

Lineær algebra og numeriske metoder. Pythonversjon

Øyvind Ryan
Matematisk institutt
Universitetet i Oslo

31. oktober 2025

Forord

Disse notatene ble til i forbindelse med etableringen av det nye førstesemesterkurset MAT1105 - Lineær algebra og numeriske metoder - ved Universitetet i Oslo. Kurset ble holdt første gang høsten 2024, og er et introduksjonskurs til grunnleggende matematisk tenkning, lineær algebra og numeriske metoder. I første versjon av kurset ble lineær algebra-delen dekket av boka 'Flervariabel analyse med lineær algebra' av Tom Lindstrøm (kap.1 + kap. 4 der), mens de to andre delene av pensum ble dekket av egne kompendier. I 2025-versjonen har vi også laget eget stoff for lineær algebra-delen. Dette kompendiet har også numerisk analyse-delen som en siste del, selv om denne delen bare bygger løst på lineær algebra. Grunnleggende matematisk tenkning har eget kompendium.

Kompendiet inneholder mange bevis, men de fleste bevis i lineær algebra er ganske korte. Flere av oppgavene i boka består av tidligere eksamensoppgaver.

Kompendiet inneholder en del programmeringseksempler i Matlab, med en innføring i grunnleggende Matlabkommandoer. En versjon av kompendiet som bruker Python vil også bli skrevet. Dere velger selv om dere programmerer i Matlab eller Python, men det anbefales at dere bruker tid på begge deler.

Blindern, August 2025

Øyvind Ryan

Innhold

I	Lineær algebra	1
1	Vektorer	3
1.1	Lage vektorer	3
1.2	Operasjoner på vektorer	5
1.3	Lengde, skalarprodukt, og projeksjon	10
2	Matriser og matrisemultiplikasjon	19
2.1	Vektoriserte operasjoner på matriser	22
2.2	Matrisemultiplikasjon	25
2.3	Lineære avbildninger og matrisemultiplikasjon	30
2.4	Plotting av polynomer	32
2.5	Blokkmatriser	35
3	Gausseliminasjon og redusert trappeform	39
3.1	Gausseliminasjon og kompleksitet	48
3.2	Simultan løsning av likningssystemer	49
4	Den inverse matrisen	55
5	Lineær uavhengighet og basiser	63
5.1	Underrom	67
6	Determinanter	75
6.1	Definisjon av determinanten	76
6.2	Egenskaper til determinanten	78
6.3	Forstørrelsesfaktorer for matriser(*)	84
6.4	Geometrisk tolkning av determinanten. Vektorproduktet	88

7 Egenverdier og egenvektorer	93
8 Diagonalisering. Systemer av differensiallikninger	103
8.1 Diagonalisering og systemer av førsteordens likninger	105
8.2 Høyereordens lineære differensiallikninger	110
9 Ortogonalt diagonaliserbare matriser. Kondisjonstall	115
9.1 Ortogonalt diagonaliserbare matriser	116
9.2 Relativ og absolutt feil	119
9.3 Definisjon av kondisjonstall	120
 II Numerisk analyse	 125
10 Interpolasjon med polynomer	127
10.1 Interpolasjon på kanonisk form	128
10.2 Interpolasjon med Newtonformen	132
10.3 Interpolasjonsfeilen	137
11 Numerisk derivasjon	145
11.1 Newtonkvotienten	146
11.2 Alternativer til newtonkvotienten	149
12 Numerisk integrasjon	161
12.1 Hva er integralet?	161
12.1.1 Definisjonen av integralet	162
12.2 Midtpunktsmetoden	164
12.2.1 Feilestimat for midtpunktsmetoden	166
12.2.2 Estimering av steglengden	169
12.3 Alternativer til midtpunktsmetoden	169
12.3.1 Trapesmetoden ($k = 1$)	171
12.3.2 $k = 2$: Simpsons metode	173
 Solutions	 181

Figurer

1.1	Operasjoner på vektorer.	6
1.2	Ganging av en vektor med en skalar.	7
1.3	Et plan gitt ved en normalvektor og ett punkt.	11
1.4	Projeksjon av vektoren a ned på vektoren b i planet.	15
1.5	Parallelogrammet utspent av vektorene a og b	18
2.1	Plott av polynomene $p_1(x) = x$, $p_2(x) = x^2$, og $p_3(x) = x^3$	33
6.1	Et parallelepiped i rommet.	86
8.1	Plott av ni løsninger for systemet med matrise $A = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}$	107
8.2	Løsningene på systemene i Eksempel 8.8.	110
8.3	Plott av $x_1(t)$ for systemet med matrise A_1 . Øvre og nedre grense fra (8.8) er også vist.	111
9.1	Kondisjonstall for en $n \times n$ -hilbertmatrise, for forskjellige verdier av n	123
10.1	De interpolerende polynomene p_2 , p_3 , og p_4 , for $f(x) = \cos(2\pi x)$. f er også vist.	131
10.2	Tre interpolasjonspunkter og det tilsvarende kvadratiske interpolasjonspolynomet.	132
10.3	Interpolasjon av e^x i to punkter med en sekant (a), og i tre punkter med en parabel (b).	134
10.4	Interpolasjon av $f(x) = \sin(x)$ med det interpolerende polynomet p_n for forskjellige n	136

10.5 Funksjonen $f(x) = \sqrt{x}$ (heltrukken linje) og den kubiske interpolanten i de fire punktene 0, 1, 2, og 3 (stiplet linje).	136
10.6 Interpolasjon av funksjonen $f(x) = 1/(1 + x^2)$ på intervallet $[-5, 5]$ med polynomer av grad n . Punktene er uniformt fordelt i intervallet i hvert tilfelle.	137
11.1 Sekanten (rød kuve) til en funksjon (svart kurve) i punktene a og $a + h$, samt tangenten i a (blå kurve).	147
11.2 Trunkeringsfeilen ved bruk av newtonkvotienten for tilnærming til den deriverte til $f(x) = \sin x$ i punktet $x = 0.5$, for $h \in [10^{-6}, 10^0]$. Plottet er et logaritmisk plott.	150
11.3 Sekanten i a (rød kurve) til en funksjon f (svart kurve) basert i $a - h$ og $a + h$, samt tangenten i a (blå kurve). Parabelen som interpolerer f i $a - h$, a , og $a + h$ er også vist (svart kurve)	151
11.4 loglog-plott av estimatene på trunkeringsfeilen for de fire metodene vi ser på, for $f(x) = \sin x$ i $x = 0.5$, og for $h \in [10^{-6}, 10^0]$	154
12.1 Arealet under grafen til en funksjon.	162
12.2 Innskrevne (til venstre) og omskrevne (til høyre) stegfunksjoner for en grov partisjon (øverst) og en fin partisjon (nederst).	163
12.3 Midtpunktsmetoden med ett delintervall (a), og fem delintervaller (b).	165
12.4 Trapesmetoden med ett delintervall (a) og fem delintervaller (b).	172
12.5 Simpsons metode med ett delintervall (a) og tre delintervaller (b).	173

Listings

python/gaussel.py	52
python/qract.py	74
python/detdef.py	77
python/plotdiff.py	108
python/plotcond.py	122
10.1 Kode for å plotte det interpolerende polynomet	129
10.2 Plotting av interpolasjonspolynomet på Newtonform	142
11.1 Sammenligning av trunkeringsfeil	153
approximation/plotintpolcos.py	208
differentiation/plotgrafsekanter.py	215
1 Utregning av øvre/nedre-integralet	215
integration/integration.py	217
integration/integration.py	218
integration/integration.py	219

Del I

Lineær algebra

KAPITTEL 1

Vektorer

En *vektor* er en samling av tall listet opp etter hverandre. Tallene som listes opp kalles *komponentene* i vektoren. Vi skal representere vektorer med små bokstaver i fet skrift (slik som **a**, **b**, e.t.c.), og komponentene med små bokstaver (slik som $a_1, \dots, a_n$). Alternativt skal vi også skrive $(a_1, a_2, \dots, a_n)$ for vektoren **a**. I starten skal vi anta at vektorene er reelle, det vil si alle komponentene er reelle ($a_i \in \mathbb{R}$ for alle i). Dette skal vi også skrive ved at $\mathbf{a} \in \mathbb{R}^n$, der n er antall komponenter i vektoren. Senere skal vi også skrive $\mathbf{a} \in \mathbb{C}^n$ hvis komponentene i **a** mer generelt er komplekse tall.

1.1 Lage vektorer

I det følgende trenger vi pakken numpy i Python:

```
import numpy as np
```

Når vi definerer en vektor bruker vi firkantklammer til å markere start og slutt på vektoren, og komma til å adskille komponentene. Under ser du et eksempel på dette, der >> angir promptet.

```
>> a = np.array([2,4,6,8,10,12,14,16])
>> a
array([ 2,  4,  6,  8, 10, 12, 14, 16])
```

En vektor kan defineres på mange forskjellige måter:

```
>> np.linspace(0,1,6)
array([0. , 0.2, 0.4, 0.6, 0.8, 1. ])
```

Mer generelt gir `linspace(a, b, n)` vektoren $\left\{a + \frac{k(b-a)}{n-1}\right\}_{k=0}^{n-1}$, det vil si vektoren med n komponenter i stigende rekkefølge fra intervallet $[a, b]$, der endepunktene a og b er med, og der det er lik avstand mellom komponentene. Vi kan også lage nye vektorer på følgende måte:

```
>> np.arange(1,7)
array([1, 2, 3, 4, 5, 6])
>> np.arange(1,7,2)
array([1, 3, 5])
```

Mer generelt har vi at

- $a:b$ gir vektoren $(a, a+1, a+2, \dots)$, og
- $a:h:b$ gir vektoren $(a, a+h, a+2h, \dots)$,

der bare tall $\leq b$ listes opp (h indikerer altså en steglengde). Legg spesielt merke til at $1:2:6$ ikke inneholder det høyre endepunktet 6. Kolon brukes med andre ord til å lage vektorer, noe vi skal komme tilbake til.

Kommandoene over er nyttige for å lage et sett med indekser inn i en vektor. Hvis $\mathbf{a} = (a_1, a_2, \dots, a_n)$, og komponentene i vektoren $\mathbf{I}$ er positive heltall $\leq n$, så vil $\mathbf{a}(\mathbf{I})$ returnere en vektor med de tilhørende verdiene:

```
>> a[0:6]
array([ 2,  4,  6,  8, 10, 12])
>> a[0:6:2]
array([ 2,  6, 10])
```

Vi har også en spesiell notasjon for å angi siste del av en vektor:

```
>> a[3:]
array([ 8, 10, 12, 14, 16])
```

$3:$ betyr her "alle indekser ≥ 3 ". Man trenger altså ikke vite hvor lang vektoren er, for å få ut siste del av den!

Eksempel 1.1 (max, min, sum, og prod). Skal man finne maksimum, minimum, summen, eller produktet av alle komponentene i en vektor, så kan man skrive

```
>> np.max(a)
16
>> np.min(a)
2
>> np.sum(a)
```

```
72
>> np.prod(a)
10321920
```

1.2 Operasjoner på vektorer

Addisjon og subtraksjon for vektorer defineres *komponentvis*. Dette betyr at, hvis **a** og **b** begge er vektorer med n komponenter, så er

- $\mathbf{a} + \mathbf{b} = (a_1 + b_1, a_2 + b_2, \dots, a_n + b_n),$
- $\mathbf{a} - \mathbf{b} = (a_1 - b_1, a_2 - b_2, \dots, a_n - b_n).$
- $t\mathbf{a} = (ta_1, ta_2, \dots, ta_n)$ når t er en skalar.

Vektoraddisjon krever altså to vektorer med like mange komponenter, og returnerer en vektor med samme antall komponenter.

Det er vanlig å tegne vektorer som piler som starter i origo, og ender i punktet angitt av vektoren. Men en vektor kan også representeres med en pil som starter i et annet punkt - det er selve retningen vektoren peker i som angir dens komponenter. Figur 1.1 illustrerer addisjon og subtraksjon av vektorer. Å regne ut $\mathbf{a} + \mathbf{b}$ svarer til å plassere vektoren **b** på slutten av vektoren **a**, og lese av koordinatene du da får. $\mathbf{a} - \mathbf{b}$ kan visualiseres på to forskjellige måter: Enten ved å plassere vektoren $-\mathbf{b}$ på slutten av vektoren **a**, eller ved å tegne pilen som begynner i **b** og slutter i **a** ((b) i figuren). Figur 1.2 illustrerer multiplikasjon av en vektor med en skalar.

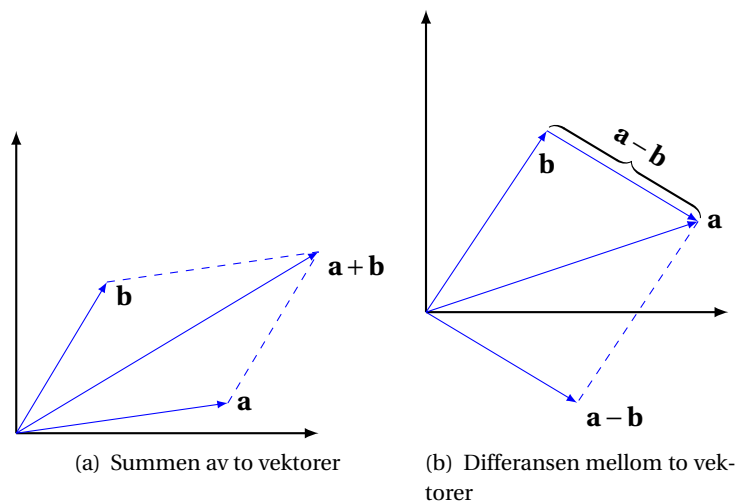
Nullvektoren, skrevet **0**, er vektoren der alle komponenter er 0, det vil si at $\mathbf{0} = (0, 0, \dots, 0)$. Det er klart at nullvektoren oppfyller at $\mathbf{a} + \mathbf{0} = \mathbf{a}$, og at $0\mathbf{a} = \mathbf{0}$, for alle **a**.

Eksempel 1.2. La $\mathbf{a} = (1, 2, 3)$, $\mathbf{b} = (4, 5, 6)$, og $t = 2$. Da er

$$\begin{aligned}\mathbf{a} + \mathbf{b} &= (1, 2, 3) + (4, 5, 6) = (1 + 4, 2 + 5, 3 + 6) = (5, 7, 9) \\ \mathbf{a} - \mathbf{b} &= (1, 2, 3) - (4, 5, 6) = (1 - 4, 2 - 5, 3 - 6) = (-3, -3, -3) \\ t\mathbf{a} &= 2(1, 2, 3) = (2 \cdot 1, 2 \cdot 2, 2 \cdot 3) = (2, 4, 6).\end{aligned}$$

Dette kan vi sjekke som følger:

```
>> a = np.array([1,2,3])
>> b = np.array([4,5,6])
>> t = 2
>> a + b
```



Figur 1.1. Operasjoner på vektorer.

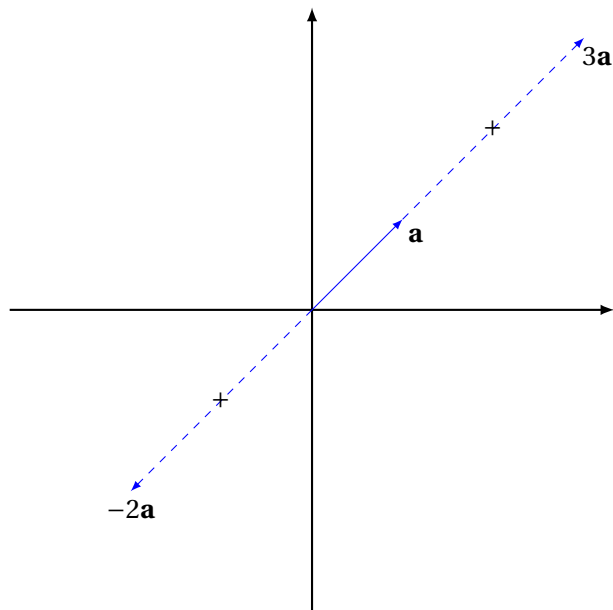
```
array([5, 7, 9])
>> a-b
array([-3, -3, -3])
>> t*a
array([2, 4, 6])
```

Legg merke til at vi brukte semikolon for å avslutte første linje. Dette medfører at ingenting blir skrevet ut på slutten av denne linja. Legg også merke til forskjellen mellom å skrive $a+b$, og $\text{sum}(a)$. Den første regner ut mange summer av to komponenter. Den siste regner ut én sum av mange komponenter.

Ofta skal man anvende en operasjon på alle komponenter i en vektor. Man kan da skrive en løkke som gjør dette. For at man skal unngå å skrive løkker i massevis støtter de fleste programmeringsspråk det som kalles *vektorisering* - det vil si at operasjoner som bare er definert på $\mathbb{R}$ eller $\mathbb{C}$, automatisk blir definert for vektorer ved å gjøre operasjonen på hver komponent. Fra definisjonen av addisjon og subtraksjon på vektorer følger det at disse er vektoriserte operasjoner.

Definisjon 1.3. Anta f er en funksjon på $\mathbb{R}$ eller $\mathbb{C}$, og at $\mathbf{a} = (a_1, a_2, \dots, a_n)$. Vektoriseringen av f er definert ved

$$f(\mathbf{a}) = (f(a_1), f(a_2), \dots, f(a_n)).$$



Figur 1.2. Ganging av en vektor med en skalar.

Vektoriseringen av binære funksjoner (funksjoner som tar to variable) er definert tilsvarende: Som et eksempel er vektoriseringen av addisjon og potens gitt ved

$$\mathbf{a} + \mathbf{b} = (a_1 + b_1, a_2 + b_2, \dots, a_n + b_n)$$

$$\mathbf{a}^{\mathbf{b}} := (a_1^{b_1}, a_2^{b_2}, \dots, a_n^{b_n}),$$

og vektoriseringen av multiplikasjon er gitt ved $(a_1 b_1, a_2 b_2, \dots, a_n b_n)$. Her er både $\mathbf{a}$ og $\mathbf{b}$ vektorer med n komponenter.

Addisjon av en vektor og en skalar er også tillatt i mange språk, selv om det matematisk ikke er definert. Da regnes ut

$$\mathbf{a} + t = (a_1 + t, a_2 + t, \dots, a_n + t).$$

```
>> a+t
array([3, 4, 5])
```

Eksempel 1.4 (Vektoriserte funksjoner). De fleste funksjoner er vektorisert:

```
>> np.sqrt(np.array([1,4,9,16]))
array([1., 2., 3., 4.])
>> np.abs(np.array([-2,-1,1,2]))
array([2, 1, 1, 2])
```

Dette gjelder også trigonometriske funksjoner:

```
>> np.sin(np.array([0,np.pi/2,np.pi,3*np.pi/2,2*np.pi]))
array([ 0.0000000e+00,  1.0000000e+00,  1.2246468e-16, -1.0000000e+00,
        -2.4492936e-16])
>> np.cos(np.array([0,np.pi/2,np.pi,3*np.pi/2,2*np.pi]))
array([ 1.0000000e+00,  6.1232340e-17, -1.0000000e+00, -1.8369702e-16,
        1.0000000e+00])
```

Koden under bekrefter at logaritme og eksponentialfunksjon er omvendte funksjoner:

```
>> np.exp(np.log(np.array([1,2,3])))
array([1., 2., 3.])
>> np.log(np.exp(np.array([0,1,2])))
array([0., 1., 2.])
```

Eksempel 1.5 (Vektoriserte binære funksjoner). max og min kan også brukes på samme måte som + og -, hvis man i stedet sender inn to argumenter:

```
>> np.maximum(a,b)
array([4, 5, 6])
>> np.minimum(a,b)
array([1, 2, 3])
```

Spesielt har vi at

$$\begin{aligned} & \max((a_1, a_2, \dots, a_n), (b_1, b_2, \dots, b_n)) \\ &= (\max(a_1, b_1), \max(a_2, b_2), \dots, \max(a_n, b_n)). \end{aligned}$$

Multiplikasjon og potens ser også vektorisert.

```
>> a*b
array([ 4, 10, 18])
>> a**b
array([ 1, 32, 729])
```

Det finnes mange operasjoner vi kan gjøre på vektorer. Som et eksempel kan de settes sammen etter hverandre:

```
>> np.block([a,b])  
array([1, 2, 3, 4, 5, 6])
```

Siden addisjon av vektorer er vektorisert så arver vektorer regneregler fra de reelle tall:

Proposisjon 1.6. *Vi har at*

1. $\mathbf{a} + \mathbf{b} = \mathbf{b} + \mathbf{a}$
2. $(\mathbf{a} + \mathbf{b}) + \mathbf{c} = \mathbf{a} + (\mathbf{b} + \mathbf{c})$
3. $s(\mathbf{a} + \mathbf{b}) = s\mathbf{a} + s\mathbf{b}$
4. $(s + t)\mathbf{a} = s\mathbf{a} + t\mathbf{a}$,

der $\mathbf{a}$, $\mathbf{b}$, og $\mathbf{c}$ er vektorer, og s og t er skalarer.

