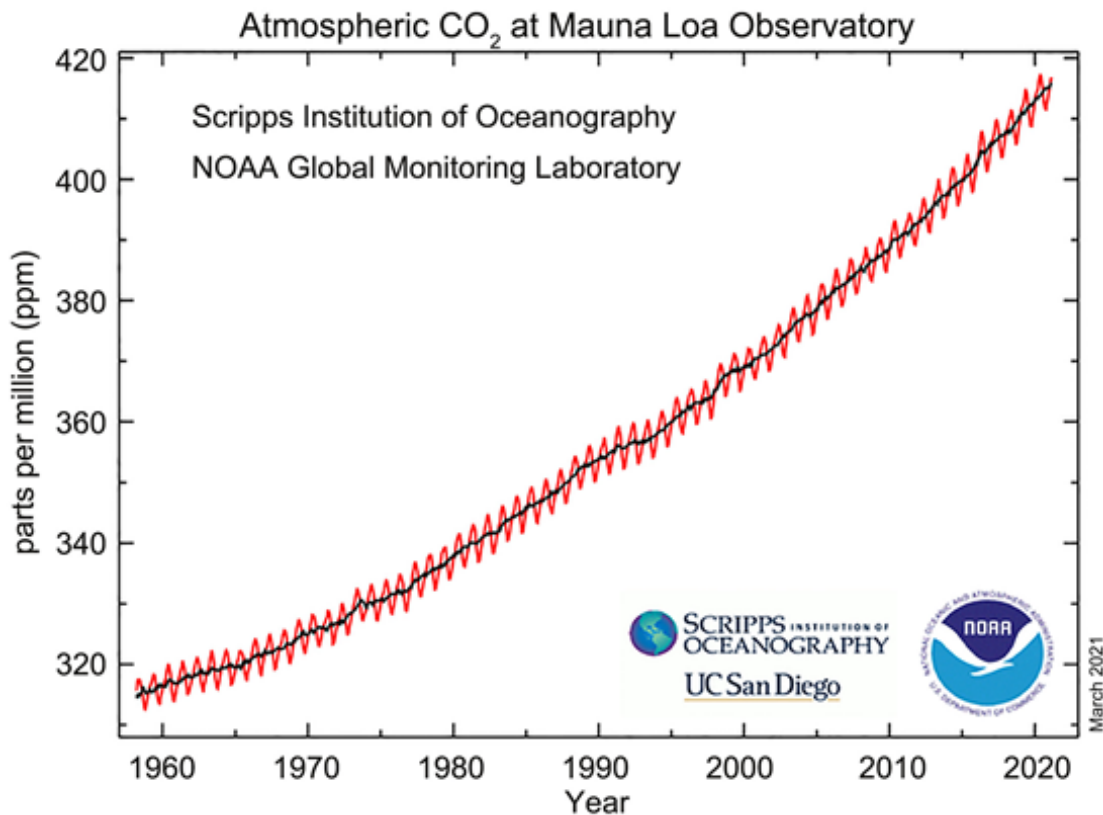
Øvelse

- Beregn gennemsnittet af funktionen $f(x) = \sin(x)$ over de tre intervaller $[0; \pi]$, $[\pi, 2\pi]$, $[0, 2\pi]$.
- Indtegn funktionen og de tre gennemsnit i samme koordinatsystem og tjek beregningerne. Som det ses behøver integralet ikke starte i 0, men kan som her starte i π . Gennemsnittet over intervallet a til b kan skrives som

$$\overline{f(x)} = \frac{1}{b-a} \int_a^b f(x)dx$$

Glidene gennemsnit

Glidende gennemsnit er gennemsnit over et interval $M = b - a$ som forskydes langs x-aksen. Det bruges ofte i databehandling for at fjerne støj. Ved at tage gennemsnittet bliver den hurtigt skiftende støj fjernet. Det kan også være brugbart ved tidsserier hvor man eks. vil fjerne sæsonvariation. Det ses i figur xx hvor mængden af CO₂ i atmosfæren svinger med årstiden på grund af løvfald på den nordlige halvkugle. Ved et glidende gennemsnit over et år, laves den sorte kurve som viser den voksende mængde CO₂ i atmosfæren.