I beviset, og flere ganger senere, skal vi bruke at to vektorer $\mathbf{a}$ og $\mathbf{b}$ er like hvis og bare hvis $a_i = b_i$ for alle i .

Bevis. 1.-3. følger rett fra definisjonen av regnereglene for vektorer, samt regnereglene for reelle tall. La oss derfor begrense oss til å vise 4..

- Venstresiden $(s + t)\mathbf{a}$ har komponenter $(s + t)a_i$ (ved definisjonen av skalarvektor multiplikasjon).
- På høyresiden har $s\mathbf{a}$ komponenter sa_i , og $t\mathbf{a}$ har komponenter ta_i . Ved definisjonen av vektoraddisjon har derfor høyresiden $s\mathbf{a} + t\mathbf{a}$ komponenter $sa_i + ta_i$.

Høyresiden og venstresiden er derfor like hvis og bare hvis $(s + t)a_i = sa_i + ta_i$ for alle i . Men dette er en regneregul vi vet holder for alle reelle tall. ■

Eksempel 1.7 (Linjer). La oss finne et uttrykk for linja i $\mathbb{R}^n$ som går gjennom to gitte punkter $\mathbf{x}_1$ og $\mathbf{x}_2$. Denne må ha retningsvektor $\mathbf{x}_2 - \mathbf{x}_1$. Siden linja skal gå gjennom $\mathbf{x}_1$ så blir uttrykket

$$\mathbf{y}(t) = \mathbf{x}_1 + t(\mathbf{x}_2 - \mathbf{x}_1) = (1 - t)\mathbf{x}_1 + t\mathbf{x}_2.$$

Vi ser også at $\mathbf{y}(0) = \mathbf{x}_1$, og $\mathbf{y}(1) = \mathbf{x}_2$.

Vektorene til nå har alle vært reelle, men de kunne like godt vært komplekse. Det komplekse tallet $2 + 3i$ skrives som $2+3\cdot i$. Siden i er forbeholdt den imaginære enheten, så bør du kalle variablene i koden din noe annet.

1.3 Lengde, skalarprodukt, og projeksjon

Vi skal nå definere begrepene lengde og skalarprodukt, og da skal vi først anta at vektorene er reelle. Hvis $\mathbf{a}$ er en reell vektor definerer vi *lengden* til $\mathbf{a}$ som

$$|\mathbf{a}| = \sqrt{a_1^2 + a_2^2 + \cdots + a_n^2}. \quad (1.1)$$

Dette er bare en generalisering av vår intuisjon om lengde fra $\mathbb{R}^2$ og $\mathbb{R}^3$. Spesielt har vi at $|\mathbf{a}| \geq 0$, og $|\mathbf{a}| = 0$ hvis og bare hvis $\mathbf{a} = \mathbf{0}$.

Hvis $\mathbf{a}, \mathbf{b} \in \mathbb{R}^n$ så definerer vi det *reelle skalarproduktet* $\mathbf{a} \cdot \mathbf{b}$ ved

$$\mathbf{a} \cdot \mathbf{b} = a_1 b_1 + a_2 b_2 + \cdots + a_n b_n. \quad (1.2)$$

Vi sier at $\mathbf{a}$ og $\mathbf{b}$ er *ortogonale* hvis $\mathbf{a} \cdot \mathbf{b} = 0$.

Eksempel 1.8. La oss se på vektorene $\mathbf{a} = (1, -4, 9)$ og $\mathbf{b} = (2, 3, -6)$. Lengdene til disse er

$$\begin{aligned} |\mathbf{a}| &= \sqrt{1^2 + (-4)^2 + 9^2} = \sqrt{1 + 16 + 81} = \sqrt{98} = 7\sqrt{2} \\ |\mathbf{b}| &= \sqrt{2^2 + 3^2 + (-6)^2} = \sqrt{4 + 9 + 36} = 7. \end{aligned}$$

Skalarproduktet av de to er

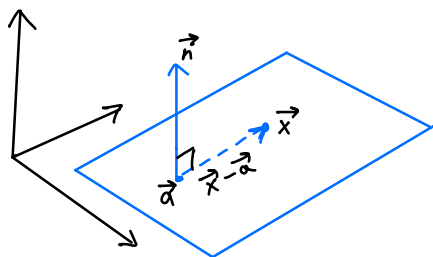
$$\mathbf{a} \cdot \mathbf{b} = (1, -4, 9) \cdot (2, 3, -6) = 1 \cdot 2 + (-4) \cdot 3 + 9 \cdot (-6) = 2 - 12 - 54 = -64.$$

Vi har følgende resultat.

Proposisjon 1.9. For alle vektorer $\mathbf{a}, \mathbf{b}, \mathbf{c} \in \mathbb{R}^n$, og skalarer s , har vi at

1. $\mathbf{a} \cdot \mathbf{a} = |\mathbf{a}|^2$,
2. $\mathbf{a} \cdot \mathbf{b} = \mathbf{b} \cdot \mathbf{a}$,
3. $\mathbf{c} \cdot (\mathbf{a} + \mathbf{b}) = \mathbf{c} \cdot \mathbf{a} + \mathbf{c} \cdot \mathbf{b}$ og $(\mathbf{a} + \mathbf{b}) \cdot \mathbf{c} = \mathbf{a} \cdot \mathbf{c} + \mathbf{b} \cdot \mathbf{c}$,
4. $(s\mathbf{a}) \cdot \mathbf{b} = \mathbf{a} \cdot (s\mathbf{b}) = s(\mathbf{a} \cdot \mathbf{b})$

når $\mathbf{a} \cdot \mathbf{b}$ er definert ved (1.2).



Figur 1.3. Et plan gitt ved en normalvektor og ett punkt.

Bevis. 1.-3. følger greit fra regnereglene for reelle tall, og er gitt som en oppgave. 4. kan vi bevise slik:

$$(sa) \cdot b = (sa_1, \dots, sa_n) \cdot (b_1, \dots, b_n) = sa_1 b_1 + \dots + sa_n b_n$$

$$a \cdot (sb) = (a_1, \dots, a_n) \cdot (sb_1, \dots, sb_n) = sa_1 b_1 + \dots + sa_n b_n$$

$$s(a \cdot b) = s(a_1 b_1 + \dots + a_n b_n) = sa_1 b_1 + \dots + sa_n b_n$$

der vi brukte regnereglene for reelle tall. ■

Omvendt kan man også vise at skalarproduktet er den eneste funksjonen i to variable (**a** og **b**) som oppfyller egenskapene 1.-3 (se Oppgave 11).

Eksempel 1.10 (Plan). La oss også finne et uttrykk for planet som går gjennom et gitt punkt (x_0, y_0, z_0) , og har en gitt normalvektor $\mathbf{n} = (a, b, c)$. (x, y, z) ligger i dette planet hvis og bare hvis $(x, y, z) - (x_0, y_0, z_0)$ er ortogonal på $\mathbf{n}$, se Figur 1.3. Dette er det samme som at

$$\begin{aligned} 0 &= ((x, y, z) - (x_0, y_0, z_0)) \cdot \mathbf{n} \\ &= (x - x_0, y - y_0, z - z_0) \cdot (a, b, c) \\ &= a(x - x_0) + b(y - y_0) + c(z - z_0). \end{aligned}$$

Uttrykket for planet blir dermed

$$ax + by + cz = d,$$

der $d = ax_0 + by_0 + cz_0$.

La oss nå se på hvordan lengde og skalarprodukt kan defineres for komplekse vektorer. For lengde vil ikke definisjonen (1.1) fungere, siden dette gir komplekse tall når **a** er kompleks (vi vil at lengde skal være en reell størrelse). Vi definerer i stedet

$$|\mathbf{a}| = \sqrt{|a_1|^2 + |a_2|^2 + \dots + |a_n|^2}, \quad (1.3)$$

som reduseres til definisjon (1.1) når $\mathbf{a}$ er reell. Under rottegnet her tar vi lengder av komplekse tall, slik at resultatet her blir ≥ 0 , og 0 kun når $\mathbf{a} = \mathbf{0}$. Lengden til en vektor kan regnes ut som følger.

```
>> a=np.array([3,4])
>> np.linalg.norm(a)
5.0
```

Denne kan også regnes ut som følger.

```
>> np.sqrt(np.sum(np.abs(a)**2))
5.0
```

Her kombineres fire operasjoner. Med $\mathbf{a} = (a_1, a_2, \dots, a_n)$ regnes ut i rekkefølge:

1. $(|a_1|, |a_2|, \dots, |a_n|)$ (vektorisert absoluttverdi),
2. $(|a_1|^2, |a_2|^2, \dots, |a_n|^2)$ (vektorisert potens),
3. $|a_1|^2 + |a_2|^2 + \dots + |a_n|^2$,
4. og til slutt $\sqrt{|a_1|^2 + |a_2|^2 + \dots + |a_n|^2}$.

De to siste operasjonene er ikke vektorisert, siden de returnerer skalarer.

Så kommer vi til hvordan vi bør definere det komplekse skalarproduktet. I tråd med det reelle skalarproduktet vil vi ha at

$$\mathbf{a} \cdot \mathbf{a} = |\mathbf{a}|^2 = a_1 \overline{a_1} + \dots + a_n \overline{a_n}.$$

Det er klart at uttrykket (1.2) ikke vil fungere, siden dette ikke fanger opp konjugasjonen her. La oss derfor i stedet definere

$$\mathbf{a} \cdot \mathbf{b} = a_1 \overline{b_1} + \dots + a_n \overline{b_n}, \quad (1.4)$$

siden dette ville stemme overens med uttrykket for lengde når $\mathbf{a} = \mathbf{b}$. Vi har følgende resultat

Proposisjon 1.11. For alle vektorer $\mathbf{a}, \mathbf{b}, \mathbf{c} \in \mathbb{C}^n$, og $s \in \mathbb{C}$, har vi at

1. $\mathbf{a} \cdot \mathbf{a} = |\mathbf{a}|^2$,
2. $\mathbf{a} \cdot \mathbf{b} = \overline{\mathbf{b} \cdot \mathbf{a}}$,
3. $\mathbf{c} \cdot (\mathbf{a} + \mathbf{b}) = \mathbf{c} \cdot \mathbf{a} + \mathbf{c} \cdot \mathbf{b}$ og $(\mathbf{a} + \mathbf{b}) \cdot \mathbf{c} = \mathbf{a} \cdot \mathbf{c} + \mathbf{b} \cdot \mathbf{c}$,
4. $(s\mathbf{a}) \cdot \mathbf{b} = s(\mathbf{a} \cdot \mathbf{b})$ og $\mathbf{a} \cdot (s\mathbf{b}) = \overline{s}(\mathbf{a} \cdot \mathbf{b})$,

når $\mathbf{a} \cdot \mathbf{b}$ er definert ved (1.4).

Bevis. 1. er opplagt. 2. følger fra at $\overline{ab} = \overline{b\overline{a}}$ for alle komplekse tall a og b . 3. følger som før. 4. følger fra at $(sa)\overline{b} = s(\overline{a\overline{b}})$ og $\overline{as\overline{b}} = \overline{s}\overline{a\overline{b}}$ når $a, b, s \in \mathbb{C}$. ■

Eksempel 1.12. La oss se på vektorene $\mathbf{a} = (2i, 1+i, 3)$ og $\mathbf{b} = (1+2i, -1, 2-i)$. Vi regner først ut lengdene til komponentene

$$|2i|^2 = 4 \quad |1+i|^2 = 2 \quad |3|^2 = 9 \quad |1+2i|^2 = 5 \quad |-1|^2 = 1 \quad |2-i|^2 = 5$$

Lengdene til $\mathbf{a}$ og $\mathbf{b}$ blir da

$$\begin{aligned} |\mathbf{a}| &= \sqrt{|2i|^2 + |1+i|^2 + 3^2} = \sqrt{4+2+9} = \sqrt{15} \\ |\mathbf{b}| &= \sqrt{|1+2i|^2 + (-1)^2 + |2-i|^2} = \sqrt{5+1+5} = \sqrt{11} \end{aligned}$$

og skalarproduktet av de to blir

$$\begin{aligned} \mathbf{a} \cdot \mathbf{b} &= (2i, 1+i, 3) \cdot (1+2i, -1, 2-i) \\ &= 2i(1+2i) + (1+i)(-1) + 3(2-i) \\ &= 2i(1-2i) - (1+i) + 3(2-i) \\ &= 2i + 4 - 1 - i + 6 + 3i = 9 + 4i. \end{aligned}$$

Det komplekse skalarproduktet får du ved å skrive np.vdot(a,b), men her er det en liten forskjell:

```
>> a = np.array([2*1j, 1+1j, 3])
>> b = np.array([1+2*1j, -1, 2-1j])
>> np.vdot(a,b)
(9-4j)
```

Her brukes altså en litt annen definisjon enn (1.4). Maskinen regner faktisk ut det komplekse skalarproduktet av $\mathbf{a}$ og $\mathbf{b}$ ved $\overline{a_1}b_1 + \dots + \overline{a_n}b_n$:

```
>> np.sum(np.conj(a) .* b)
(9-4j)
```

Hvis $\mathbf{a}$ og $\mathbf{b}$ er ortogonale så gir en enkel utregning at

$$|\mathbf{a} + \mathbf{b}|^2 = (\mathbf{a} + \mathbf{b}) \cdot (\mathbf{a} + \mathbf{b}) = \mathbf{a} \cdot \mathbf{a} + \mathbf{a} \cdot \mathbf{b} + \mathbf{b} \cdot \mathbf{a} + \mathbf{b} \cdot \mathbf{b} = |\mathbf{a}|^2 + |\mathbf{b}|^2,$$

der vi brukte at $\mathbf{b} \cdot \mathbf{a} = \overline{\mathbf{a} \cdot \mathbf{b}} = \overline{0} = 0$. Det følger at $|\mathbf{a} + \mathbf{b}|^2 = |\mathbf{a}|^2 + |\mathbf{b}|^2$. Setter vi spesielt

$$n = 2 \quad \mathbf{a} = (a, 0) \quad \mathbf{b} = (0, b) \quad \mathbf{c} = \mathbf{a} + \mathbf{b} = (a, b),$$

så sier dette at $c^2 = a^2 + b^2$, der a, b, c er lengdene til $\mathbf{a}, \mathbf{b}, \mathbf{c}$. Siden a og b er katetene, og c hypotenusen i en rettvinklet trekant. så har vi altså vist en generalisering av Pythagoras læresetning til n dimensjoner!

Vi sier at $\mathbf{a} \neq \mathbf{0}$ og $\mathbf{b} \neq \mathbf{0}$ er *parallelle* hvis $\mathbf{a} = s\mathbf{b}$ for en skalar s . Vi skal vise følgende.

Proposisjon 1.13. Anta $\mathbf{b} \neq \mathbf{0}$. $\frac{\mathbf{a} \cdot \mathbf{b}}{|\mathbf{b}|^2} \mathbf{b}$ er det unike punktet på linja gjennom $\mathbf{0}$ og $\mathbf{b}$ som ligger nærmest $\mathbf{a}$. Vi skriver $\text{proj}_{\mathbf{b}}(\mathbf{a})$ for $\frac{\mathbf{a} \cdot \mathbf{b}}{|\mathbf{b}|^2} \mathbf{b}$, og kaller dette for *projeksjonen av $\mathbf{a}$ ned på $\mathbf{b}$* .

Bevis. Vi starter med å observere at $\mathbf{a} - \frac{\mathbf{a} \cdot \mathbf{b}}{|\mathbf{b}|^2} \mathbf{b}$ og $\mathbf{b}$ er ortogonale:

$$\left(\mathbf{a} - \frac{\mathbf{a} \cdot \mathbf{b}}{|\mathbf{b}|^2} \mathbf{b} \right) \cdot \mathbf{b} = \mathbf{a} \cdot \mathbf{b} - \frac{\mathbf{a} \cdot \mathbf{b}}{|\mathbf{b}|^2} \mathbf{b} \cdot \mathbf{b} = \mathbf{a} \cdot \mathbf{b} - \mathbf{a} \cdot \mathbf{b} = 0.$$

Dette er i tråd med vår intuisjon om projeksjon: Nærmest punkt til et plan må vi jo få ved å trekke en linje normalt ned til planet. Dekomposisjonen

$$\mathbf{a} = \left(\mathbf{a} - \frac{\mathbf{a} \cdot \mathbf{b}}{|\mathbf{b}|^2} \mathbf{b} \right) + \frac{\mathbf{a} \cdot \mathbf{b}}{|\mathbf{b}|^2} \mathbf{b} \quad (1.5)$$

splitter altså opp $\mathbf{a}$ i en sum der en vektor er parallell med $\mathbf{b}$, og den andre står ortogonalt på $\mathbf{b}$. For $s \in \mathbb{C}$ har vi

$$|\mathbf{a} - s\mathbf{b}|^2 = \left| \mathbf{a} - \frac{\mathbf{a} \cdot \mathbf{b}}{|\mathbf{b}|^2} \mathbf{b} + \left(\frac{\mathbf{a} \cdot \mathbf{b}}{|\mathbf{b}|^2} \mathbf{b} - s\mathbf{b} \right) \right|^2 = \left| \mathbf{a} - \frac{\mathbf{a} \cdot \mathbf{b}}{|\mathbf{b}|^2} \mathbf{b} \right|^2 + \left| \frac{\mathbf{a} \cdot \mathbf{b}}{|\mathbf{b}|^2} \mathbf{b} - s\mathbf{b} \right|^2 \geq \left| \mathbf{a} - \frac{\mathbf{a} \cdot \mathbf{b}}{|\mathbf{b}|^2} \mathbf{b} \right|^2$$

det vi brukte Pythagoras læresetning. Det følger at $\frac{\mathbf{a} \cdot \mathbf{b}}{|\mathbf{b}|^2} \mathbf{b}$ er det unike punktet på linja som er nærmest $\mathbf{a}$. ■

Projeksjonen $\text{proj}_{\mathbf{b}}(\mathbf{a})$ kan regnes ut med koden

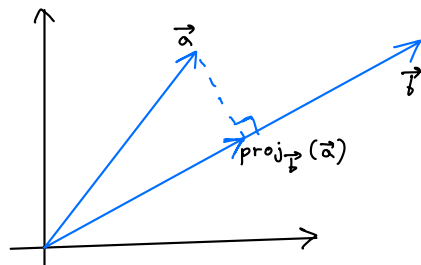
```
vdot(b,a)*b/linalg.norm(b**2)
```

Legg merke til at vi har snudd rekkefølgen på $\mathbf{a}$ og $\mathbf{b}$ når vi tar skalarproduktet her. Skjønner du hvorfor?

Vi ser at $|\text{proj}_{\mathbf{b}}(\mathbf{a})| = \frac{\mathbf{a} \cdot \mathbf{b}}{|\mathbf{b}|}$. Fra beviset over følger det også at vi kan skrive

$$\mathbf{a} = \left(\mathbf{a} - \text{proj}_{\mathbf{b}}(\mathbf{a}) \right) + \text{proj}_{\mathbf{b}}(\mathbf{a}),$$

der de to vektorene på høyre side er ortogonale. Ved hjelp av projeksjon kan vi altså skrive $\mathbf{a}$ som en sum av to vektorer, der en er ortogonal på $\mathbf{b}$, og en parallell med $\mathbf{b}$. Dette kalles også for en *ortogonal dekomposisjon*.



Figur 1.4. Projeksjon av vektoren $\mathbf{a}$ ned på vektoren $\mathbf{b}$ i planet.

Eksempel 1.14. La $\mathbf{a} = (10, 5)$, og $\mathbf{b} = (2, 4)$. Da er

$$|\mathbf{b}|^2 = 4 + 16 = 20 \quad \mathbf{a} \cdot \mathbf{b} = 20 + 20 = 40,$$

slik at

$$\text{proj}_{\mathbf{b}}(\mathbf{a}) = \frac{\mathbf{a} \cdot \mathbf{b}}{|\mathbf{b}|^2} \mathbf{b} = \frac{40}{20} (2, 4) = (4, 8).$$

Vi får også

$$\mathbf{a} - \text{proj}_{\mathbf{b}}(\mathbf{a}) = (10, 5) - (4, 8) = (6, -3).$$

Derfor har vi at

$$(10, 5) = (4, 8) + (6, -3)$$

er en ortogonal dekomposisjon av $(10, 5)$ i en vektor ortogonal på $\mathbf{b}$, og en parallell med $\mathbf{b}$.

Figur 1.4 illustrerer projeksjon i $\mathbb{R}^2$. Det nærmeste punktet til $\mathbf{a}$ på linja gjennom $\mathbf{b}$ og origo fås ved å tegne linja som står vinkelrett på $\mathbf{b}$. Fra figuren er det også klart at

$$\frac{|\text{proj}_{\mathbf{b}}(\mathbf{a})|}{|\mathbf{a}|} = \cos \theta,$$

der θ er vinkelen mellom $\mathbf{a}$ og $\mathbf{b}$. Setter vi inn $\frac{\mathbf{a} \cdot \mathbf{b}}{|\mathbf{b}|^2} \mathbf{b}$ for $\text{proj}_{\mathbf{b}}(\mathbf{a})$ her får vi etter litt regning at $|\mathbf{a} \cdot \mathbf{b}| = |\mathbf{a}| |\mathbf{b}| \cos \theta$, som vel er en kjent formel for de fleste. Fra denne formelen kan vi utlede flere kjente trigonometriske identiteter. Som et eksempel, med $\mathbf{a} = (\cos \theta_1, \sin \theta_1)$, $\mathbf{b} = (\cos \theta_2, \sin \theta_2)$ har vi at

$$\mathbf{a} \cdot \mathbf{b} = \cos \theta_1 \cos \theta_2 + \sin \theta_1 \sin \theta_2 \quad |\mathbf{a}| = 1 \quad |\mathbf{b}| = 1.$$

Siden også vinkelen mellom $\mathbf{a}$ og $\mathbf{b}$ er $\theta = \theta_2 - \theta_1$, så gir innsetting i $|\mathbf{a} \cdot \mathbf{b}| = |\mathbf{a}| |\mathbf{b}| \cos \theta$ derfor at

$$\cos(\theta_2 - \theta_1) = \cos \theta_1 \cos \theta_2 + \sin \theta_1 \sin \theta_2,$$

det vil si at vi har utledet den kjente formelen for cosinus til en differanse av to vinkler.

Fra $|\mathbf{a} \cdot \mathbf{b}| = |\mathbf{a}||\mathbf{b}|\cos\theta$ over følger også at $|\mathbf{a} \cdot \mathbf{b}| \leq |\mathbf{a}||\mathbf{b}|$, ihvertfall for vektorer i planet. Denne ulikheten viser seg å holde mer generelt. For å se dette skal vi bruke Pythagoras læresetning på den ortogonale dekomposisjonen (1.5), som gir

$$|\mathbf{a}|^2 = |\mathbf{a} - \text{proj}_{\mathbf{b}}(\mathbf{a})|^2 + |\text{proj}_{\mathbf{b}}(\mathbf{a})|^2 \geq |\text{proj}_{\mathbf{b}}(\mathbf{a})|^2. \quad (1.6)$$

Tar vi kvadratrøtter på hver side her får vi

$$|\mathbf{a}| \geq |\text{proj}_{\mathbf{b}}(\mathbf{a})| = \left| \frac{\mathbf{a} \cdot \mathbf{b}}{|\mathbf{b}|^2} \mathbf{b} \right| = \left| \frac{\mathbf{a} \cdot \mathbf{b}}{|\mathbf{b}|^2} \right| |\mathbf{b}| = \frac{|\mathbf{a} \cdot \mathbf{b}|}{|\mathbf{b}|}.$$

Ganger vi opp får vi følgende resultat.

Proposisjon 1.15 (Schwarz' ulikhet). Vi har at $|\mathbf{a} \cdot \mathbf{b}| \leq |\mathbf{a}||\mathbf{b}|$ for alle vektorer $\mathbf{a}, \mathbf{b} \in \mathbb{C}^n$. Vi har likhet hvis og bare hvis enten $\mathbf{a}$ og $\mathbf{b}$ er parallelle, eller minst en av $\mathbf{a}$ og $\mathbf{b}$ er 0.

La oss kommentere hvordan man kan finne betingelsene for likhet i Schwarz' ulikhet. På grunn av (1.6) kan dette inntreffe bare når $\text{proj}_{\mathbf{b}}(\mathbf{a}) = \mathbf{a}$. Hvis $\mathbf{a}, \mathbf{b} \neq \mathbf{0}$ er det klart at dette er tilfelle bare når $\mathbf{a}$ og $\mathbf{b}$ er parallelle. Til slutt, likhet i Schwarz ulikhet når enten $\mathbf{a}$ eller $\mathbf{b}$ er $\mathbf{0}$ er opplagt.

En umiddelbar konsekvens av Schwarz' ulikhet er *trekantulikheten*.

Proposisjon 1.16 (Trekantulikheten). Vi har at $|\mathbf{a} + \mathbf{b}| \leq |\mathbf{a}| + |\mathbf{b}|$ for alle vektorer $\mathbf{a}, \mathbf{b} \in \mathbb{C}^n$. Vi har likhet hvis og bare hvis enten $\mathbf{a} = s\mathbf{b}$ for en $s > 0$, eller minst en av $\mathbf{a}$ og $\mathbf{b}$ er 0.

Den opplagte tolkningen av trekantulikheten er at "korteste vei mellom to punkter er en rett linje". Dette ser vi ved å sammenligne avstanden fra origo til $\mathbf{a} + \mathbf{b}$, med lengden du får ved å gå til $\mathbf{a} + \mathbf{b}$ ved først å følge vektoren $\mathbf{a}$, og deretter vektoren $\mathbf{b}$ (se Figur 1.1(a) igjen).

Bevis. Vi har at

$$\begin{aligned} |\mathbf{a} + \mathbf{b}|^2 &= |\mathbf{a}|^2 + \mathbf{a} \cdot \mathbf{b} + \mathbf{b} \cdot \mathbf{a} + |\mathbf{b}|^2 = |\mathbf{a}|^2 + 2\Re(\mathbf{a} \cdot \mathbf{b}) + |\mathbf{b}|^2 \\ &\leq |\mathbf{a}|^2 + 2|\mathbf{a} \cdot \mathbf{b}| + |\mathbf{b}|^2 \leq |\mathbf{a}|^2 + 2|\mathbf{a}||\mathbf{b}| + |\mathbf{b}|^2 = (|\mathbf{a}| + |\mathbf{b}|)^2. \end{aligned}$$

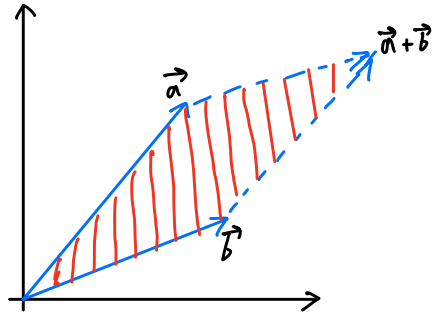
der vi brukte Schwarz ulikhet. Resultatet følger nå ved å ta kvadratrøtter.

For å få likhet må vi først og fremst ha likhet i Schwarz' ulikhet, d.v.s. $\mathbf{a}$ og $\mathbf{b}$ er parallelle, eller minst en av $\mathbf{a}$ og $\mathbf{b}$ er 0. Hvis minst en av $\mathbf{a}$ og $\mathbf{b}$ er 0 ser vi at vi også har likhet i den andre ulikheten, og da blir det også likhet i trekantulikheten. Anta så at $\mathbf{a}$ og $\mathbf{b}$ er parallelle og begge $\neq \mathbf{0}$. Vi har likhet i den andre ulikheten hvis og bare hvis $|\mathbf{a} \cdot \mathbf{b}| = \Re(\mathbf{a} \cdot \mathbf{b})$. Men da må $\mathbf{a} \cdot \mathbf{b}$ være reell og positiv. Siden $\mathbf{b} = s\mathbf{a}$ så blir $\mathbf{a} \cdot \mathbf{b} = \bar{s}|\mathbf{a}|^2$. Det følger at $s > 0$. ■

Oppgaver for Kapittel 1

1. Regn ut $\begin{pmatrix} 2 \\ 1 \\ 4 \end{pmatrix} + 3 \begin{pmatrix} -1 \\ -4 \\ 3 \end{pmatrix}$ og $\begin{pmatrix} -i \\ -1 \\ 3 \end{pmatrix} + i \begin{pmatrix} -i \\ 2 \\ 2 \end{pmatrix}$.
2. Finn en parameterfremstilling for linja som går gjennom de to punktene $(1, 1, 1)$ og $(2, 4, 3)$. Finn også et uttrykk for planet gjennom $(3, -1, -1)$ med normalvektor $(1, 1, 1)$.
3. Regn ut lengden til vektorene $(3, 4, 5)$ og $(i, -i, 1, -1)$.
4. Regn ut
 - (a). Skalarproduktet av $(-1, 2, 4)$ og $(3, 3, -1)$.
 - (b). Skalarproduktet av $(-i, 2, 3 + i)$ og $(2 - i, -4i, -2)$.
5. Regn ut projeksjonen av vektoren $\mathbf{a} = (1, 2, 4)$ ned på $\mathbf{b} = (2, 4, 2)$. Finn også en ortogonal dekomposisjon av $\mathbf{a}$ i en vektor som er ortogonal på $\mathbf{b}$, og en som er parallell med $\mathbf{b}$.
6. Vis egenskap 1.-3. i Proposisjon 1.6 og 1.9.
7. Vis at $|\mathbf{sa}| = |s||\mathbf{a}|$ når $\mathbf{a} \in \mathbb{C}^n$ og $s \in \mathbb{C}$.
8. Bevis at $||\mathbf{a}| - |\mathbf{b}|| \leq |\mathbf{a} - \mathbf{b}|$ for alle $\mathbf{a}, \mathbf{b} \in \mathbb{C}^n$. Dette kalles den *omvendte trekantulikheten*.
9. Med *parallellogrammet* utspent av vektorene $\mathbf{a}$ og $\mathbf{b}$ mener vi firkanten med hjørner $\mathbf{0}$, $\mathbf{a}$, $\mathbf{b}$, og $\mathbf{a} + \mathbf{b}$. Figur 1.5 hvordan et parallellogram kan se ut i planet. Navnet kommer fra at de to parene av motstående sider er parallelle.
 - (a). Forklar geometrisk at parallellogrammet utspent av $\mathbf{a}$ og $\mathbf{b}$ kan skrives som mengden

$$\{r\mathbf{a} + s\mathbf{b} : 0 \leq r, s \leq 1\}.$$



Figur 1.5. Parallelogrammet utspent av vektorene $\mathbf{a}$ og $\mathbf{b}$.

(b). Hvis $\mathbf{a}$ og $\mathbf{b}$ er vektorer enten i planet eller i rommet så gir det mening å snakke om vinkelen mellom de to. La oss kalle denne for θ . Forklar at arealet av parallelogrammet utspent av $\mathbf{a}$ og $\mathbf{b}$ er gitt ved $|\mathbf{a}||\mathbf{b}|\sin\theta$ (vi kan anta at $0 \leq \theta \leq \pi$, slik at $\sin\theta \geq 0$).

10. Vis at, for $\mathbf{a}, \mathbf{b} \in \mathbb{C}^n$, så er

$$|\mathbf{a} + \mathbf{b}|^2 + |\mathbf{a} - \mathbf{b}|^2 = 2(|\mathbf{a}|^2 + |\mathbf{b}|^2).$$

Denne identiteten kalles for parallelogramloven. Kan du tenke deg til hvorfor identiteten kalles dette?

11. Anta at f er en reell funksjon som tar to vektorer i $\mathbb{R}^n$ som parametre. Anta at

1. $f(\mathbf{a}, \mathbf{a}) = |\mathbf{a}|^2$
2. $f(\mathbf{a}, \mathbf{b}) = f(\mathbf{b}, \mathbf{a})$
3. $f(\mathbf{c}, \mathbf{a} + \mathbf{b}) = f(\mathbf{c}, \mathbf{a}) + f(\mathbf{c}, \mathbf{b})$ og $f(\mathbf{a} + \mathbf{b}, \mathbf{c}) = f(\mathbf{a}, \mathbf{c}) + f(\mathbf{b}, \mathbf{c})$

for alle vektorer $\mathbf{a}, \mathbf{b}, \mathbf{c}$ i $\mathbb{R}^n$. Vis at

$$f(\mathbf{a}, \mathbf{b}) = \mathbf{a} \cdot \mathbf{b}$$

for alle $\mathbf{a}, \mathbf{b}$. Spesielt følger det at $f(s\mathbf{a}, \mathbf{b}) = f(\mathbf{a}, s\mathbf{b}) = sf(\mathbf{a}, \mathbf{b})$, siden vi vet at $(s\mathbf{a}) \cdot \mathbf{b} = \mathbf{a} \cdot (s\mathbf{b}) = s(\mathbf{a} \cdot \mathbf{b})$ holder.

KAPITTEL 2

Matriser og matrisemultiplikasjon

I en vektor listes komponentene opp sekvensielt. I en *matrise* listes de opp i to retninger: vertikalt og horisontalt. Vi skal bruke store bokstaver til å representere matriser. Hvis A er en matrise skriver vi a_{ij} for komponenten i rad i og søyle j , og vi bruker parentes til å angi start og slutt på matrisen:

$$A = \begin{pmatrix} a_{11} & a_{12} & \cdots & a_{1n} \\ a_{21} & a_{22} & \cdots & a_{2n} \\ \vdots & \vdots & \vdots & \vdots \\ a_{m1} & a_{m2} & \cdots & a_{mn} \end{pmatrix}.$$

Vi antar alltid at alle radene har like mange komponenter, og at alle søylene har like mange komponenter. Vi sier at A er en $m \times n$ -matrise (eller har dimensjon $m \times n$) hvis den har m rader og n søyler. A kalles *kvadratisk* hvis $m = n$.

Et horisontalt segment i A kalles en *radvektor*, et vertikalt segment en *søylevektor*. En vektor kan tolkes både som en radvektor og som en søylevektor. I koden

```
A = np.array([[1,2,3]])  
B = np.array([[1],[2],[3]])
```

er A en radvektor (eller en 1×3 -matrise), B en søylevektor (eller en 3×1 -matrise). Vi vil som før skrive a_i for komponentene i en vektor, uavhengig av om det er en radvektor eller en søylevektor. Rader og søyler nummereres fra 0 og oppover:

```
>> A[0,1]
```

```
2
>> B[1,0]
2
```

I mange matematikkbøker er det vanlig å tolke en vektor som en søylevektor hvis ikke annet er oppgitt. Når vi skriver $\mathbf{a} = (a_1, a_2, \dots, a_n)$ så skal vi derfor tolke $\mathbf{a}$ som en søylevektor.

3×4 -matrisen $A = \begin{pmatrix} 1 & 2 & 3 & 4 \\ 5 & 6 & 7 & 8 \\ 9 & 10 & 11 & 12 \end{pmatrix}$ fås ved å skrive

```
A = np.array([[1,2,3,4], [5,6,7,8], [9,10,11,12]])
```

Det er enkelt å finne dimensjonene til A :

```
>> m,n=shape(A)
>> m
3
>> n
4
```

Av og til vil vi skrive $\mathbf{a}_{i,:}$ for radvektoren vi får fra rad i i A , og $\mathbf{a}_{:,j}$ for søylevektoren vi får fra søyle j i A . Disse kan også fås med koden

```
A[i, :]
A[:, j]
```

Hakeparenteser blir altså brukt for å markere hvilken del av vektoren vi vil ha tak i. Kolon står her for "alle komponenter". Vi har også en mer generell syntaks for å trekke ut undermatriser av en matrise. Koden under trekker ut 2×3 -matrisen som består av alle komponenter fra rad 1 og 3 som også er i søyle 1, 3, og 4.

```
>> A[np.ix_([0,2], [0,2,3])]
array([[ 1,  3,  4],
       [ 9, 11, 12]])
```

Hvis A er en $m \times n$ -matrise definerer vi den *transponerte* til A som $n \times m$ -matrisen der komponent (i, j) er lik $a_{j,i}$. Vi skriver A^T for denne. For matrisen

$A = \begin{pmatrix} 1 & 2 & 3 & 4 \\ 5 & 6 & 7 & 8 \\ 9 & 10 & 11 & 12 \end{pmatrix}$ over får vi spesielt at $A^T = \begin{pmatrix} 1 & 5 & 9 \\ 2 & 6 & 10 \\ 3 & 7 & 11 \\ 4 & 8 & 12 \end{pmatrix}$. Det finnes en enkel

kommando for å transponere en matrise:

```
>> A.T
array([[ 1,  5,  9],
       [ 2,  6, 10],
       [ 3,  7, 11],
       [ 4,  8, 12]])
```

Hvis $A = A^T$ sier vi at A er *symmetrisk*. Spesielt er

$$A = \begin{pmatrix} 5 & 2 & 3 \\ 2 & 6 & 4 \\ 3 & 4 & 7 \end{pmatrix}.$$

symmetrisk. Symmetriske matriser er alltid kvadratiske, og komponentene er symmetriske om diagonalen. Diagonalen i dette eksemplet består av tallene 5, 6, 7.

Transponering oppfyller enkle regneregler:

$$(A + B)^T = A^T + B^T \quad (A - B)^T = A^T - B^T \quad (A^T)^T = A \quad (sA)^T = sA^T$$

Noen enkle matriser vil være veldig nyttige for oss

- *Nullmatrisen*, skrevet $\mathbf{0}$, er matrisen der alle komponentene er 0.
- *Enermatrisen*, skrevet $\mathbf{1}$, er matrisen der alle komponentene er 1.
- En *diagonalmatrise* er en matrise der alt utenfor diagonalen er null.
- *Identitetsmatrisen*, skrevet I_n , er $n \times n$ -diagonalmatrisen der alt på diagonalen er én.

Disse kan fåes slik:

```
np.zeros((m,n)) # Lager en mxn matrise med bare nuller
np.ones((m,n))  # Lager en mxn matrise med bare enere
np.diag(a)      # Lager en diagonalmatr. med komponentene i a paa diag.
np.eye(n)       # Lager en nxn identitetsmatrise
```

Mange operasjoner gir mening både for vektorer og matriser:

```
>> np.max(A,axis=0)
array([ 9, 10, 11, 12])
>> np.min(A,axis=0)
array([1, 2, 3, 4])
>> np.sum(A,axis=0)
```

```
array([15, 18, 21, 24])
>> np.prod(A,axis=0)
array([ 45, 120, 231, 384])
```

Forskjellen fra vektorversjonen er at operasjonene gjøres søyle for søyle.

2.1 Vektoriserte operasjoner på matriser

Addisjon og subtraksjon av matriser defineres vektorisert, på samme måte som for vektorer: Hvis A og B er begge $m \times n$ så er

- $A + B$ matrisen med komponent (i, j) lik $a_{ij} + b_{ij}$,
- $A - B$ matrisen med komponent (i, j) lik $a_{ij} - b_{ij}$,
- tA matrisen med komponent (i, j) lik $ta_{i,j}$ (der t er en skalar).

Vi har altså at

$$A + B = \begin{pmatrix} a_{11} + b_{11} & a_{12} + b_{12} & \cdots & a_{1n} + b_{1n} \\ a_{21} + b_{21} & a_{22} + b_{22} & \cdots & a_{2n} + b_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ a_{m1} + b_{m1} & a_{m2} + b_{m2} & \cdots & a_{mn} + b_{mn} \end{pmatrix}$$

Det er en enkel oppgave å sjekke at regnereglene i Proposisjon 1.6 også holder hvis vi erstatter vektorene **a** og **b** med matriser.

Eksempel 2.1. Vi har at

$$\begin{aligned} 3 \begin{pmatrix} 1 & 4 \\ 2 & 1 \end{pmatrix} + 2 \begin{pmatrix} -1 & 2 \\ -2 & 1 \end{pmatrix} &= \begin{pmatrix} 3 \cdot 1 & 3 \cdot 4 \\ 3 \cdot 2 & 3 \cdot 1 \end{pmatrix} + \begin{pmatrix} 2 \cdot (-1) & 2 \cdot 2 \\ 2 \cdot (-2) & 2 \cdot 1 \end{pmatrix} \\ &= \begin{pmatrix} 3 & 12 \\ 6 & 3 \end{pmatrix} + \begin{pmatrix} -2 & 4 \\ -4 & 2 \end{pmatrix} = \begin{pmatrix} 3-2 & 12+4 \\ 6-4 & 3+2 \end{pmatrix} \\ &= \begin{pmatrix} 1 & 16 \\ 2 & 5 \end{pmatrix} \end{aligned}$$

Også for matriser tillater Matlab/Python vektoriserte operasjoner med en skalar: Skriver vi $A + b$ der A er en matrise og b en skalar, så returneres

$$\begin{pmatrix} a_{11} + b & a_{12} + b & \cdots & a_{1n} + b \\ a_{21} + b & a_{22} + b & \cdots & a_{2n} + b \\ \vdots & \vdots & \ddots & \vdots \\ a_{m1} + b & a_{m2} + b & \cdots & a_{mn} + b \end{pmatrix}$$

Spesielt vil $A - 1$ returnere matrisen der 1 trekkes fra i alle komponenter, og $1/A$ vil returnere matrisen med komponenter $1/a_{ij}$.

Vektoriste funksjoner defineres på samme måte for matriser. La oss se på noen eksempler.