CO₂, kilde: NOAA (2021-b). Sixty Years of Global CO₂ Levels, 1958 to 2018.

Funktionen som giver det glidende gennemsnit er

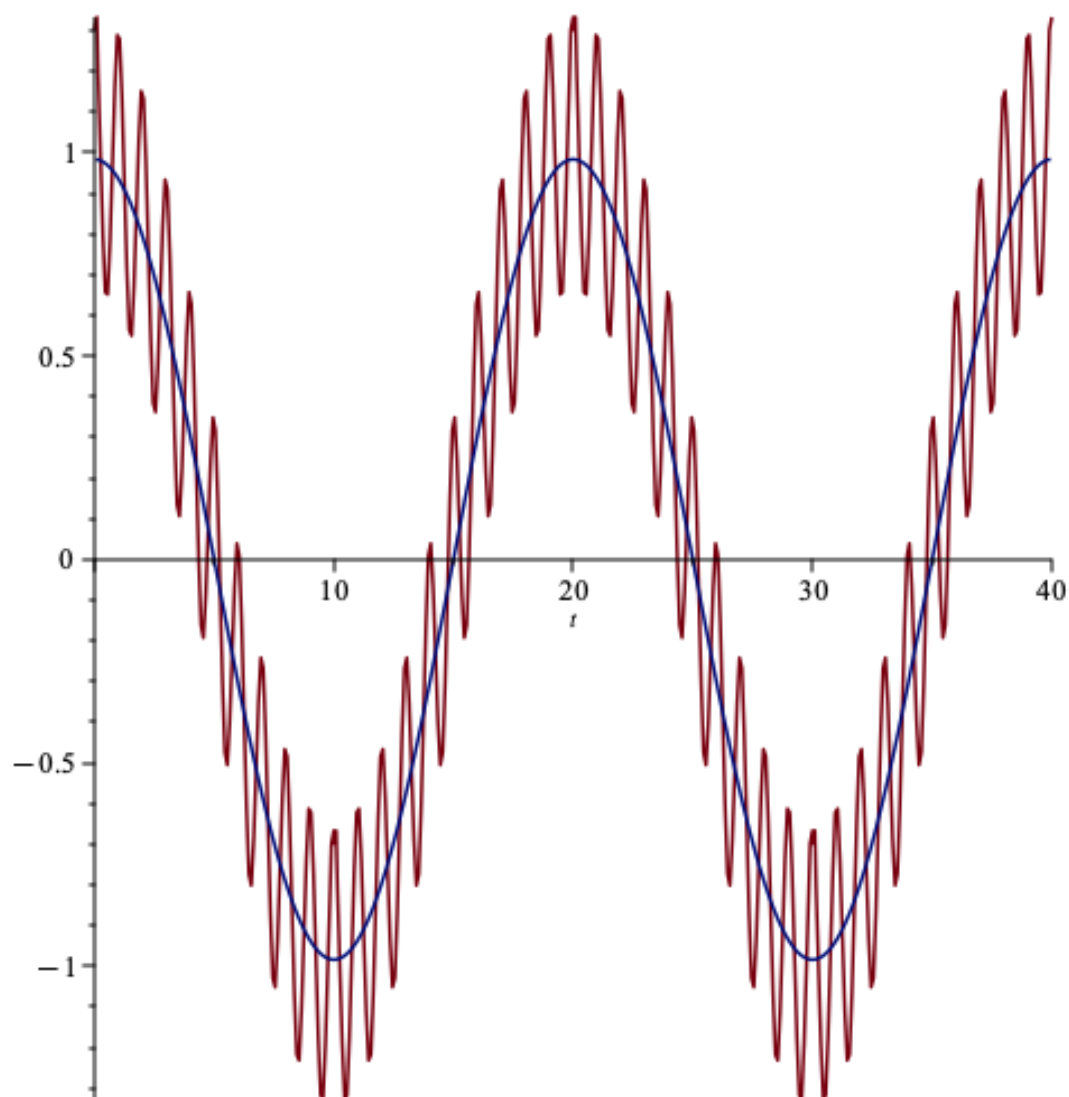
$$g(t-x) = \begin{cases} \frac{1}{M} & , -\frac{1}{2}M < x \leq \frac{1}{2}M \\ 0 & , \text{ellers.} \end{cases}$$