Eksempel 2.2. Operasjonene `sqrt`, `abs`, `exp`, og `log` så vi var vektorisert for vektorer. Det samme gjelder for matriser. Det samme gjelder de trigonometriske funksjonene:

$$\sin\begin{pmatrix} 0 & \pi/2 \\ \pi & 3\pi/2 \end{pmatrix} = \begin{pmatrix} \sin 0 & \sin(\pi/2) \\ \sin \pi & \sin(3\pi/2) \end{pmatrix} = \begin{pmatrix} 0 & 1 \\ 0 & -1 \end{pmatrix}$$

Dette lar seg enkelt verifisere:

```
>> np.sin(np.array([[0,pi/2],[pi,3*pi/2]]))
array([[ 0.0000000e+00,  1.0000000e+00],
       [ 1.2246468e-16, -1.0000000e+00]])
```

Eksempel 2.3. `max` og `min` kan også brukes vektorisert for matriser.

```
>> A = np.array([[1,2,3],[4,5,6]])
>> B = np.array([[4,5,6],[1,2,3]])
>> np.maximum(A,B)
array([[4, 5, 6],
       [4, 5, 6]])
>> np.minimum(A,B)
array([[1, 2, 3],
       [1, 2, 3]])
```

Det samme gjelder multiplikasjon og potens

```
>> A*B
array([[ 4, 10, 18],
       [ 4, 10, 18]])
>> A**B
array([[ 1, 32, 729],
       [ 4, 25, 216]])
```

Det eksisterer en spesiell syntaks når A er en søylevektor (a) og B en radvek-

tor (b): $a + b$ vil da returnere

$$\begin{pmatrix} a_1 + b_1 & a_1 + b_2 & \cdots & a_1 + b_n \\ a_2 + b_1 & a_2 + b_2 & \cdots & a_2 + b_n \\ \vdots & \vdots & \ddots & \vdots \\ a_m + b_1 & a_m + b_2 & \cdots & a_m + b_n \end{pmatrix},$$

det vil si matrisen med komponent (i, j) lik $a_i + b_j$. Det er jo egentlig ikke definert matematisk å legge sammen en søylevektor og en radvektor, men flere programmeringsspråk tillater likevel dette. Dette høres kanskje rart ut, men kan være nyttig i en del tilfeller, siden det kan gi veldig kompakt kode. La oss se på noen eksempler på dette.

Eksempel 2.4 (Hilbertmatrisen). *Hilbertmatrisen* H har komponenter

$$h_{kl} = 1/(k + l - 1).$$

En $n \times n$ -Hilbertmatrise får man med funksjonen `hilbert(n)` i pakken `scipy.linalg`. Man kan enkelt lage en 4×4 Hilbertmatrise ved hjelp av syntaksen over:

```
>> x = arange(1,5).reshape((1,4))
>> 1/(x + x.T - 1)
array([[1.          , 0.5          , 0.33333333, 0.25          ],
       [0.5          , 0.33333333, 0.25          , 0.2          ],
       [0.33333333, 0.25          , 0.2          , 0.16666667],
       [0.25          , 0.2          , 0.16666667, 0.14285714]])
```

Vi skal vende tilbake til Hilbertmatrisen senere, da det vil vise seg at den er ekstra sårbar med tanke på avrundingsfeil på en datamaskin. Dette gjør den vanskelig å bruke numerisk.

Eksempel 2.5 (Vandermondematriser). Matrisen

$$V = \begin{pmatrix} 1 & x_1 & \cdots & x_1^n \\ 1 & x_2 & \cdots & x_2^n \\ \vdots & \vdots & \ddots & \vdots \\ 1 & x_m & \cdots & x_m^n \end{pmatrix} \quad (2.1)$$

kalles for en *vandermondematrise*, og $x_1, \dots, x_m$ kalles dens *generatorer*. Vandermondematriser fås altså ved å samle potenser av generatorene i hver rad. Slike matriser trenger ikke være kvadratiske - antall generatorer (m) kan være forskjellig fra antall potenser ($n + 1$). Disse matrisene vil dukke opp senere i forbindelse med interpolasjon, og numerisk derivasjon og -integrasjon. Samler vi generatorene til vandermondematrisen i søylevektoren x , og bruker potensene $0, \dots, n$, så får vi matrisen i (2.1) ved å skrive

```
V = x**(arange(0,n+1).reshape((1,-1)))
```

Eksempel 2.6 (Fouriermatrisen). Matriser kan, som vektorer, ha komplekse komponenter. La oss se på et eksempel på en kompleks matrise. *Fouriermatrisen*, F_N , er definert som $N \times N$ -vandermondematriksen med generatorer $\{e^{-2\pi i k/N}\}_{k=0}^{N-1}$. F_N har altså komponenter $e^{-2\pi i kn/N}$, der $0 \leq k, n < N$, og den kan regnes ut ved hjelp av koden

```
inds = arange(0,N).reshape((1,-1))
F = exp( -2*pi*1j*inds.T*inds )
```

I eksponenten her har vi vektorisert multiplikasjon av en søylevektor med en radvektor. Dette gir oss alle mulige verdier for kn . Her brukes eksponentialfunksjonen vektorisert, det vil si at e^A gir matrisen med komponent (k, n) lik $e^{a_{kn}}$.

Fouriermatrisen er veldig nyttig med tanke på anvendelser mot lyd. I denne settingen er ofte N veldig stor, så stor at det kanskje ikke er plass til hele matrisen F_N i minnet på datamaskinen! Det viser seg at Fouriermatrisen kan omskrives slik at man kan bruke den i beregninger likevel.

2.2 Matrisemultiplikasjon

Vi skal senere se på hvordan vi kan løse lineære likningssystemer med m likninger og n ukjente. Et generelt slikt system kan skrives

$$\begin{array}{cccccc} a_{11}x_1 & +a_{12}x_2 & +\cdots & +a_{1n}x_n & = & b_1 \\ a_{21}x_1 & +a_{22}x_2 & +\cdots & +a_{2n}x_n & = & b_2 \\ \vdots & \vdots & \vdots & \vdots & & \vdots \\ a_{m1}x_1 & +a_{m2}x_2 & +\cdots & +a_{mn}x_n & = & b_m \end{array} \quad (2.2)$$

La oss skrive

$$A = \begin{pmatrix} a_{11} & a_{12} & \cdots & a_{1n} \\ a_{21} & a_{22} & \cdots & a_{2n} \\ \vdots & \vdots & \vdots & \vdots \\ a_{m1} & a_{m2} & \cdots & a_{mn} \end{pmatrix} \quad \mathbf{x} = \begin{pmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{pmatrix} \quad \mathbf{b} = \begin{pmatrix} b_1 \\ b_2 \\ \vdots \\ b_m \end{pmatrix}$$

A kalles for *koeffisientmatrisen* til systemet. For å kunne skrive (2.2) mer kompakt er følgende definisjon nyttig.

Definisjon 2.7 (Matrise-vektor multiplikasjon). La A være en $m \times n$ -matrise, og $\mathbf{x}$ en søylevektor av lengde n . Vi definerer $A\mathbf{x}$ som vektoren

$$A\mathbf{x} = \begin{pmatrix} a_{11}x_1 + a_{12}x_2 + \cdots + a_{1n}x_n \\ a_{21}x_1 + a_{22}x_2 + \cdots + a_{2n}x_n \\ \vdots \\ a_{m1}x_1 + a_{m2}x_2 + \cdots + a_{mn}x_n \end{pmatrix} = x_1\mathbf{a}_1 + x_2\mathbf{a}_2 + \cdots + x_n\mathbf{a}_n.$$

Med denne definisjonen kan (2.2) skrives mer kompakt som $A\mathbf{x} = \mathbf{b}$. Omskrivingen til $x_1\mathbf{a}_1 + x_2\mathbf{a}_2 + \cdots + x_n\mathbf{a}_n$ over følger fra regnereglene for vektorer. Merk at $A\mathbf{x}$ er definert bare hvis antall søyler i A er lik antall komponenter i $\mathbf{x}$, og at $A\mathbf{x}$ da kan tolkes som samlingen av m skalarprodukter (mellom radene i A , og $\mathbf{x}$).

Eksempel 2.8. La oss regne ut $A\mathbf{x}$ når $A = \begin{pmatrix} 2 & 1 & -2 \\ 3 & 0 & -1 \\ 7 & -4 & 2 \end{pmatrix}$ og $\mathbf{x} = \begin{pmatrix} 3 \\ -1 \\ -2 \end{pmatrix}$. Dette er veldefinert siden antall søyler i A er lik antall komponenter i $\mathbf{x}$ (begge er 3). Vi får

$$\begin{pmatrix} 2 & 1 & -2 \\ 3 & 0 & -1 \\ 7 & -4 & 2 \end{pmatrix} \begin{pmatrix} 3 \\ -1 \\ -2 \end{pmatrix} = \begin{pmatrix} 2 \cdot 3 + 1 \cdot (-1) + (-2) \cdot (-2) \\ 3 \cdot 3 + 0 \cdot (-1) + (-1) \cdot (-2) \\ 7 \cdot 3 + (-4) \cdot (-1) + 2 \cdot (-2) \end{pmatrix} = \begin{pmatrix} 6 - 1 + 4 \\ 9 + 0 + 2 \\ 21 + 4 - 4 \end{pmatrix} = \begin{pmatrix} 9 \\ 11 \\ 21 \end{pmatrix}$$

Anta at vi vil løse likningssystemet (2.2) for forskjellige høyresider $\mathbf{b}_1, \dots, \mathbf{b}_k$:

$$A\mathbf{x}_1 = \mathbf{b}_1 \quad A\mathbf{x}_2 = \mathbf{b}_2 \quad \cdots \quad A\mathbf{x}_k = \mathbf{b}_k$$

Vi søker en kompakt notasjon også for samlingen av disse systemene. For dette er følgende definisjon nyttig.

Definisjon 2.9 (Matrisemultiplikasjon). Hvis A er en $m \times n$ matrise, og X en $n \times k$ -matrise, så definerer vi AX som $m \times k$ -matrisen

$$AX = (AX_{:,1} \quad AX_{:,2} \quad \cdots \quad AX_{:,k}) \quad (2.3)$$

Setter vi nå

$$X = (\mathbf{x}_1 \quad \mathbf{x}_2 \quad \cdots \quad \mathbf{x}_k) \quad B = (\mathbf{b}_1 \quad \mathbf{b}_2 \quad \cdots \quad \mathbf{b}_k),$$

så kan søyle j i likningen $AX = B$ skrives som $A\mathbf{x}_j = \mathbf{b}_j$. $AX = B$ kan altså tolkes som sammensetningen av k likningssystemer. AX er bare definert når antall søyler i A er lik antall rader i X .

La oss finne en praktisk formel for å regne ut et matriseprodukt. Fra definisjonen av matrise-vektor produktet vet vi at $AX_{:,j} = \sum_{r=1}^n a_{ir}x_{rj}$. Likning (2.3) sier nå at dette er lik søyle j i AX , og det følger at

$$(AX)_{ij} = \sum_{r=1}^n a_{ir}x_{rj}. \quad (2.4)$$

Dette kalles også for *indre form* for et matriseprodukt.

Eksempel 2.10. La oss regne ut matriseproduktet

$$\begin{pmatrix} 2 & 4 & 3 \\ 1 & -1 & 3 \end{pmatrix} \begin{pmatrix} 2 & -1 \\ 4 & 1 \\ 2 & -1 \end{pmatrix}$$

Produktet skal da bli en 2×2 -matrise. Bruker vi (2.4) til å regne ut komponent (1, 1) i dette får vi

$$2 \cdot 2 + 4 \cdot 4 + 3 \cdot 2 = 4 + 16 + 6 = 26$$

Gjør vi det samme for komponent (1, 2) får vi

$$2 \cdot (-1) + 4 \cdot 1 + 3 \cdot (-1) = -2 + 4 - 3 = -1.$$

Vi har dermed regnet ut først rad i matriseproduktet. Det er vanlig at man skriver ut disse utregningene som følger:

$$\begin{aligned} \begin{pmatrix} 2 & 4 & 3 \\ 1 & -1 & 3 \end{pmatrix} \begin{pmatrix} 2 & -1 \\ 4 & 1 \\ 2 & -1 \end{pmatrix} &= \begin{pmatrix} 2 \cdot 2 + 4 \cdot 4 + 3 \cdot 2 & 2 \cdot (-1) + 4 \cdot 1 + 3 \cdot (-1) \\ 1 \cdot 2 - 1 \cdot 4 + 3 \cdot 2 & 1 \cdot (-1) - 1 \cdot 1 + 3 \cdot (-1) \end{pmatrix} \\ &= \begin{pmatrix} 4 + 16 + 6 & -2 + 4 - 3 \\ 2 - 4 + 6 & -1 - 1 - 3 \end{pmatrix} = \begin{pmatrix} 26 & -1 \\ 4 & -5 \end{pmatrix} \end{aligned}$$

Vi kan kontrollregne dette ved å bruke den innebygde operatoren for matrisemultiplikasjon.

```
>> A = array([[2,4,3],[1,-1,3]])
>> B = array([[2,-1],[4,1],[2,-1]])
>> A@B
array([[26, -1],
       [ 4, -5]])
```

Også matrisemultiplikasjon kan tolkes ved hjelp av skalarprodukter: Over regnet vi ut 4 skalarprodukter: Hvis matrisen til venstre har rader $\mathbf{a}_1$ og $\mathbf{a}_2$, og matrisen til høyre har søyler $\mathbf{b}_1$ og $\mathbf{b}_2$, så regnet vi ut

$$\mathbf{a}_1 \cdot \mathbf{b}_1 = 26 \quad \mathbf{a}_1 \cdot \mathbf{b}_2 = -1 \quad \mathbf{a}_2 \cdot \mathbf{b}_1 = 4 \quad \mathbf{a}_2 \cdot \mathbf{b}_2 = -5.$$

Eksempel 2.11 (Ytreproduktet av to vektorer). La $\mathbf{a} \in \mathbb{R}^m$ og $\mathbf{x} \in \mathbb{R}^n$ være søylevektorer. Da er matriseproduktet

$$\mathbf{a}\mathbf{x}^T = \begin{pmatrix} a_1 \\ a_2 \\ \vdots \\ a_m \end{pmatrix} \begin{pmatrix} x_1 & x_2 & \cdots & x_n \end{pmatrix} = \begin{pmatrix} a_1 x_1 & a_1 x_2 & \cdots & a_1 x_n \\ a_2 x_1 & a_2 x_2 & \cdots & a_2 x_n \\ \vdots & \vdots & \ddots & \vdots \\ a_m x_1 & a_m x_2 & \cdots & a_m x_n \end{pmatrix}$$

$\mathbf{a}\mathbf{x}^T$ kalles også for *ytreproduktet* av $\mathbf{a}$ og $\mathbf{x}$.

Matrisemultiplikasjon tilfredsstiller mange kjente regneregler:

Proposisjon 2.12. La A , B , og C være matriser. Da har vi at

1. $(AB)C = A(BC)$
2. $A(B + C) = AB + AC$
3. $(B + C)A = BA + CA$
4. For alle skalarer s har vi at $s(AB) = (sA)B = A(sB)$.
5. Hvis A er $m \times n$ så er $AI_n = A$ og $I_m A = A$ (det vil si at identitetsmatrisen oppfører seg på samme måte som tallet 1 ved multiplikasjon av reell tall).

Resultatet krever at de de tilhørende matriseproduktene er definert.

Hvis vi ganger sammen mange matriser så medfører 1. her at det ikke spiller noen rolle hvilke "nabomatriser" vi ganger sammen først. Det spiller derfor ingen rolle hvor vi setter parentesene ved matrisemultiplikasjon, og det er derfor vanlig å ikke skrive de opp. Vi skriver altså ABC i stedet for $(AB)C = A(BC)$. Vi kan derimot ikke bytte rekkefølgen på matrisene: Hvis A er $m \times n$ og B er $n \times m$, så er det klart at AB og BA begge er definert, men de har forskjellig dimensjon når $m \neq n$. Selv om $m = n$ er vanligvis $AB \neq BA$, som følgende eksempel viser:

$$\begin{pmatrix} 1 & 0 \\ 0 & 0 \end{pmatrix} \begin{pmatrix} 0 & 0 \\ 1 & 0 \end{pmatrix} = \begin{pmatrix} 0 & 0 \\ 0 & 0 \end{pmatrix} \quad \begin{pmatrix} 0 & 0 \\ 1 & 0 \end{pmatrix} \begin{pmatrix} 1 & 0 \\ 0 & 0 \end{pmatrix} = \begin{pmatrix} 0 & 0 \\ 1 & 0 \end{pmatrix}.$$

Mens rekkefølgen ved multiplikasjon ikke betyr noe for reelle tall, så er altså ikke det samme tilfellet for matriser!

Bevis. 1. er kanskje vanskeligst å bevise, siden vi får en ”dobbeltsum” når vi ganger sammen tre matriser. La oss derfor se nærmere på denne. Hvis A er $m \times k$, B $k \times l$, C $l \times n$, så er både $(AB)C$ og $A(BC)$ veldefinerte. Videre er AB $m \times l$, og BC $k \times n$, og både $(AB)C$ og $A(BC)$ er $m \times n$. Definisjonen av matrisemultiplikasjon gir

$$\begin{aligned} ((AB)C)_{ij} &= \sum_{s=1}^l (AB)_{is} c_{sj} = \sum_{s=1}^l \left(\sum_{r=1}^k a_{ir} b_{rs} \right) c_{sj} = \sum_{s=1}^l \sum_{r=1}^k a_{ir} b_{rs} c_{sj} \\ (A(BC))_{ij} &= \sum_{r=1}^k a_{ir} (BC)_{rj} = \sum_{r=1}^k a_{ir} \left(\sum_{s=1}^l b_{rs} c_{sj} \right) = \sum_{r=1}^k \sum_{s=1}^l a_{ir} b_{rs} c_{sj}. \end{aligned}$$

Et par kommentarer er på sin plass:

- i den første overgangen på første linje brukte vi definisjonen av matriseproduktet mellom AB og C ,
- i den første overgangen på andre linje brukte vi definisjonen av matriseproduktet mellom A og BC ,
- i den andre overgangen på første linje brukte vi definisjonen av matriseproduktet AB ,
- i den andre overgangen på andre linje brukte vi definisjonen av matriseproduktet BC .

Summasjonsrekkefølgen spiller her ingen rolle, slik at vi får samme resultat. Derfor er alle komponentene i $(AB)C$ og $A(BC)$ like, slik at $(AB)C = A(BC)$.

2.-4. er noe enklere å vise, og er gitt som en oppgave.

5.: Vi har at

$$(AI_n)_{ij} = \sum_{r=1}^n a_{ir} (I_n)_{rj} = a_{ij} (I_n)_{jj} = a_{ij},$$

der vi brukte at I_n bare har en ikke-null i hver søyle. Siden alle komponentene er like følger det at $AI_n = A$. På samme måte viser man at $I_m A = A$. ■

Det finnes også en annen form for et matriseprodukt, som er nyttig enkelte ganger, og som vi skal komme tilbake til. Denne formen bruker notasjonen $\mathbf{a}_{\cdot r}$ for søyle r i A , og $\mathbf{x}_r$ for rad r i X :

Proposisjon 2.13 (Ytre form for et matriseprodukt). Vi har at

$$AX = \sum_{r=1}^n \mathbf{a}_{:r} \mathbf{x}_r.$$

Dette kalles også for den ytre formen til et matriseprodukt.

Bevis. La $\mathbf{e}_i$ være søylevektoren som er 1 i komponent i , og 0 ellers. Fra Eksempel 2.11 følger det at

- $\mathbf{a}_{:j} \mathbf{e}_j^T$ er matrisen av samme størrelse som A , der søyle j er lik søyle j i A , og der alle andre søyler er $\mathbf{0}$.
- $\mathbf{e}_i \mathbf{x}_i$ er matrisen av samme størrelse som X , der rad i er lik rad i i X , og der alle andre rader er $\mathbf{0}$.

Dette betyr at vi kan skrive $A = \sum_{j=1}^n \mathbf{a}_{:j} \mathbf{e}_j^T$, og $X = \sum_{i=1}^n \mathbf{e}_i \mathbf{x}_i$. Bruker vi dette får vi

$$AX = \left(\sum_{j=1}^n \mathbf{a}_{:j} \mathbf{e}_j^T \right) \left(\sum_{i=1}^n \mathbf{e}_i \mathbf{x}_i \right) = \sum_{i=1}^n \sum_{j=1}^n \mathbf{a}_{:j} \mathbf{e}_j^T \mathbf{e}_i \mathbf{x}_i = \sum_{r=1}^n \mathbf{a}_{:r} \mathbf{x}_r.$$

der vi brukte at $\mathbf{e}_j^T \mathbf{e}_i = 1$ hvis $i = j$, og 0 ellers. Dette var det vi skulle fram til. ■

I beviset over brukte vi flere av egenskapene fra Proposisjon 2.12. Hvilke?

2.3 Lineære avbildninger og matrisemultiplikasjon

En avbildning $\mathbf{F}$ fra $\mathbb{R}^n$ til $\mathbb{R}^m$ kalles *lineær* hvis vi har at, for alle $\mathbf{a}, \mathbf{a}_1, \mathbf{a}_2 \in \mathbb{R}^n$, og alle skalarer c ,

1. $\mathbf{F}(\mathbf{a}_1 + \mathbf{a}_2) = \mathbf{F}(\mathbf{a}_1) + \mathbf{F}(\mathbf{a}_2)$
2. $\mathbf{F}(c\mathbf{a}) = c\mathbf{F}(\mathbf{a})$

Fra Proposisjon 2.12 følger det at avbildningen $\mathbf{x} \rightarrow A\mathbf{x}$, der A er en matrise, er lineær. Omvendt skal vi nå vise at alle lineære avbildninger kan skrives på denne formen:

Proposisjon 2.14. Anta $\mathbf{F}$ er en lineær avbildning fra $\mathbb{R}^n$ til $\mathbb{R}^m$. Da finnes det en $m \times n$ -matrise A slik at $\mathbf{F}(\mathbf{x}) = A\mathbf{x}$ for alle $\mathbf{x} \in \mathbb{R}^n$. Mer presist er

$$A = (\mathbf{F}(\mathbf{e}_1) \quad \mathbf{F}(\mathbf{e}_2) \quad \cdots \quad \mathbf{F}(\mathbf{e}_n)), \quad (2.5)$$

der $\mathbf{e}_i$ var søylevektoren som er 1 i komponent i , og 0 ellers.

Bevis. For beviset skal vi bruke at

$$(x_1, x_2, \dots, x_n) = x_1 \mathbf{e}_1 + x_2 \mathbf{e}_2 + \cdots + x_n \mathbf{e}_n,$$

som er lett å vise. Dette gir

$$\begin{aligned} \mathbf{F}(x_1, x_2, \dots, x_n) &= \mathbf{F}(x_1 \mathbf{e}_1 + x_2 \mathbf{e}_2 + \cdots + x_n \mathbf{e}_n) \\ &= x_1 \mathbf{F}(\mathbf{e}_1) + x_2 \mathbf{F}(\mathbf{e}_2) + \cdots + x_n \mathbf{F}(\mathbf{e}_n). \end{aligned}$$

Ved definisjonen av matrise-vektor multiplikasjonen kan dette skrives

$$(\mathbf{F}(\mathbf{e}_1) \quad \mathbf{F}(\mathbf{e}_2) \quad \cdots \quad \mathbf{F}(\mathbf{e}_n)) \begin{pmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{pmatrix}.$$

Det følger at alle lineære avbildninger kan skrives på formen $\mathbf{x} \rightarrow A\mathbf{x}$, og at A er gitt ved (2.5). ■

Vi trenger altså bare regne ut $\mathbf{F}(\mathbf{e}_i)$ for å finne matrisen til en lineær avbildning. La oss se på noen nyttige lineære avbildninger, og finne matrisene deres.

Eksempel 2.15 (Projeksjoner). Det følger fra regnereglene for skalarproduktet at projeksjonen $\text{proj}_{\mathbf{b}}(\mathbf{a}) = \frac{\mathbf{a} \cdot \mathbf{b}}{|\mathbf{b}|^2} \mathbf{b}$ er lineær. Matrisen for denne er

$$\begin{aligned} &(\text{proj}_{\mathbf{b}}(\mathbf{e}_1) \quad \text{proj}_{\mathbf{b}}(\mathbf{e}_2) \quad \cdots \quad \text{proj}_{\mathbf{b}}(\mathbf{e}_n)) \\ &= \left(\frac{\mathbf{e}_1 \cdot \mathbf{b}}{|\mathbf{b}|^2} \mathbf{b} \quad \frac{\mathbf{e}_2 \cdot \mathbf{b}}{|\mathbf{b}|^2} \mathbf{b} \quad \cdots \quad \frac{\mathbf{e}_n \cdot \mathbf{b}}{|\mathbf{b}|^2} \mathbf{b} \right) = \left(\frac{b_1}{|\mathbf{b}|} \frac{\mathbf{b}}{|\mathbf{b}|} \quad \frac{b_2}{|\mathbf{b}|} \frac{\mathbf{b}}{|\mathbf{b}|} \quad \cdots \quad \frac{b_n}{|\mathbf{b}|} \frac{\mathbf{b}}{|\mathbf{b}|} \right) \\ &= \left(\frac{\mathbf{b}}{|\mathbf{b}|} \frac{b_1}{|\mathbf{b}|} \quad \frac{\mathbf{b}}{|\mathbf{b}|} \frac{b_2}{|\mathbf{b}|} \quad \cdots \quad \frac{\mathbf{b}}{|\mathbf{b}|} \frac{b_n}{|\mathbf{b}|} \right) = \frac{\mathbf{b}}{|\mathbf{b}|} \begin{pmatrix} \frac{b_1}{|\mathbf{b}|} & \frac{b_2}{|\mathbf{b}|} & \cdots & \frac{b_n}{|\mathbf{b}|} \end{pmatrix} = \frac{\mathbf{b}}{|\mathbf{b}|} \left(\frac{\mathbf{b}}{|\mathbf{b}|} \right)^T. \end{aligned}$$

Nest siste overgang her brukte definisjonen av matrisemultiplikasjon, mens siste overgang brukte at $\frac{\mathbf{b}}{|\mathbf{b}|} = \begin{pmatrix} b_1/|\mathbf{b}| \\ b_2/|\mathbf{b}| \\ \vdots \\ b_n/|\mathbf{b}| \end{pmatrix}$, slik at $\left(\frac{\mathbf{b}}{|\mathbf{b}|} \right)^T = \begin{pmatrix} \frac{b_1}{|\mathbf{b}|} & \frac{b_2}{|\mathbf{b}|} & \cdots & \frac{b_n}{|\mathbf{b}|} \end{pmatrix}$. Det følger

at, hvis $\mathbf{b}$ er en vektor med lengde 1, så regner matrisen $\mathbf{b}\mathbf{b}^T$ ut projeksjon ned på linja gjennom origo med retningsvektor $\mathbf{b}$. Slike matriser kalles derfor *projeksjonsmatriser*.

Eksempel 2.16 (Rotasjoner). La oss se på avbildningen $\mathbf{F}$ som roterer en vektor i $\mathbb{R}^2$ med vinkel θ . Bruker vi polarform får vi

$$\begin{aligned}\mathbf{F}\begin{pmatrix} r \cos \alpha \\ r \sin \alpha \end{pmatrix} &= \begin{pmatrix} r \cos(\alpha + \theta) \\ r \sin(\alpha + \theta) \end{pmatrix} = \begin{pmatrix} \cos \theta r \cos \alpha - \sin \theta r \sin \alpha \\ \sin \theta r \cos \alpha + \cos \theta r \sin \alpha \end{pmatrix} \\ &= \begin{pmatrix} \cos \theta & -\sin \theta \\ \sin \theta & \cos \theta \end{pmatrix} \begin{pmatrix} r \cos \alpha \\ r \sin \alpha \end{pmatrix}\end{aligned}$$

Det følger at $\mathbf{F}$ er lineær, med matrise $\begin{pmatrix} \cos \theta & -\sin \theta \\ \sin \theta & \cos \theta \end{pmatrix}$. Hvis vi tar utgangspunkt i at $\mathbf{F}$ er lineær, så er det enkelt å regne ut matrisen til $\mathbf{F}$:

$$\mathbf{F}(\mathbf{e}_1) = \begin{pmatrix} \cos \theta \\ \sin \theta \end{pmatrix} \quad \mathbf{F}(\mathbf{e}_2) = \begin{pmatrix} \cos(\theta + \pi/2) \\ \sin(\theta + \pi/2) \end{pmatrix} = \begin{pmatrix} -\sin \theta \\ \cos \theta \end{pmatrix}.$$

2.4 Plotting av polynomer

La oss se på hvordan matrisemultiplikasjon naturlig dukker i forbindelse med plotting av polynomer. Kommandoen `plot` i pakken `matplotlib.pyplot` kan brukes til å plote kurver i planet. Den vanlige syntaksen for `plot` tar to argumenter x og y , som inneholder x - og y -koordinatene til plottepunktene, respektive. `plot`-kommandoen kan brukes i mange variasjoner. Skal vi for eksempel plote polynomene

$$p_1(x) = x \qquad p_2(x) = x^2 \qquad p_3(x) = x^3$$

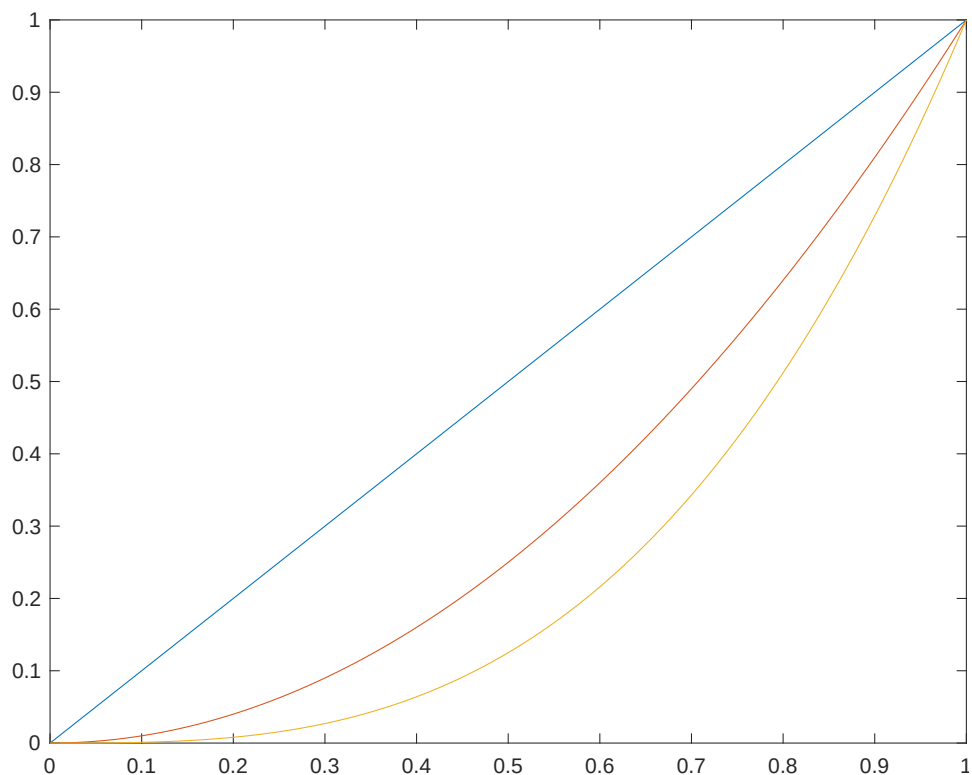
mot hverandre over intervallet $[0, 1]$, så kan dette gjøres slik:

```
import matplotlib.pyplot as plt

x=np.linspace(0,1,100)
plt.plot(x, x)
plt.plot(x, x**2)
plt.plot(x, x**3)
```

Enda mer kompakt kan de fire siste linjene over skrives

```
plt.plot(x, x, x, x**2, x, x**3)
```



Figur 2.1. Plott av polynomene $p_1(x) = x$, $p_2(x) = x^2$, og $p_3(x) = x^3$.

der det er gitt 6 argumenter, og der hvert par representerer ett plott. Resultatet kan du se i Figur 2.1. Hvis polynomene er av høyere grad, og med mange ledd, så blir ting litt mer komplisert. La oss se på hvordan vi kan systematisere dette ved å samle koeffisientene til polynomene i vektorer. La det første polynomet være

$$p_1(x) = c_0 + c_1 x + \cdots + c_n x^n,$$

og la plottepunktene være $x_1, \dots, x_m$. Vi skriver

$$\begin{aligned} p_1(x_1) &= c_0 + x_1 c_1 + \cdots + x_1^n c_n \\ p_1(x_2) &= c_0 + x_2 c_1 + \cdots + x_2^n c_n \\ &\vdots \\ p_1(x_m) &= c_0 + x_m c_1 + \cdots + x_m^n c_n. \end{aligned}$$

Ved definisjonen av matrise-vektor-multiplikasjon, kan dette skrives

$$\begin{pmatrix} p_1(x_1) \\ p_1(x_2) \\ \vdots \\ p_1(x_m) \end{pmatrix} = \begin{pmatrix} 1 & x_1 & \dots & x_1^n \\ 1 & x_2 & \dots & x_2^n \\ \vdots & \vdots & \ddots & \vdots \\ 1 & x_m & \dots & x_m^n \end{pmatrix} \begin{pmatrix} c_0 \\ c_1 \\ \vdots \\ c_n \end{pmatrix}, \quad (2.6)$$

der vi gjenkjenner vandermondematriksen fra Eksempel 2.5. Hvis vi nå har enda et polynom, $p_2(x) = d_0 + d_1x + \dots + d_nx^n$, kan vi på samme måte skrive

$$\begin{pmatrix} p_2(x_1) \\ p_2(x_2) \\ \vdots \\ p_2(x_m) \end{pmatrix} = \begin{pmatrix} 1 & x_1 & \dots & x_1^n \\ 1 & x_2 & \dots & x_2^n \\ \vdots & \vdots & \ddots & \vdots \\ 1 & x_m & \dots & x_m^n \end{pmatrix} \begin{pmatrix} d_0 \\ d_1 \\ \vdots \\ d_n \end{pmatrix}. \quad (2.7)$$

Ved definisjonen av matrisemultiplikasjon kan (2.6) og (2.7) slås sammen til

$$\begin{pmatrix} p_1(x_1) & p_2(x_1) \\ p_1(x_2) & p_2(x_2) \\ \vdots & \vdots \\ p_1(x_m) & p_2(x_m) \end{pmatrix} = \begin{pmatrix} 1 & x_1 & \dots & x_1^n \\ 1 & x_2 & \dots & x_2^n \\ \vdots & \vdots & \ddots & \vdots \\ 1 & x_m & \dots & x_m^n \end{pmatrix} \begin{pmatrix} c_0 & d_0 \\ c_1 & d_1 \\ \vdots & \vdots \\ c_n & d_n \end{pmatrix}, \quad (2.8)$$

og på samme måte hvis vi har enda flere polynomer. Vi definerer plottepunktene over $[0, 1]$, og regner ut vandermondematriksen som står på høyre side:

```
x = np.linspace(0,1,100).reshape((-1,1))
V = x**(arange(0,n+1).reshape((1,-1)))
```

Hvis vi samler koeffisientene til de tre polynomene p_1 , p_2 , og p_3 , i søylevektorer p_1 , p_2 , og p_3 , så sier (2.8) at

```
V @ np.block([p1,p2,p3])
```

gir en matrise med tre søyler, der søyle 1 inneholder punktverdiene for p_1 , søyle 2 for p_2 , og søyle 3 for p_3 . Polynomene kan nå plottes sammen ved å skrive

```
plt.plot(x, V @ np.block([p1,p2,p3]))
plt.legend(['p_1', 'p_2', 'p_3'])
```

Dette er en litt annen syntaks for plot-kommandoen, der

- første argument er søylevektoren med plottepunkter, og

- andre argument er en matrise med like mange rader som i første argument, og tre søyler.

Hvis x er en søylevektor og Y en matrise med like mange rader som x og tre søyler, så er $\text{plot}(x, Y)$ det samme som å skrive

```
plt.plot(x, V@p1, x, V@p2, x, V@p3)
```

Dette gjelder også mer generelt hvis vi har flere søyler med polynomer: Syntaksen $\text{plot}(x, Y)$ plottet alle søyler i Y mot samme vektor x .

2.5 Blokkmatriser

Matriser kan også brukes som byggeklosser for større matriser. Hvis vi har fire matriser A, B, C, D , så skriver vi

$$\left(\begin{array}{c|c} A & B \\ \hline C & D \end{array} \right) \quad (2.9)$$

for matrisen der komponentene i disse fire matrisene er plassert over, under, til venstre, eller til høyre for hverandre, som angitt. Streker brukes til å adskille blokkene. Dette får vi med koden

```
np.block([[A,B],[C,D]])
```

Vi kaller dette for en *blokkmatrise*. Uttrykket (2.9) gir bare mening hvis

- A og B har like mange rader,
- C og D har like mange rader,

for da kan jo disse settes ved siden av hverandre. På samme måte må A og C ha like mange søyler, B og D ha like mange søyler, for at de skal kunne settes under hverandre. Det er lett å generalisere til blokkmatriser med flere enn 4 blokker.

Eksempel 2.17. Hvis

$$A = \begin{pmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 \end{pmatrix} \quad B = \begin{pmatrix} 2 & 2 \\ 2 & 2 \end{pmatrix} \quad C = \begin{pmatrix} 3 & 3 & 3 \\ 3 & 3 & 3 \\ 3 & 3 & 3 \end{pmatrix} \quad D = \begin{pmatrix} 4 & 4 \\ 4 & 4 \\ 4 & 4 \end{pmatrix}$$

så er

$$\left(\begin{array}{c|c} A & B \\ \hline C & D \end{array} \right) = \begin{pmatrix} 1 & 1 & 1 & 2 & 2 \\ 1 & 1 & 1 & 2 & 2 \\ 3 & 3 & 3 & 4 & 4 \\ 3 & 3 & 3 & 4 & 4 \\ 3 & 3 & 3 & 4 & 4 \end{pmatrix}$$

noe vi lett kan sjekke med følgende kode:

```
>> A = array([[1,1,1],[1,1,1]])
>> B = array([[2,2],[2,2]])
>> C = array([[3,3,3],[3,3,3],[3,3,3]])
>> D = array([[4,4],[4,4],[4,4]])
>> np.block([[A,B],[C,D]])
array([[1, 1, 1, 2, 2],
       [1, 1, 1, 2, 2],
       [3, 3, 3, 4, 4],
       [3, 3, 3, 4, 4],
       [3, 3, 3, 4, 4],
       [3, 3, 3, 4, 4]])
```

Produktet

$$\left(\begin{array}{c|c} A & B \\ \hline C & D \end{array} \right) \left(\begin{array}{c|c} E & F \\ \hline G & H \end{array} \right)$$

av to blokkmatriser gir mening hvis

- antall søyler i A er lik antall rader i E ,
- antall søyler i B er lik antall rader i G .

Da blir dette nemlig

$$\left(\begin{array}{c|c} AE + BG & AF + BH \\ \hline CE + DG & CF + DH \end{array} \right)$$

det vil si at blokkmatriser kan multipliseres som for matriser. Koden blir da

```
np.block([[A,B],[C,D]])@np.block([[E,F],[G,H]])
```

Oppgaver for Kapittel 2

1. Regn ut

$$(a). \begin{pmatrix} 2 & 4 & 3 \\ -1 & 2 & 4 \\ 4 & 1 & -1 \end{pmatrix} - \begin{pmatrix} -1 & 2 & 8 \\ -1 & -2 & -5 \\ 2 & 4 & -2 \end{pmatrix}$$

$$(b). 2 \begin{pmatrix} 3 & 2 \\ -1 & -2 \end{pmatrix} + 4 \begin{pmatrix} -3 & 3 \\ -1 & -1 \end{pmatrix}$$

2. Vis at regnereglene i Proposisjon 1.6 også holder hvis vi erstatter vektorene $\mathbf{a}$ og $\mathbf{b}$ med matriser.

3. Regn ut de transponerte av matrisene $\begin{pmatrix} 2 & 4 \\ 3 & -1 \\ 0 & 1 \end{pmatrix}$ og $\begin{pmatrix} 1 & 4 & 7 \\ 2 & 4 & 3 \\ -1 & -2 & 1 \end{pmatrix}$.

4. Funksjonen `rand(m,n)` returnerer en $m \times n$ -matrise med komponenter mellom 0 og 1, og som er tilfeldig generert. La oss se på koden

```
m = 10
n = 7
A = np.random.rand(m,n)
B = np.random.rand(m,n)
C = np.random.rand(m,n)
print( ((A+B)+C) - (A+(B+C)) )
```

Hvilken regneregel for matriser vil denne koden verifisere?

5 (Midtveiseksamen 2024). Vi har matrisene $A = \begin{pmatrix} 1 & 4 & -1 \\ -2 & -1 & 4 \end{pmatrix}$ og $B = \begin{pmatrix} 1 & 0 \\ 3 & 2 \\ 4 & 2 \end{pmatrix}$.

Hvilket utsagn er riktig?

1. $AB = \begin{pmatrix} 2 & 7 \\ 11 & 6 \end{pmatrix}$

2. $AB = \begin{pmatrix} 9 & 6 \\ 11 & 6 \end{pmatrix}$

3. $AB = \begin{pmatrix} 9 & 6 \\ 10 & 8 \end{pmatrix}$

4. $AB = \begin{pmatrix} 1 & 4 & -1 \\ -1 & 10 & 5 \\ 0 & 14 & 4 \end{pmatrix}$

5. AB er ikke definert

6 (Midtveiseksamen 2024). La $A \in \mathbb{R}^{3 \times 5}$ og $B \in \mathbb{R}^{6 \times 3}$. Hvilket av følgende utsagn er sant?

1. Produktet AB er definert.

2. Ligningen $BA = C$ har løsningen $A = B^{-1}C$.

3. Matrisen B inneholder dobbelt så mange komponenter som A .

4. Produktet $A^T B$ er ikke definert.
5. Ingen av de andre utsagnene er sanne.
7. La oss regne ut noen enkle matriseprodukter.

(a). Regn ut $\begin{pmatrix} 2 & 4 & 3 \\ 1 & -1 & -2 \\ -3 & -1 & 1 \end{pmatrix} \begin{pmatrix} 2 \\ 5 \\ 3 \end{pmatrix}$.