Foldningen bliver

$$(f * g)(t) = \int_{-\infty}^{\infty} f(x)g(x-t)dx = \frac{1}{M} \int_{t-\frac{1}{2}M}^{t+\frac{1}{2}M} f(x)dx.$$

Eksempel

Funktionen $f(x) = \cos\left(\frac{\pi}{10}x\right) + \frac{1}{3}\cos(2\pi x)$ indeholder to cosinusfunktioner med en periode på $T = 0$ og $T = 1$ og en amplitude på $A = 1$ og $A = \frac{1}{3}$. Den langsomme bevægelse kan findes ved et glidende gennemsnit, hvilket ses i figur xx som den blå graf.



Gliden gennemsnit

Den blå graf kommer af et glidende gennemsnit med $M = 2$. Integralet kan beregnes til,

$$h(t) = (f * g)(t) = \frac{1}{2} \int_{t-1}^{t+1} \cos\left(\frac{2\pi}{10}x\right) + \frac{1}{3} \cos(2\pi x) dx = \frac{10 \cos\left(\frac{\pi t}{10}\right) \sin\left(\frac{\pi}{10}\right)}{\pi}.$$

Det noget vanskelige udtryk kan omskrives til

$$h(t) = k \cdot \cos\left(\frac{\pi}{10} \cdot t\right), \text{ hvor } k = \frac{\pi}{10} \sin\left(\frac{\pi}{10}\right) \approx 0.98.$$

$h(t)$ giver altså den langsomme del af svingningen. Perioden forbliver den samme, men amplituden bliver lidt mindre efter foldningen.

Øvelse

- Undersøg hvordan foldningen ændrer sig når man ændrer konstanten M ved at tegne grafen ved forskellige værdier, $M = [0.01, 0.1, 1.0, 4.0, 8.0, 16.0]$.
- Hvorfor bliver amplituden på den langsomme svingning mindre når M bliver større?
- Hvor lille kan M være før den hurtige svingning forsvinder?

Elektronisk kan man fjerne højfrekvent støj med et såkaldt Low Pass filter, hvor de lave frekvenser kommer igenne. Et Low Pass filter gør præcis det sammen som foldningen ovenover.

High pass filter

Hvis man i stedet vil have de hurtige svingninger, men fjerne de langsomme kan det gøres ved at trække det glidende gennemsnit fra det oprindelige signal. Hvis $f(t)$ er det oprindelige signal og $h(t)$ det glidende gennemsnit er high pass filtret $f(t) - h(t)$.

Hvis man har en højdemåler med ud og vandre bruger den trykforskellen når man bevæger sig op til at beregne højdeforskellen. Trykket bliver lavere når man bevæger sig op men trykket kan også skifte i forhold til vejret med høj- og lavtryk. Hvis ændringen i vejret er langsom i forhold til hvor hurtigt man bevæger sig op vil det kunne udlignes ved at beregne $f(t) - h(t)$. Hvis det gøres elektronisk kaldes det et High Pass Filter (HPF), fordi den højfrekvente information bliver bevaret.

Øvelse

- Plot $f(t) - h(t)$ og se om det passer med et HPF.

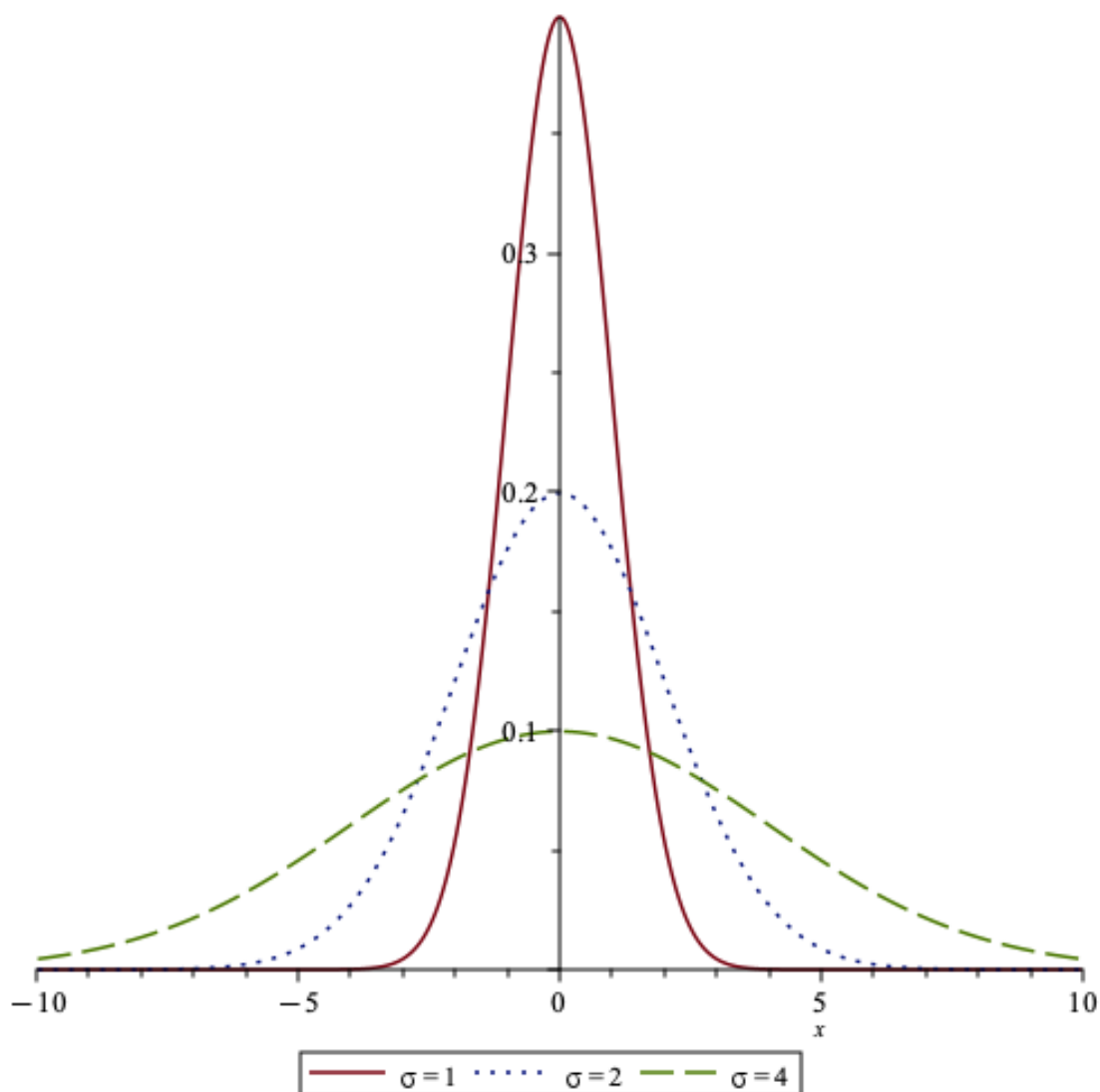
Billedbehandling og gauss-funktionen.

Gauss-funktionen har mange anvendelsesmuligheder, også til billedbehandling. Vi vil i dette se på gaussisk-blur.

Den generelle gauss-funktion er,

$$f(x) = \frac{1}{\sqrt{2\pi}\sigma} e^{-\frac{1}{2}\left(\frac{x-\mu}{\sigma}\right)^2},$$

hvor spredningen σ giver bredden og μ middelværdien .