(b). Regn ut $\begin{pmatrix} 1 & 3 \\ 2 & 4 \end{pmatrix} \begin{pmatrix} -1 & 0 \\ 3 & 2 \end{pmatrix}$.

(c). Regn ut $\begin{pmatrix} -1 & 2 & 4 & 1 \\ 1 & 3 & 2 & 4 \end{pmatrix} \begin{pmatrix} 2 & 1 \\ 2 & 0 \\ 1 & 0 \\ 3 & 4 \end{pmatrix}$.

8. Vis egenskap 2.-4. i Proposisjon 2.12.
9. Vi skriver $A^n = AA \cdots A$, det vil si matrisen der vi ganger inn A n ganger. Regn ut A^3 og A^4 når $A = \begin{pmatrix} 1 & 4 \\ -2 & -1 \end{pmatrix}$.
- 10 (Speiling). La $\mathbf{F}$ være den lineære avbildningen fra $\mathbb{R}^2$ til $\mathbb{R}^2$ som speiler et punkt om linjen $y = x$. Finn matrisen til $\mathbf{F}$.
11. Vis at $(AB)^T = B^T A^T$, gitt at A har like mange søyler som B har rader.

KAPITTEL 3

Gausseliminasjon og redusert trappeform

Vi skal nå se nærmere på generelle systemer med m likninger og n ukjente:

$$\begin{array}{cccccc} a_{11}x_1 & +a_{12}x_2 & +\cdots & +a_{1n}x_n & = & b_1 \\ a_{21}x_1 & +a_{22}x_2 & +\cdots & +a_{2n}x_n & = & b_2 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \\ a_{m1}x_1 & +a_{m2}x_2 & +\cdots & +a_{mn}x_n & = & b_m \end{array}$$

Gausseliminasjon løser et slikt system ved systematisk å eliminere variablene fra venstre mot høyre. For å utdype dette, anta at variablene $x_1, \dots, x_n$ allerede er eliminert. Det er nå 2 muligheter

1. x_{n+1} forekommer ikke i de gjenværende likningene. Da er x_{n+1} allerede eliminert, og vi går videre til eliminering av x_{n+2} .
2. x_{n+1} forekommer i en av de gjenværende likningene. Gjør da følgende:
 - (a) Ta en av likningene der x_{n+1} forekommer, og flytt denne slik at den står øverst blant de gjenværende likningene.
 - (b) Skaler denne likningen slik at det står et ettall foran x_{n+1} . Vi skal kalle dette for en *ledende ener* (det er ingen bidrag til venstre for x_{n+1} i denne likningen).
 - (c) Legg til multiple av denne likningen til de andre likningene slik at x_{n+1} blir eliminert der. Vi har nå eliminert x_{n+1} , og går videre til eliminering av x_{n+2} .

Noen konkrete eksempler vil klargjøre dette.

Eksempel 3.1. Vi starter med systemet

$$\begin{array}{rrcr} & -8x_2 & +3x_3 & = & 6 \\ 4x_1 & -3x_2 & +2x_3 & = & 8 \\ x_1 & -2x_2 & +x_3 & = & 3 \end{array}$$

Først eliminerer vi x_1 . Siden x_1 ikke bidrar i første likning bytter vi om første og tredje likning og får

$$\begin{array}{rrcr} x_1 & -2x_2 & +x_3 & = & 3 \\ \textcolor{red}{4}x_1 & -3x_2 & +2x_3 & = & 8 \\ & -8x_2 & +3x_3 & = & 6 \end{array}$$

Vi må også sørge for at vi får en ledende ener i første likning, men det har vi allerede. Vi eliminerer nå x_1 i den andre likningen. Dette kan vi gjøre ved å trekke fra fire ganger den første likningen i den andre ligningen (vi markerer i rødt hva vi eliminerer). Da får vi

$$\begin{array}{rrcr} x_1 & -2x_2 & +x_3 & = & 3 \\ & 5x_2 & -2x_3 & = & -4 \\ & -8x_2 & +3x_3 & = & 6 \end{array}$$

Vi lager så en ledende ener i den andre likningen:

$$\begin{array}{rrcr} x_1 & -2x_2 & +x_3 & = & 3 \\ & x_2 & -\frac{2}{5}x_3 & = & -\frac{4}{5} \\ & \textcolor{red}{-8}x_2 & +3x_3 & = & 6 \end{array}$$

Vi eliminerer x_2 i den tredje likningen:

$$\begin{array}{rrcr} x_1 & -2x_2 & +x_3 & = & 3 \\ & x_2 & -\frac{2}{5}x_3 & = & -\frac{4}{5} \\ & & -\frac{1}{5}x_3 & = & -\frac{2}{5} \end{array}$$

Vi lager en ledende ener i den tredje likningen:

$$\begin{array}{rrcr} x_1 & -2x_2 & +\textcolor{red}{1}x_3 & = & 3 \\ & x_2 & -\textcolor{red}{\frac{2}{5}}x_3 & = & -\frac{4}{5} \\ & & x_3 & = & 2 \end{array}$$

Vi sier nå at systemet er på *trappeform*, som betyr at:

- hver likning er enten null eller har en ledende ener,

- hver likning begynner med minst en null-koeffisient mer enn likningen over.

Når vi har kommet til en trappeform er vi ferdige med å eliminere variable fra venstre mot høyre. Vi fortsetter nå med å eliminere variable fra høyre mot venstre: x_3 kan elimineres i de to første likningene ved å legge til multiple av den tredje likningen. Slår vi sammen disse to operasjonene får vi

$$\begin{array}{rcl} x_1 - 2x_2 & = & 1 \\ x_2 & = & 0 \\ x_3 & = & 2 \end{array}$$

Til slutt eliminerer vi x_2 i den første likningen med å legge til et multiplum av den andre likningen:

$$\begin{array}{rcl} x_1 & = & 1 \\ x_2 & = & 0 \\ x_3 & = & 2 \end{array}$$

Vi har altså den unike løsningen $\mathbf{x} = (x_1, x_2, x_3) = (1, 0, 2)$ av systemet. Systemet er nå på det vi kaller *redusert trappeform*: Det vil si, den er på trappeform, og i tillegg bidrar variablene med ledende ener bare i én likning. Disse kalles også for *pivotvariable*. Vi kaller også posisjonen til en ledende ener for et *pivotelement*.

For å forenkle det vi gjorde over kan det være lurt å samle koeffisientene og høyresiden i en matrise - uten at vi skriver ned likhetstegn og variabler. Matrisen $(A \quad \mathbf{b})$ kalles for den *utvidede matrisen* til systemet $A\mathbf{x} = \mathbf{b}$. For eksemplet over har vi at

$$A = \begin{pmatrix} 0 & -8 & 3 \\ 4 & -3 & 2 \\ 1 & -2 & 1 \end{pmatrix} \quad \mathbf{b} = \begin{pmatrix} 6 \\ 8 \\ 3 \end{pmatrix} \quad (A \quad \mathbf{b}) = \begin{pmatrix} 0 & -8 & 3 & 6 \\ 4 & -3 & 2 & 8 \\ 1 & -2 & 1 & 3 \end{pmatrix}$$

En likning kan vi nå tenke på som en rad, en søyle som koeffisientene til én variabel. Søylene i A med et pivotelement kalles *pivotsøylar*. På samme måte kalles radene i A med et pivotelement for *pivotrader*. Hvis vi lar $\sim$ betegne operasjone-

ne vi gjorde over, så kan vi skrive alt mer kompakt som:

$$\begin{aligned}
 \begin{pmatrix} 0 & -8 & 3 & 6 \\ 4 & -3 & 2 & 8 \\ 1 & -2 & 1 & 3 \end{pmatrix} &\sim \begin{pmatrix} 1 & -2 & 1 & 3 \\ 4 & -3 & 2 & 8 \\ 0 & -8 & 3 & 6 \end{pmatrix} \sim \begin{pmatrix} 1 & -2 & 1 & 3 \\ 0 & 5 & -2 & -4 \\ 0 & -8 & 3 & 6 \end{pmatrix} \\
 &\sim \begin{pmatrix} 1 & -2 & 1 & 3 \\ 0 & 1 & -\frac{2}{5} & -\frac{4}{5} \\ 0 & -8 & 3 & 6 \end{pmatrix} \sim \begin{pmatrix} 1 & -2 & 1 & 3 \\ 0 & 1 & -\frac{2}{5} & -\frac{4}{5} \\ 0 & 0 & -\frac{1}{5} & -\frac{34}{5} \end{pmatrix} \\
 &\sim \begin{pmatrix} 1 & -2 & 1 & 3 \\ 0 & 1 & -\frac{2}{5} & -\frac{4}{5} \\ 0 & 0 & 1 & 2 \end{pmatrix} \sim \begin{pmatrix} 1 & -2 & 0 & 1 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 2 \end{pmatrix} \\
 &\sim \begin{pmatrix} 1 & 0 & 0 & 1 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 2 \end{pmatrix} \tag{3.1}
 \end{aligned}$$

Etter de første fem operasjonene er matrisen på trappeform. Den siste matrisen er på redusert trappeform. Vi kan bringe en matrise på redusert trappeform med kommandoen `rref` i klassen `Matrix` i pakken `sympy`. Tester vi den på matrisen over får vi

```
>> C, cols = Matrix([[0,-8,3,6],[4,-3,2,8],[1,-2,1,3]]).rref()
C = Matrix([[1, 0, 0, 1], [0, 1, 0, 0], [0, 0, 1, 2]])
```

Vi ser at vi fikk samme matrise. Dette er ikke opplagt, siden det er mange forskjellige måter å bringe en matrise på redusert trappeform, og hver slik kan i prinsippet lede til forskjellige reduserte trappeformer. Vi skal senere se at den reduserte trappeformen er unik - uansett rekkefølge på operasjonene vil vi alltid ende opp med samme reduserte trappeform.

I Likning 3.1 indikerte vi ikke operasjonene vi gjorde. La oss etablere litt notasjon for de første operasjonene:

1. I den første overgangen byttet vi om rad 1 og 3. Dette skal vi skrive som $I \leftrightarrow III$.
2. I den andre overgangen trakk vi fra 4 ganger likning 1. Dette er det samme som å trekke fra 4 ganger rad 1 i rad 2, og dette skal vi skrive som $II - 4I$.
3. I den tredje overgangen ganget vi rad 2 med $1/5$. Dette skal vi skrive som $II \cdot (1/5)$.

Ved hjelp av denne notasjonen kan (3.1) presiseres som følger:

$$\begin{aligned}
 \begin{pmatrix} 0 & -8 & 3 & 6 \\ 4 & -3 & 2 & 8 \\ 1 & -2 & 1 & 3 \end{pmatrix} &\stackrel{I \leftrightarrow III}{\sim} \begin{pmatrix} 1 & -2 & 1 & 3 \\ 4 & -3 & 2 & 8 \\ 0 & -8 & 3 & 6 \end{pmatrix} \stackrel{II - 4I}{\sim} \begin{pmatrix} 1 & -2 & 1 & 3 \\ 0 & 5 & -2 & -4 \\ 0 & -8 & 3 & 6 \end{pmatrix} \\
 &\stackrel{II \cdot (1/5)}{\sim} \begin{pmatrix} 1 & -2 & 1 & 3 \\ 0 & 1 & -\frac{2}{5} & -\frac{4}{5} \\ 0 & -8 & 3 & 6 \end{pmatrix} \stackrel{III + 8II}{\sim} \begin{pmatrix} 1 & -2 & 1 & 3 \\ 0 & 1 & -\frac{2}{5} & -\frac{4}{5} \\ 0 & 0 & -\frac{1}{5} & -\frac{32}{5} \end{pmatrix} \\
 &\stackrel{III \cdot (-5)}{\sim} \begin{pmatrix} 1 & -2 & 1 & 3 \\ 0 & 1 & -\frac{2}{5} & -\frac{4}{5} \\ 0 & 0 & 1 & 2 \end{pmatrix} \stackrel{II + \frac{2}{5}III, I - III}{\sim} \begin{pmatrix} 1 & -2 & 0 & 1 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 2 \end{pmatrix} \\
 &\stackrel{I + 2II}{\sim} \begin{pmatrix} 1 & 0 & 0 & 1 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 2 \end{pmatrix}
 \end{aligned}$$

Systemet vårt kunne altså bringes til redusert trappeform ved hjelp av tre typer operasjoner:

1. legge et multiplum av en likning til en annen,
2. gange en likning med en konstant $\neq 0$,
3. bytte om to likninger.

Disse kaller vi *elementære radoperasjoner*. De kan enkelt implementeres:

```

A[j,:] = A[j,:] + c*A[i,:]
A[i,:] = c*A[i,:]
A[[i,j],:] = A[[j,i],:]

```

Spesielt kan de tre første radoperasjonene fra Eksempel 3.1 programmeres slik

```

A=np.array([[0,-8,3,6],[4,-3,2,8],[1,-2,1,3]])
A[[0,2],:] = A[[2,0],:]
A[1,:] = A[1,:] - 4*A[0,:]
A[1,:] = A[1,:]/5

```

I praksis gjøres ikke Gausseliminasjon slik: Programmet selv finner ut hvilke radoperasjoner som skal utføres (se Oppgave 12).

Vi sier også at A og B er *radekvivalente* hvis vi kan transformere A til B ved hjelp av en serie elementære radoperasjoner. Fra diskusjonen over har vi at

$$\begin{pmatrix} 0 & -8 & 3 & 6 \\ 4 & -3 & 2 & 8 \\ 1 & -2 & 1 & 3 \end{pmatrix} \sim \begin{pmatrix} 1 & 0 & 0 & 1 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 2 \end{pmatrix}.$$

Enhver matrise er radekvivalent med en matrise på reduserte trappeform. Elementære radoperasjoner kan også kan uttrykkes ved matriser:

Proposisjon 3.2. Anta B fremkommer fra $m \times n$ -matrisen A ved en elementær radoperasjon. Da er $B = EA$ der E er matrisen du får ved å anvende samme radoperasjon på identitetsmatrisen I_m .

Bevis. La $F(A)$ være matrisen vi får ved å anvende radoperasjonen på matrisen A . Anvender vi radoperasjonen søylevis får vi

$$F(A) = (F(\mathbf{a}_1) \quad F(\mathbf{a}_2) \quad \cdots \quad F(\mathbf{a}_n)).$$

Videre er $\mathbf{a} \rightarrow F(\mathbf{a})$ en lineær avbildning fra $\mathbb{R}^m$ til $\mathbb{R}^m$ med matrise

$$(F(\mathbf{e}_1) \quad F(\mathbf{e}_2) \quad \cdots \quad F(\mathbf{e}_m)) = F(I_m).$$

Det følger at

$$F(A) = (F(I_m)\mathbf{a}_1 \quad F(I_m)\mathbf{a}_2 \quad \cdots \quad F(I_m)\mathbf{a}_n) = F(I_m)(\mathbf{a}_1 \quad \mathbf{a}_2 \quad \cdots \quad \mathbf{a}_n) = F(I_m)A,$$

som var det vi skulle vise. ■

En matrise som fremkommer ved en elementær radoperasjon på identitetsmatrisen kalles *elementær*. Følgende matriser er elementære, og svarer til radoperasjonene der vi legger 2 ganger rad 2 til rad 1, ganger rad 3 med 3, og bytter om rad 1 og 2, respektive:

$$E_1 = \begin{pmatrix} 1 & 2 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix} \quad E_2 = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 3 \end{pmatrix} \quad E_3 = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 0 & 1 \\ 0 & 1 & 0 \end{pmatrix} \quad (3.2)$$

Radoperasjoner kan reverseres i den forstand at vi kan finne en radoperasjon som reversere det den første gjorde. De reverse radoperasjonene til (3.2) er å legge -2 ganger rad 2 til rad 1, gange rad 3 med $1/3$, og bytte om rad 1 og 2, respektive. De elementære matrisene for disse er

$$F_1 = \begin{pmatrix} 1 & -2 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix} \quad F_2 = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1/3 \end{pmatrix} \quad F_3 = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 0 & 1 \\ 0 & 1 & 0 \end{pmatrix}$$

At F_i reverserer E_i betyr at $F_1E_1 = I_n$, $F_2E_2 = I_n$, $F_3E_3 = I_n$, som også kan sjekkes ved å gange sammen matrisene. Fra Proposisjon 3.2 får vi følgende.

Proposisjon 3.3. For enhver matrise A kan vi skrive

$$C = E_k \cdots E_2 E_1 A$$

der C er en matrise på redusert trappeform, og der $E_1, E_2, \dots, E_k$ er elementære matriser.

Vi har også følgende resultat.

Proposisjon 3.4. La et system fremkomme fra et annet ved en elementær radoperasjon. Da har de to systemene de samme løsningene.

Bevis. La E være den elementære matrisen som svarer til radoperasjonen, og F den elementære matrisen som svarer til den reverse radoperasjonen. Når vi anvender en elementær radoperasjon på det opprinnelige systemet $A\mathbf{x} = \mathbf{b}$ så får vi det nye systemet $E A \mathbf{x} = E \mathbf{b}$.

- Hvis $A\mathbf{x} = \mathbf{b}$ så er det klart at også $E A \mathbf{x} = E \mathbf{b}$, slik at $\mathbf{x}$ også løser det nye systemet.
- Hvis $E A \mathbf{x} = E \mathbf{b}$ så får vi ved å gange med F på begge sider at $A\mathbf{x} = \mathbf{b}$, slik at $\mathbf{x}$ også løser det opprinnelige systemet.

Det følger at det opprinnelige og det nye systemet har de samme løsningene. ■

Over radreduserte vi et system som hadde en unik løsning. La oss gjøre det samme for systemer med ingen, eller uendelig mange løsninger.

Eksempel 3.5. La oss se på systemet

$$\begin{array}{rrcr} x_1 & +x_2 & +x_3 & = & 4 \\ 2x_1 & & -x_3 & = & 2. \end{array}$$

Vi skriver ned den utvidede matrisen, og gjør elementære radoperasjoner:

$$\begin{pmatrix} 1 & 1 & 1 & 4 \\ 2 & 0 & -1 & 2 \end{pmatrix} \xrightarrow{II-2I} \begin{pmatrix} 1 & 1 & 1 & 4 \\ 0 & -2 & -3 & -6 \end{pmatrix} \xrightarrow{II \cdot (-1/2)} \begin{pmatrix} 1 & 1 & 1 & 4 \\ 0 & 1 & 3/2 & 3 \end{pmatrix} \xrightarrow{I-\widetilde{II}} \begin{pmatrix} 1 & 0 & -1/2 & 1 \\ 0 & 1 & 3/2 & 3 \end{pmatrix}$$

Igjen klarte vi å få systemet på redusert trappeform. På likningsform sier denne

$$\begin{array}{rcl} x_1 & -\frac{1}{2}x_3 & = 1 \\ x_2 & +\frac{3}{2}x_3 & = 3 \end{array}$$

som også kan skrives $x_1 = 1 + \frac{1}{2}x_3$, $x_2 = 3 - \frac{3}{2}x_3$. Vi ser at x_3 kan velges fritt, og at en generell løsning kan skrives som

$$\mathbf{x} = \begin{pmatrix} x_1 \\ x_2 \\ x_3 \end{pmatrix} = \begin{pmatrix} 1 + \frac{1}{2}x_3 \\ 3 - \frac{3}{2}x_3 \\ x_3 \end{pmatrix} = \begin{pmatrix} 1 \\ 3 \\ 0 \end{pmatrix} + x_3 \begin{pmatrix} \frac{1}{2} \\ -\frac{3}{2} \\ 1 \end{pmatrix}$$

Vi ser nå at det er uendelig mange løsninger. x_1 og x_2 er pivotvariable, og vi ser at disse kan uttrykkes med x_3 . x_3 er ikke en pivotvariabel siden søyle 3 ikke har en ledende ener. x_3 kan settes til hva som helst, og kalles derfor en *fri variabel*.

Eksempel 3.6. La oss se på systemet

$$\begin{array}{rcl} x_1 & +x_2 & = 2 \\ 3x_1 & +3x_2 & = 5 \end{array}$$

Radreduserer vi den utvidede matrisen får vi

$$\begin{pmatrix} 1 & 1 & 2 \\ 3 & 3 & 5 \end{pmatrix} \xrightarrow{II-3I} \begin{pmatrix} 1 & 1 & 2 \\ 0 & 0 & -1 \end{pmatrix} \xrightarrow{II \cdot (-1)} \begin{pmatrix} 1 & 1 & 2 \\ 0 & 0 & 1 \end{pmatrix} \xrightarrow{I-2II} \begin{pmatrix} 1 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix}.$$

Igjen klarte vi å få systemet på redusert trappeform. Legg merke til her at, når vi eliminerte x_1 , så ble også x_2 eliminert. På likningsform sier dette at

$$\begin{array}{rcl} x_1 & +x_2 & = 0 \\ 0 & & = 1 \end{array}$$

$0 = 1$ er umulig, slik at vi ikke får noen løsninger.