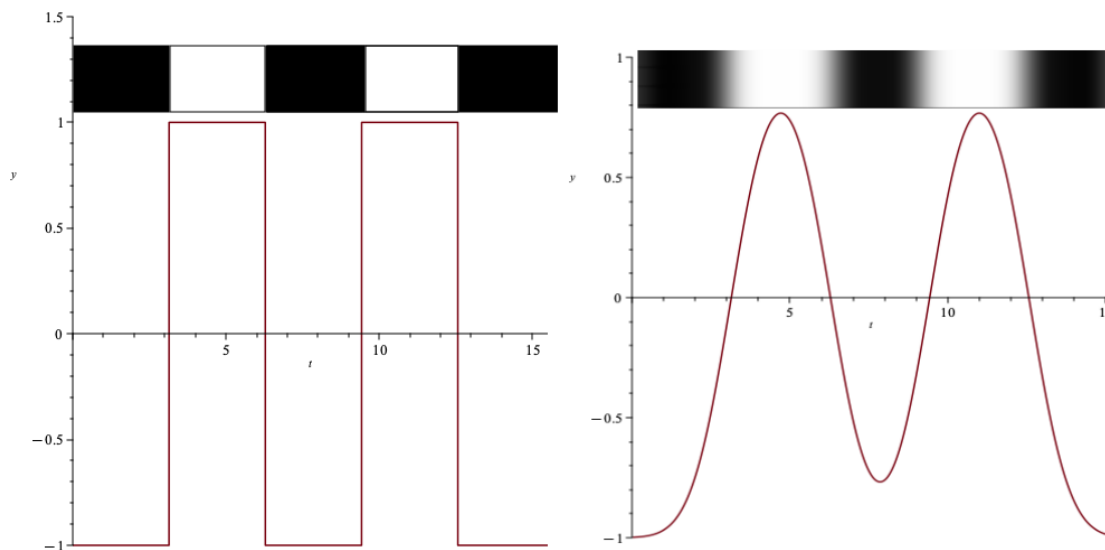


Gauss-funktioner

Figur xx viser tre gauss-funktioner centreret om $\mu = 0$ med forekkelig spredning. Alle tre funktioner har samme areal under kurven, 1. Ved sløring virker gauss-funktionerne som det glidende gennemsnit, men her er der mest vægt på den midsterste observation og mindre på de omkringliggende. Ved at gøre spredningen større bliver der taget et gennemsnit af flere pixels og billedet bliver mere sløret.

Eksempel

Figur 4 viser et sort-hvidt billede og en repræsentation med en firkantgraf, hvor høj værdi, 1, giver hvid og lav værdi, -1 , giver sort.



Sort hvid skarpt, sort hvid sløret

Blur filter

Vi laver en foldning mellem firkant-funktionen og gauss-funktionen,

$$f(x) = \begin{cases} -1 & x < \pi \\ 1 & \pi \leq x < 2\pi. \\ \dots \end{cases}$$

Ved foldningen trækkes konstanterne udenfor integralet,

$$(f * g)(t) = \frac{1}{\sqrt{2\pi}\sigma} \int_{-\infty}^{\infty} f(x) e^{-\frac{1}{2}\left(\frac{t-x}{\sigma}\right)^2} dx.$$

Dette integral løses med computer og giver den røde graf i figur xx.

Den røde graf i figur 5 er nu blevet “blødere” end firkantgrafen i figur 4. Det ses på billedet ovenover, hvor de to farver nu fader over i hinanden.

Ved at ændre på σ kan man lave sit billede mere eller mindre sløret.

Øvelse

- Lav en simpel firkant-funktion og prøv at lav en foldning med en gaussisk funktion.
- Undersøg hvordan man kan gøre den oprindelige funktion mere blød ved at ændre på standardafvigelsen.
- Prøv med et 2. grads polynomium i stedet for gauss-funktionen.

Der er en online udgave af GIMP, <https://fixthephoto.com/online-gimp-editor.html>

- Afprøv gaussian blur filtret på et selvvalgt billed.

Ved billedbehandling er det mere relevant at bruge funktioner af to variable, så man kan have pixels angivet ved (x, y) . Den tilhørende foldning bliver

$$(f * g)(t, s) = \frac{1}{2\pi\sigma_x\sigma_y} \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} f(x, y) e^{-\frac{1}{2}\left(\frac{t-x}{\sigma_x}\right)^2} e^{-\frac{1}{2}\left(\frac{s-y}{\sigma_y}\right)^2} dx dy.$$

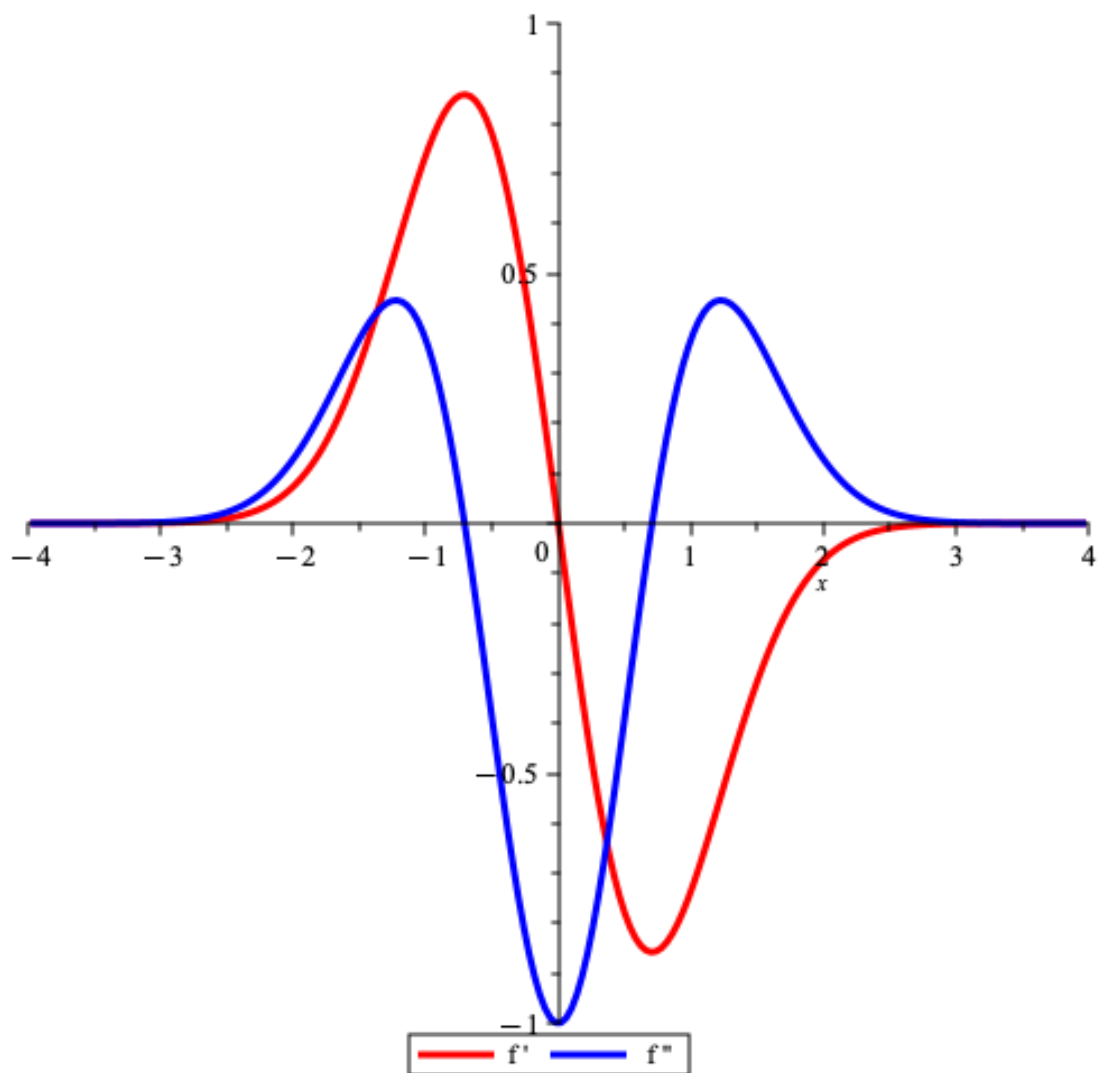
I praksis bruges den diskrete foldning som vi arbejdede med i afsnittet ovenfor, så vi vil ikke gøre mere ud af den kontinuerte her.

Andre filtre

Gauss-funktionen kan bruges til andet end en blur filter. Her bruger man den afledte af gauss-funktionen hvor,

- $\frac{df}{dx}$ kan bruges til kantdetektion.
- $\frac{d^2f}{dx^2}$ kan bruges til at førstærke kontrasten.

De to funktionen kan ses i figur xx



afledte gauss-funktioner

Kontrast og kantdetektion bruges blandt andet i sundhedsvæsnet til at detekterer sygdomme på billeder.