De tre eksemplene over har illustrert tre muligheter: Et system kan ha

1. en unik løsning, som i Eksempel 3.1. Her hadde vi kun pivotvariable, og ikke noe pivotelement i siste søyle.
2. uendelig mange løsninger, som i Eksempel 3.5. Her hadde vi en fri variabel, og ikke noe pivotelement i siste søyle.
3. ingen løsninger, som i Eksempel 3.6. Her hadde vi et pivotelement i siste søyle.

Videre fant vi en redusert trappeform i alle eksemplene. Disse observasjonene viser seg å holde mer generelt:

Proposisjon 3.7. *Ethvert system kan bringes på redusert trappeform, og vi har tre muligheter:*

1. *systemet har ingen løsning (siste søyle er en pivotsøyle),*
2. *systemet har løsning(er) (siste søyle er ikke en pivotsøyle). Da gjelder at*
 - (a) *når vi har frie variable har systemet uendelig mange løsninger (ethvert valg av frie variable gir en løsning),*
 - (b) *når vi ikke har frie variable (alle søyler, unntatt den siste, er pivotsøyler) har systemet en unik løsning.*

Fra diskusjonen over har vi også følgende:

Teorem 3.8. *Følgende er ekvivalent for en $n \times n$ -matrise A :*

1. *A er radekvivalent med I_n .*
2. *$A\mathbf{x} = \mathbf{0}$ har kun løsningen $\mathbf{x} = \mathbf{0}$.*
3. *$A\mathbf{x} = \mathbf{b}$ har en entydig løsning for alle høyresider $\mathbf{b}$.*
4. *A kan skrives som et produkt av elementære matriser.*

Vi skal senere utvide dette resultatet med flere ekvivalenser.

Bevis. 1. $\Rightarrow$ 3.: Anta at A kan radreduseres til identitetsmatrisen. Vi kan løse $A\mathbf{x} = \mathbf{b}$ ved å radredusere den utvidede matrisen $(A \mid \mathbf{b})$ med de samme radoperasjonene. Da får vi til slutt likninger på formen $x_1 = b'_1, x_2 = b'_2, \dots, x_n = b'_n$, slik at systemet har en unik løsning for alle høyresider $\mathbf{b}$.

3. $\Rightarrow$ 2. er opplagt.

2. $\Rightarrow$ 1.: Vi løser $A\mathbf{x} = \mathbf{0}$ ved å radredusere den utvidede matrisen $(A \mid \mathbf{0})$. Siden systemet bare har en løsning så kan vi ikke ha noen frie variable. Men da er alle søyler i A pivotsøyler. Siden A er kvadratisk så er også alle rader pivotrader. Det følger at A er radekvivalent med I_n .

1. $\Rightarrow$ 4.: At $A \sim I_n$ betyr at vi kan skrive $I_n = E_k \cdots E_1 A$, der E_i er elementære matriser. La F_i være den elementære matrisen som svarer til den reverse radoperasjonen til E_i . Ganger vi med F_k til venstre på begge sider får vi at $F_k = E_{k-1} \cdots E_1 A$. Ganger vi så med F_{k-1} til venstre på begge sider får vi at $F_{k-1} F_k =$

$E_{k-2} \cdots E_1 A$. Fortsetter vi slik får vi til slutt at $A = F_1 \cdots F_k$, slik at A kan skrives om et produkt av elementære matriser.

4. $\Rightarrow$ 1.: Hvis $A = F_1 \cdots F_k$ så kan vi som over vise at $I_n = E_k \cdots E_1 A$ (der igjen F_i representerer den inverse radoperasjonen til E_i), slik at $A \sim I_n$. ■

3.1 Gausseliminering og kompleksitet

Anta vi har et system med n likninger og n ukjente. Hvor mange regneoperasjoner (addisjoner, multiplikasjoner, etc.) trenger vi for å løse systemet, det vil si få det på redusert trappeform? Anta for enkelhets skyld at koeffisientmatrisen A er radekvivalent med identitetsmatrisen. La oss også anta at det ikke koster noe (med tanke på antall regneoperasjoner) å gjøre radbytter.

- I Gausseliminering lager vi oss en ledende ener i hver rad ved å dele med den første ikke-nullen. Dette krever

$$(n-1) + (n-2) + \cdots + 1 = (n-1)n/2$$

divisjoner.

- I første del av Gausseliminering nuller vi ut elementene under diagonalen, søyle for søyle. Under diagonalelement k i søyle k er det $n-k$ rader. For hver av disse ganger vi rad k med et tall (dette svarer til $n-k$ multiplikasjoner, siden raden starter med $k-1$ nuller), og legger dette til raden (dette svarer til $n-k$ addisjoner). Til sammen får vi $(n-k)((n-k) + (n-k)) = 2(n-k)^2$ operasjoner her. Summerer vi over alle k får vi

$$2 \sum_{k=1}^{n-1} (n-k)^2 = 2 \sum_{k=0}^n k^2 = n(n+1)(2n+1)/6 \approx 2n^3/3,$$

siden $n^3/3$ er det dominerende leddet når vi ganger ut $n(n+1)(2n+1)/6$ (som er formelen for summen av de n første kvadrattallene). Grunnen til at vi bare er interessert i det dominerende leddet er at vi vanligvis er interessert i hvor fort antall regneoperasjoner vokser når n vokser - vi er ikke interessert i det eksakte uttrykk for antall operasjoner, som typisk er gitt ved et styggere uttrykk.

- I andre del av Gausseliminering nuller vi ut elementer over diagonalen, søyle for søyle. Vi trenger ingen ekstra operasjoner for dette.

Siden $(n-1)n/2 \approx n^2/2$ er dominert av $2n^3/3$, så har vi totalt i størrelsesorden $2n^3/3$ regneoperasjoner. Vi har vist følgende.

Proposisjon 3.9 (Regneoperasjoner Gausseliminering). Anta $A \sim I_n$. Det krever omtrent $2n^3/3$ regneoperasjoner for å bringe A på redusert trappeform.

3.2 Simultan løsning av likningssystemer

Anta vi skal løse to systemer med samme koeffisientmatrise, $A\mathbf{x} = \mathbf{b}_1$ og $A\mathbf{x} = \mathbf{b}_2$. Vi radreduserer A til redusert trappeform, C . Vi gjør de samme radoperasjonene på $\mathbf{b}_1$ og $\mathbf{b}_2$, og får $\mathbf{b}'_1$ og $\mathbf{b}'_2$. Dette svarer til at

$$(A \quad \mathbf{b}_1 \quad \mathbf{b}_2) \sim (C \quad \mathbf{b}'_1 \quad \mathbf{b}'_2),$$

siden det å radredusere A , $\mathbf{b}_1$, og $\mathbf{b}_2$ separat, også kan gjøres ved å radredusere $(A \quad \mathbf{b}_1 \quad \mathbf{b}_2)$ som en helhet. Fra $(C \quad \mathbf{b}'_1 \quad \mathbf{b}'_2)$ kan vi lese ut løsningene til de to systemene. La oss se på et eksempel.

Eksempel 3.10. I Eksempel 3.1 løste vi systemet $A\mathbf{x} = \mathbf{b}_1$ der $\mathbf{b}_1 = (6, 8, 3)$. Ved å bruke radoperasjonene fra dette eksemplet kan vi simultant løse systemet $A\mathbf{x} = \mathbf{b}_2$ der $\mathbf{b}_2 = (1, 8, 2)$ som følger:

$$\begin{aligned} (A \quad \mathbf{b}_1 \quad \mathbf{b}_2) &= \begin{pmatrix} 0 & -8 & 3 & 6 & 1 \\ 4 & -3 & 2 & 8 & 8 \\ 1 & -2 & 1 & 3 & 2 \end{pmatrix} \xrightarrow{I \leftrightarrow III} \begin{pmatrix} 1 & -2 & 1 & 3 & 2 \\ 4 & -3 & 2 & 8 & 8 \\ 0 & -8 & 3 & 6 & 1 \end{pmatrix} \xrightarrow{II \sim 4I} \begin{pmatrix} 1 & -2 & 1 & 3 & 2 \\ 0 & 5 & -2 & -4 & 0 \\ 0 & -8 & 3 & 6 & 1 \end{pmatrix} \\ &\xrightarrow{II \cdot (1/5)} \begin{pmatrix} 1 & -2 & 1 & 3 & 2 \\ 0 & 1 & -2/5 & -4/5 & 0 \\ 0 & -8 & 3 & 6 & 1 \end{pmatrix} \xrightarrow{III \sim 8II} \begin{pmatrix} 1 & -2 & 1 & 3 & 2 \\ 0 & 1 & -2/5 & -4/5 & 0 \\ 0 & 0 & -1/5 & -3/5 & 1 \end{pmatrix} \\ &\xrightarrow{III \cdot (-5)} \begin{pmatrix} 1 & -2 & 1 & 3 & 2 \\ 0 & 1 & -2/5 & -4/5 & 0 \\ 0 & 0 & 1 & 2 & -5 \end{pmatrix} \xrightarrow{II + \frac{2}{5}III, I - III} \begin{pmatrix} 1 & -2 & 0 & 1 & 7 \\ 0 & 1 & 0 & 0 & -2 \\ 0 & 0 & 1 & 2 & -5 \end{pmatrix} \\ &\xrightarrow{I + 2II} \begin{pmatrix} 1 & 0 & 0 & 1 & 3 \\ 0 & 1 & 0 & 0 & -2 \\ 0 & 0 & 1 & 2 & -5 \end{pmatrix} \end{aligned}$$

Det følger at $A\mathbf{x}_1 = \mathbf{b}_1$ som før har løsningen $\mathbf{x}_1 = (1, 0, 2)$, men også at $A\mathbf{x}_2 = \mathbf{b}_2$ har løsningen $\mathbf{x}_2 = (3, -2, -5)$.

Oppgaver for Kapittel 3

1. Finn løsningen(e) til følgende systemer.

(a).

$$\begin{array}{rrcr} 2x_1 & -7x_2 & +x_3 & = & -21 \\ x_1 & -4x_2 & -3x_3 & = & -23 \\ x_1 & -x_2 & +2x_3 & = & 4 \end{array}$$

(b).

$$\begin{array}{rrcr} x_1 & -2x_2 & & = & 3 \\ x_1 & -3x_2 & & = & 4 \\ -x_1 & +5x_2 & & = & 2 \end{array}$$

(c).

$$\begin{array}{rrcr} x_1 & -2x_2 & +2x_3 & = & 2 \\ 2x_1 & +3x_2 & -4x_3 & = & 2 \end{array}$$

2. Vi er gitt matrisen $A = \begin{pmatrix} 1 & -1 & 3 \\ 2 & 1 & 4 \\ -4 & 7 & 2 \end{pmatrix}$. Finn løsningene av systemene $A\mathbf{x} = \mathbf{b}_1$ og $A\mathbf{x} = \mathbf{b}_2$ der $\mathbf{b}_1 = (-8, -7, -7)$ og $\mathbf{b}_2 = (9, 5, -17)$.

3. Finn et annet sett med radoperasjoner som bringer matrisen fra Eksempel 3.1 på redusert trappeform. Verifiser at du får samme reduserte trappeform som i eksemplet.

4 (Midtveiseksamen 2024). Hvilken påstand er sann?

1. I en matrise på redusert trappeform er det alltid kun ett ikke-null element i hver kolonne
2. I en matrise på redusert trappeform er det kun ett ikke-null element i hver pivotkolonne
3. Den reduserte trappeformen er ikke unik.
4. En $m \times n$ -matrise har alltid n pivotkolonner
5. En $m \times n$ -matrise har alltid n rader med pivotelement i seg.

5 (Midtveiseksamen 2024). Hva kan vi si om løsningene til følgende lignings-system?

$$\begin{array}{l} 3x_1 = 7 - x_2 \\ 2x_1 - 3x_2 = -3 - 4x_3 \\ 2x_2 - 4 = 2x_3. \end{array}$$

1. Systemet har ingen løsning.
2. Systemet har uendelig mange løsninger.
3. Systemet har én løsning, nemlig $\mathbf{x} = \begin{pmatrix} 4 \\ -5 \\ -13/2 \end{pmatrix}$.
4. Systemet har én løsning, nemlig $\mathbf{x} = \begin{pmatrix} 6 \\ -11 \\ -12 \end{pmatrix}$.
5. Systemet har én løsning, nemlig $\mathbf{x} = \begin{pmatrix} 2 \\ 1 \\ -1 \end{pmatrix}$.

6 (Midtveiseksamen 2024). For hvilken verdi av a har følgende system en løsning?

$$2x + 3y + 4z = 5$$

$$2x + 5y + 4z = 7$$

$$4x + 7y + 8z = a$$

7 (Midtveiseksamen 2024). Hva er den reduserte trappeformen til følgende matrise?

$$A = \begin{pmatrix} 1 & 2 & -2 & 2 \\ 2 & 4 & -1 & 16 \\ 3 & 6 & -5 & 11 \end{pmatrix}$$

8 (Midtveiseksamen 2024). Hva er pivotkolonnene til følgende matrise?

$$\begin{pmatrix} 1 & 2 & 4 \\ 2 & 4 & 9 \\ -2 & -4 & 1 \\ -3 & -6 & 0 \end{pmatrix}$$

9 (Midtveiseksamen 2024). La $A \in \mathbb{R}^{3 \times 3}$. Anta at systemet $A\mathbf{x} = \begin{pmatrix} 1 \\ 0 \\ 2 \end{pmatrix}$ har løsningen

$\mathbf{x} = \begin{pmatrix} 5 \\ 6 \\ -3 \end{pmatrix}$, og at systemet $A\mathbf{y} = \begin{pmatrix} 0 \\ 4 \\ 0 \end{pmatrix}$ har løsningen $\mathbf{y} = \begin{pmatrix} 3 \\ -1 \\ 2 \end{pmatrix}$. Hva blir da løsningen av systemet $A\mathbf{z} = \begin{pmatrix} 2 \\ 8 \\ 4 \end{pmatrix}$?

10. Vi programmerte over de tre første radoperasjonene i Eksempel 3.1. Skriv kode som gjør de siste radoperasjonene også.

11. Vis at systemet $A\mathbf{x} = \mathbf{b}$ (med m likninger og n ukjente) har en løsning hvis og bare hvis det ikke finnes en $\mathbf{c} \in \mathbb{R}^m$ slik at $\mathbf{c}^T A = \mathbf{0}$ og $\mathbf{c}^T \mathbf{b} = 1$. Dette resultatet kalles også *Farkas lemma*.

12. Vi skal se på følgende kode for generell Gausseliminasjon:

```
import numpy as np

def gaussel(A):
    m, n = np.shape(A)
    piv = np.zeros(m).astype(int)
    i = 0
    j = 0
    while i < m and j < n:
        k = i
        while k < m and abs(A[k,j]) < 10**(-6):
            k += 1
        if k == m:
            j += 1
        else:
            if k > i:
                A[[k,i]] = A[[i,k]]
            A[i,j:] = A[i,j:]/A[i,j]
            A[(i+1):,j:] = A[(i+1):,j:] - A[(i+1):,j].reshape((-1,1)) @ A[i,j:].reshape((1,-1))
            piv[i] = j
            i += 1
            j += 1
    i -= 1
    while i > 0:
        j = piv[i]
        A[:i,j:] = A[:i,j:] - A[:i,j].reshape((-1,1)) @ A[i,j:].reshape((1,-1))
        i -= 1
    return A
```

Tanken i koden er at (i, j) skal representere et pivotelement i matrisen.

(a). Hvilke linjer i koden sørger for å nulle ut elementer under pivotelementene? Hvilke linjer nuller ut elementer over pivotelementene?

(b). Kan du tenke deg til hvorfor det er tatt med betingelsen $|a_{kj}| < 10^{-6}$ i den andre `while`-løkka? Forklar resten av koden

KAPITTEL 4

Den inverse matrisen

Vi har nå lært å løse et system på formen $A\mathbf{x} = \mathbf{b}$ ved Gausseliminering. Hvis A er kvadratisk og tilfredsstiller 1.-4. i Teorem 3.8, så kan vi se på avbildningen $\mathbf{b} \rightarrow \mathbf{x}$, det vil si avbildningen som sender en gitt høyreside på den unike løsningen $\mathbf{x}$. Det er lett å vise at denne avbildningen er lineær, slik at den er gitt ved en matrise. Under skal vi kalle denne matrisen for A^{-1} . Hvis vi har regnet ut A^{-1} så kan vi altså løse ethvert system $A\mathbf{x} = \mathbf{b}$ ved hjelp av matrisemultiplikasjonen $A^{-1}\mathbf{b}$.

I dette kapitlet skal vi definere den inverse matrisen litt mer presist. Vi skal starte med å vise følgende:

Lemma 4.1. *La A være en kvadratisk matrise. A tilfredsstiller 1.-4. i Teorem 3.8 hvis og bare hvis det finnes en matrise B slik at $AB = I_n$.*

Bevis. Anta at $A = E_k \cdots E_1$ tilfredsstiller 1.-4. i Teorem 3.8. Hvis F_i er elementære matriser som representerer de reverse operasjonene så har vi at

$$E_k \cdots E_1 F_1 \cdots F_k = I_n.$$

Matrisen $B = F_1 \cdots F_k$ tilfredsstiller altså $AB = I_n$. Omvendt, anta A og B er kvadratiske matriser slik at $AB = I_n$. Anta at $B\mathbf{x} = \mathbf{0}$ for en $\mathbf{x} \neq \mathbf{0}$. Da er $AB\mathbf{x} = \mathbf{0}$ også, som strider mot at $AB = I_n$. $B\mathbf{x} = \mathbf{0}$ har dermed kun løsningen $\mathbf{x} = \mathbf{0}$, slik at B oppfyller 1.-4. i Teorem 3.8. Men da kan B skrives som et produkt av elementære matriser, $B = F_1 \cdots F_k$. Fra $AB = I_n$ følger da at $A = E_k \cdots E_1$, der E_i representerer de reverse radoperasjonene. A tilfredsstiller altså 1.-4. i Teorem 3.8 også. ■

Når $AB = I_n$ sier vi at B er en *høyreinvers* til A . Lemmaet over gjør at vi kan legge til at A har en høyreinvers som et punkt i Teorem 3.8. Vi skal gjøre dette, men med en liten modifikasjon: Vi sier at B er en *venstreinvert* til A hvis $BA = I_n$. Anta A har en høyreinvers B . Fra resonnementet over følger at vi kan skrive $A = E_k \cdots E_1$ og $B = F_1 \cdots F_k$, der E_i er elementære, og der F_i er deres reverse motsatser. Men da er

$$BA = F_1 \cdots F_k E_k \cdots E_1 = I_n,$$

slik at B også er en venstreinvert til A . Det følger også at en venstre/høyreinvers er unik. Vi har vist følgende

Proposisjon 4.2. *Anta B er en høyreinvers til A . Da er også B en venstreinvert til A . En venstreinvert/høyreinvers til A er unik.*

Dette gjør at vi ikke trenger å skille mellom høyre- og venstre-inverser, og det blir naturlig å definere at A er *inverterbar* hvis det finnes en matrise B slik at $AB = BA = I_n$. B kalles den *inverse matrisen* til A , og vi skriver A^{-1} for denne.

Noen egenskaper for den inverse matrisen følger umiddelbart fra definisjonen. Siden $A^{-1}A = I_n$ så må spesielt $(A^{-1})^{-1} = A$. Videre, hvis $s \neq 0$ så er

$$(sA)(s^{-1}A^{-1}) = AA^{-1} = I_n,$$

slik at $(sA)^{-1} = s^{-1}A^{-1}$. I oppgavene til dette kapitlet blir du bedt om å vise andre egenskaper og regneregler for den inverse matrisen.

La oss nå utvide Teorem 3.8 med en ekvivalens om inverterbarhet, og samtidig gi den et navn.

Teorem 4.3 (Teoremet for inverterbare matriser). *Følgende er ekvivalent for en $n \times n$ -matrise A :*

1. A er radekvivalent med I_n .
2. $A\mathbf{x} = \mathbf{0}$ har kun løsningen $\mathbf{x} = \mathbf{0}$.
3. $A\mathbf{x} = \mathbf{b}$ har en entydig løsning for alle høyresider $\mathbf{b}$.
4. A kan skrives som et produkt av elementære matriser.
5. A er inverterbar.

Fra diskusjonen over er det også klart at hvis $E_1, E_2, \dots, E_k$ er elementære matriser som radreduserer A til identitetsmatrisen (slik at $E_k \cdots E_1 A = I_n$), så er

$$A = F_1 \cdots F_k \qquad A^{-1} = E_k \cdots E_1$$

der $F_i = E_i^{-1}$. Med andre ord:

- A^{-1} får vi ved å gange sammen de elementære matrisene,
- A får vi ved å gange sammen deres inverser i omvendt rekkefølge.

For å regne ut A^{-1} er det nyttig å anvende radoperasjonene ikke bare på A , men på den utvidede matrisen $(A \mid I_n)$. Da får vi

$$E_k \cdots E_1 (A \mid I_n) = (I_n \mid E_k \cdots E_1) = (I_n \mid A^{-1}).$$

Fra resultatet kan man lese ut A^{-1} fra de siste søylene. En alternativ måte å tolke dette på er at vi, når vi radreduserer $(A \mid I_n)$, simultant løser systemene

$$A\mathbf{x}_1 = \mathbf{e}_1 \qquad A\mathbf{x}_2 = \mathbf{e}_2 \qquad \dots \qquad A\mathbf{x}_n = \mathbf{e}_n,$$

og siden $A\mathbf{x}_i = \mathbf{e}_i$ medfører at $A^{-1}\mathbf{e}_i = \mathbf{x}_i$, så vil søyle i i A^{-1} være $\mathbf{x}_i$. La oss oppsummere.

Proposisjon 4.4. Anta A er inverterbar. Den reduserte trappeformen til $(A \mid I_n)$ er lik $(I_n \mid A^{-1})$.

Eksempel 4.5. La oss finne den inverse matrisen til $A = \begin{pmatrix} 2 & 3 & 1 \\ 1 & 4 & 1 \\ 2 & 4 & 2 \end{pmatrix}$:

$$\begin{aligned}
 (A \quad I_n) &= \begin{pmatrix} 2 & 3 & 1 & 1 & 0 & 0 \\ 1 & 4 & 1 & 0 & 1 & 0 \\ 2 & 4 & 2 & 0 & 0 & 1 \end{pmatrix} \xrightarrow{I \leftrightarrow II} \begin{pmatrix} 1 & 4 & 1 & 0 & 1 & 0 \\ 2 & 3 & 1 & 1 & 0 & 0 \\ 2 & 4 & 2 & 0 & 0 & 1 \end{pmatrix} \\
 &\xrightarrow{II \sim 2I} \begin{pmatrix} 1 & 4 & 1 & 0 & 1 & 0 \\ 0 & -5 & -1 & 1 & -2 & 0 \\ 2 & 4 & 2 & 0 & 0 & 1 \end{pmatrix} \xrightarrow{III \sim 2I} \begin{pmatrix} 1 & 4 & 1 & 0 & 1 & 0 \\ 0 & -5 & -1 & 1 & -2 & 0 \\ 0 & -4 & 0 & 0 & -2 & 1 \end{pmatrix} \\
 &\xrightarrow{II \leftrightarrow III} \begin{pmatrix} 1 & 4 & 1 & 0 & 1 & 0 \\ 0 & -4 & 0 & 0 & -2 & 1 \\ 0 & -5 & -1 & 1 & -2 & 0 \end{pmatrix} \xrightarrow{II / (-4)} \begin{pmatrix} 1 & 4 & 1 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 & 1/2 & -1/4 \\ 0 & -5 & -1 & 1 & -2 & 0 \end{pmatrix} \\
 &\xrightarrow{III + 5II} \begin{pmatrix} 1 & 4 & 1 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 & 1/2 & -1/4 \\ 0 & 0 & -1 & 1 & 1/2 & -5/4 \end{pmatrix} \xrightarrow{III (-1)} \begin{pmatrix} 1 & 4 & 1 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 & 1/2 & -1/4 \\ 0 & 0 & 1 & -1 & -1/2 & 5/4 \end{pmatrix} \\
 &\xrightarrow{I - 4II} \begin{pmatrix} 1 & 4 & 0 & 1 & 3/2 & -5/4 \\ 0 & 1 & 0 & 0 & 1/2 & -1/4 \\ 0 & 0 & 1 & -1 & -1/2 & 5/4 \end{pmatrix} \xrightarrow{I - 4II} \begin{pmatrix} 1 & 0 & 0 & 1 & -1/2 & -1/4 \\ 0 & 1 & 0 & 0 & 1/2 & -1/4 \\ 0 & 0 & 1 & -1 & -1/2 & 5/4 \end{pmatrix}.
 \end{aligned}$$

Vi leser ut de siste tre søylene som $A^{-1} = \begin{pmatrix} 1 & -1/2 & -1/4 \\ 0 & 1/2 & -1/4 \\ -1 & -1/2 & 5/4 \end{pmatrix}$. Vi kan sjekke svaret med den innebygde funksjonen for å finne inversen:

```
>> np.linalg.inv([[2,3,1],[1,4,1],[2,4,2]])
[[ 1.   -0.5  -0.25]
 [ 0.    0.5  -0.25]
 [-1.   -0.5   1.25]]
```

Vi kan også sjekke svaret ved å regne ut dette symbolsk:

```
>> sym.Matrix([[2,3,1],[1,4,1],[2,4,2]]).inv()
Matrix([[1, -1/2, -1/4], [0, 1/2, -1/4], [-1, -1/2, 5/4]])
```

Vi leser også ut de elementære matrisene

$$\begin{aligned}
 E_1 &= \begin{pmatrix} 0 & 1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 1 \end{pmatrix} & E_2 &= \begin{pmatrix} 1 & 0 & 0 \\ -2 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix} & E_3 &= \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ -2 & 0 & 1 \end{pmatrix} & E_4 &= \begin{pmatrix} 1 & 0 & 0 \\ 0 & 0 & 1 \\ 0 & 1 & 0 \end{pmatrix} \\
 E_5 &= \begin{pmatrix} 1 & 0 & 0 \\ 0 & -1/4 & 0 \\ 0 & 0 & 1 \end{pmatrix} & E_6 &= \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 5 & 0 & 1 \end{pmatrix} & E_7 &= \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & -1 \end{pmatrix} & E_8 &= \begin{pmatrix} 1 & 0 & -1 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix} \\
 E_9 &= \begin{pmatrix} 1 & -4 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix}
 \end{aligned}$$

og

$$\begin{aligned}
 F_1 &= \begin{pmatrix} 0 & 1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 1 \end{pmatrix} & F_2 &= \begin{pmatrix} 1 & 0 & 0 \\ 2 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix} & F_3 &= \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 2 & 0 & 1 \end{pmatrix} & F_4 &= \begin{pmatrix} 1 & 0 & 0 \\ 0 & 0 & 1 \\ 0 & 1 & 0 \end{pmatrix} \\
 F_5 &= \begin{pmatrix} 1 & 0 & 0 \\ 0 & -4 & 0 \\ 0 & 0 & 1 \end{pmatrix} & F_6 &= \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ -5 & 0 & 1 \end{pmatrix} & F_7 &= \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & -1 \end{pmatrix} & F_8 &= \begin{pmatrix} 1 & 0 & 1 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix} \\
 F_9 &= \begin{pmatrix} 1 & 4 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix}
 \end{aligned}$$

Med disse har vi at $A^{-1} = E_9 \cdots E_1$, og $A = F_1 \cdots F_9$.

Eksempel 4.6. La oss se på systemet $A\mathbf{x} = \mathbf{b}$ der $A = \begin{pmatrix} 2 & 3 & 1 \\ 1 & 4 & 1 \\ 2 & 4 & 2 \end{pmatrix}$ fra forrige ek-

sempel, og der $\mathbf{b} = \begin{pmatrix} 19 \\ 21 \\ 26 \end{pmatrix}$. Vi regnet ut A^{-1} over. Bruker vi denne ser vi at løsningen på systemet blir

$$\mathbf{x} = A^{-1}\mathbf{b} = \begin{pmatrix} 1 & -1/2 & -1/4 \\ 0 & 1/2 & -1/4 \\ -1 & -1/2 & 5/4 \end{pmatrix} \begin{pmatrix} 19 \\ 21 \\ 26 \end{pmatrix} = \begin{pmatrix} 19 - 21/2 - 13/2 \\ 21/2 - 13/2 \\ -19 - 21/2 + 65/2 \end{pmatrix} = \begin{pmatrix} 2 \\ 4 \\ 3 \end{pmatrix}.$$

Den andre måten å løse systemet på er ved radreduksjon av den utvidede matrisen $(A \quad \mathbf{b})$. Sistnevnte er å anbefale, siden radreduksjon av $(A \quad I_n)$ krever mer regning enn radreduksjon av $(A \quad \mathbf{b})$ på grunn av størrelsene på matrisene.

Eksempel 4.7 (Kvadratiske vandermondematriser er inverterbare). La oss vise at vandermondematrisen

$$\begin{pmatrix} 1 & x_0 & x_0^2 & \cdots & x_0^n \\ 1 & x_1 & x_1^2 & \cdots & x_1^n \\ 1 & x_2 & x_2^2 & \cdots & x_2^n \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 1 & x_n & x_n^2 & \cdots & x_n^n \end{pmatrix} \quad (4.1)$$

er inverterbar hvis og bare hvis generatorene $x_0, x_1, \dots, x_n$ er forskjellige. Den ene veien er enkel: Hvis to av generatorene er like så vil to av radene være like, og da er ikke matrisen inverterbar. Den andre veien er litt mer komplisert.

Vi skal bruke induksjon. Det er klart at slike 1×1 -matriser er inverterbare, siden dette bare gir matrisen (1). Anta nå at alle slike matriser av dimensjon $n \times n$ er inverterbare. For å vise at alle slike $(n+1) \times (n+1)$ -matriser er inverterbare så er det nok å vise at deres transponerte er inverterbare (Oppgave 3). Så la oss se på matrisen

$$\begin{pmatrix} 1 & 1 & \cdots & 1 \\ x_0 & x_1 & \cdots & x_n \\ x_0^2 & x_1^2 & \cdots & x_n^2 \\ \vdots & \vdots & \ddots & \vdots \\ x_0^n & x_1^n & \cdots & x_n^n \end{pmatrix}.$$

Vi trekker nå x_0 ganget med rad $i-1$ fra rad i (for $i = n, n-1, \dots, 2$). Disse rad-operasjonene nuller ut alt under eneren i øvre venstre hjørne, og endrer ikke om matrisen er inverterbar eller ikke. Det holder derfor å vise at

$$\begin{pmatrix} 1 & 1 & \cdots & 1 \\ 0 & x_1 - x_0 & \cdots & x_n - x_0 \\ 0 & x_1(x_1 - x_0) & \cdots & x_n(x_n - x_0) \\ \vdots & \vdots & \ddots & \vdots \\ 0 & x_1^{n-1}(x_1 - x_0) & \cdots & x_n^{n-1}(x_n - x_0) \end{pmatrix}$$

er inverterbar. Men dette er det samme som at

$$\begin{pmatrix} x_1 - x_0 & \cdots & x_n - x_0 \\ x_1(x_1 - x_0) & \cdots & x_n(x_n - x_0) \\ \vdots & \ddots & \vdots \\ x_1^{n-1}(x_1 - x_0) & \cdots & x_n^{n-1}(x_n - x_0) \end{pmatrix}$$

er inverterbar, som er det samme som at

$$\begin{pmatrix} x_1 - x_0 & x_1(x_1 - x_0) & \cdots & x_1^{n-1}(x_1 - x_0) \\ \vdots & \vdots & \ddots & \vdots \\ x_n - x_0 & x_n(x_n - x_0) & \cdots & x_n^{n-1}(x_n - x_0) \end{pmatrix}$$

er inverterbar (Oppgave 3 igjen). Vi gjør nå flere radperasjoner: Første rad ganges med $1/(x_1 - x_0)$ andre rad med $1/(x_2 - x_0)$, og så videre. Etter vi har gjort dette ser vi at det er nok å vise at

$$\begin{pmatrix} 1 & x_1 & \cdots & x_1^{n-1} \\ \vdots & \vdots & \ddots & \vdots \\ 1 & x_n & \cdots & x_n^{n-1} \end{pmatrix}$$

er inverterbar. Men dette er en $n \times n$ vandermondematrix, som vi antok inverterbar. Dette fullfører beviset.

Eksempel 4.8 (Fouriermatrisen). Fouriermatrisen er også en vandermondematrix, slik at den også er inverterbar. Mens vi ikke har noen generell formel for den inverse til en vandermondematrix, så er situasjonen annerledes for Fouriermatrisen. Vi påstår at den inverse matrisen har elementer $\frac{1}{N}e^{2\pi i kn}$. Vi har at

$$\sum_{n=0}^{N-1} e^{-2\pi i kn/N} \frac{1}{N} e^{2\pi i nl/N} = \frac{1}{N} \sum_{n=0}^{N-1} e^{2\pi i n(l-k)/N}.$$

Hvis $l = k$ så gir dette 1. Hvis $l \neq k$ kan vi bruke summeformelen for en geometrisk rekke, og vi får

$$\frac{1 - e^{2\pi i(l-k)}}{1 - e^{2\pi i(l-k)/N}} = 0.$$

Det følger at den inverse har elementer $\frac{1}{N}e^{2\pi i kn}$, slik at $(F_N)^{-1} = \frac{1}{N}\overline{F_N}$.

Oppgaver for Kapittel 4

1 (Prøveeksamen 2024). I denne oppgaven setter vi $A = \begin{pmatrix} 2 & -1 & 1 \\ 0 & 1 & 4 \\ 1 & -2 & -6 \end{pmatrix}$

(a). Finn den inverse matrisen til A

(b). Skriv A og A^{-1} som produkter av elementære matriser.

2. Vis at en generell 2×2 -matrise $\begin{pmatrix} a & b \\ c & d \end{pmatrix}$ inverterbar hvis og bare hvis $ad - bc \neq 0$, og at den inverse er $\frac{1}{ad-bc} \begin{pmatrix} d & -b \\ -c & a \end{pmatrix}$.

3. Vis at A er inverterbar hvis og bare hvis A^T er inverterbar. Vis at i så fall er $(A^T)^{-1} = (A^{-1})^T$.

4. Anta A er kvadratisk, og at det finnes en $k > 1$ slik at $A^k = 0$. Vis at A ikke er inverterbar.

5. Anta A og B begge er inverterbare $n \times n$ -matriser. Vis at AB også er inverterbar, og at $(AB)^{-1} = B^{-1}A^{-1}$.

6. En kvadratisk matrise A kalles øvre-triangular hvis alt under diagonalen i A er 0, og nedre-triangular hvis alt over diagonalen i A er 0. A kalles triangular hvis den er enten øvre-triangular eller nedre-triangular.

(a). Vis at en triangular matrise er inverterbar hvis og bare hvis alle elementene på diagonalen er $\neq 0$.

(b). Anta A er inverterbar og øvre/nedre-triangular. Vis at A^{-1} også er øvre/nedre-triangular.

7. La A være en inverterbar matrise. Skriv kode som verifiserer at

1. Når $\mathbf{b}$ og $\mathbf{x}$ er søylevektorer, så returnerer kommandoen `A\b` løsningen på systemet $A\mathbf{x} = \mathbf{b}$.
2. Når $\mathbf{b}$ og $\mathbf{y}$ er radvektorer, så returnerer kommandoen `b/A` løsningen på systemet $\mathbf{y}A = \mathbf{b}$.

KAPITTEL 5

Lineær uavhengighet og basiser

La $\mathbf{a}_1, \mathbf{a}_2, \dots, \mathbf{a}_n$ være vektorer i $\mathbb{R}^m$. Hvis $c_1, c_2, \dots, c_n$ er skalarer, så kaller vi

$$c_1 \mathbf{a}_1 + c_2 \mathbf{a}_2 + \dots + c_n \mathbf{a}_n$$

for en *lineær kombinasjon* av $\mathbf{a}_1, \mathbf{a}_2, \dots, \mathbf{a}_n$. *Spennet av* $\mathbf{a}_1, \mathbf{a}_2, \dots, \mathbf{a}_n$, skrevet

$$\text{Sp}(\mathbf{a}_1, \mathbf{a}_2, \dots, \mathbf{a}_n),$$

er samlingen av alle lineære kombinasjoner av vektorene. Vi sier at vektorene *utspenner* $\mathbb{R}^m$ hvis

$$\text{Sp}(\mathbf{a}_1, \mathbf{a}_2, \dots, \mathbf{a}_n) = \mathbb{R}^m.$$

Videre sier vi at vektorene er *lineært uavhengige* hvis enhver $\mathbf{b} \in \text{Sp}(\mathbf{a}_1, \mathbf{a}_2, \dots, \mathbf{a}_n)$ kan skrives som en lineær kombinasjon av $\mathbf{a}_1, \mathbf{a}_2, \dots, \mathbf{a}_n$ på en entydig måte. Følgende resultat er ofte nyttig for å vise at et sett med vektorer er lineært uavhengige.

Proposisjon 5.1. $\mathbf{a}_1, \mathbf{a}_2, \dots, \mathbf{a}_n$ er lineært uavhengige hvis og bare hvis

$$c_1 \mathbf{a}_1 + c_2 \mathbf{a}_2 + \dots + c_n \mathbf{a}_n = \mathbf{0} \quad (5.1)$$

impliserer at $c_1 = c_2 = \dots = c_n = 0$.

En relasjon av types 5.1 kalles også for en *lineær avhengighetsrelasjon*. Med andre ord, hvis $\mathbf{c} = (c_1, \dots, c_n)$, så kan $A\mathbf{c} = \mathbf{0}$ tolkes som en lineær avhengighetsrelasjon mellom søylene i A .

Bevis. Hvis $\mathbf{a}_1, \mathbf{a}_2, \dots, \mathbf{a}_n$ er lineært uavhengige så har spesielt

$$c_1 \mathbf{a}_1 + c_2 \mathbf{a}_2 + \dots + c_n \mathbf{a}_n = \mathbf{0}$$

den unike løsningen $c_1 = c_2 = \dots = c_n = 0$, siden $\mathbf{0}$ ligger i spennet.

Omvendt, hvis $\mathbf{a}_1, \mathbf{a}_2, \dots, \mathbf{a}_n$ ikke er lineært uavhengige så finnes en $\mathbf{b}$ i spennet som kan skrives på to forskjellige måter

$$\mathbf{b} = c_1 \mathbf{a}_1 + c_2 \mathbf{a}_2 + \dots + c_n \mathbf{a}_n$$

$$\mathbf{b} = d_1 \mathbf{a}_1 + d_2 \mathbf{a}_2 + \dots + d_n \mathbf{a}_n$$

Trekkes disse fra hverandre får vi at

$$(c_1 - d_1) \mathbf{a}_1 + (c_2 - d_2) \mathbf{a}_2 + \dots + (c_n - d_n) \mathbf{a}_n = \mathbf{0}$$

Setter vi $t_k = c_k - d_k$ får vi at $t_1 \mathbf{a}_1 + t_2 \mathbf{a}_2 + \dots + t_n \mathbf{a}_n = \mathbf{0}$, der t_k 'ene ikke alle er null. ■

Under skal vi skrive $\mathbf{a}_i$ for søylene i A . Vi vil få bruk for følgende resultat om lineære avhengighetsrelasjoner.

Proposisjon 5.2. *Lineære avhengighetsrelasjoner mellom søylene i A bevares ved elementære radoperasjoner.*

Bevis. Hvis B kan fås fra A ved hjelp av elementære radoperasjoner så kan vi skrive $B = E_1 \dots E_k A$, der E_i er elementære matriser. Vi trenger å vise at $A\mathbf{x} = \mathbf{0} \iff B\mathbf{x} = \mathbf{0}$. $B\mathbf{x} = \mathbf{0}$ medfører at $E_1 \dots E_k A\mathbf{x} = \mathbf{0}$, som medfører at $A\mathbf{x} = \mathbf{0}$ ved å gange med E_i^{-1} på begge sider. At $A\mathbf{x} = \mathbf{0}$ medfører at $B\mathbf{x} = \mathbf{0}$ følger på samme måte ved å skrive $A = E_k^{-1} \dots E_1^{-1} B$. ■

Begrepene lineær uavhengighet og spenn kan tolkes ved hjelp av pivotelementer:

- At $\mathbf{b}$ er en lineær kombinasjon av $\mathbf{a}_1, \dots, \mathbf{a}_n$ er det samme som at vi kan finne $x_1, x_2, \dots, x_n$ slik at

$$x_1 \mathbf{a}_1 + x_2 \mathbf{a}_2 + \dots + x_n \mathbf{a}_n = \mathbf{b},$$

som er det samme som at $A\mathbf{x} = \mathbf{b}$ har en løsning, som igjen er det samme som at siste søyle i den utvidede matrisen $(A \quad \mathbf{b})$ ikke er en pivotsøyle.

- At $\mathbf{b}$ kan skrives som en lineær kombinasjon av $\mathbf{a}_1, \dots, \mathbf{a}_n$ på en entydig måte krever i tillegg at alle søylene i A er pivotsøyer.
- At $\mathbf{a}_1, \mathbf{a}_2, \dots, \mathbf{a}_n$ utspenner $\mathbb{R}^m$ er det samme som at $A\mathbf{x} = \mathbf{b}$ alltid har en løsning, som er det samme som at alle radene i A er pivotrader (siden dette medfører at siste søyle i $(A \quad \mathbf{b})$ aldri blir en pivotsøyle).
- I en redusert trappeform er pivotsøylene lineært uavhengige, og de andre søylene kan uttrykkes ved disse. Dette gjelder derfor også for selve matrisen - pivotsøylene er lineært uavhengige, og, de andre søylene er lineære kombinasjoner av disse.

Fra dette skal vi vise følgende.

Proposisjon 5.3. *Følgende holder*

- Hvis $\mathbf{a}_1, \dots, \mathbf{a}_n$ utspenner $\mathbb{R}^m$ så har vi at $n \geq m$.
- Hvis $\mathbf{a}_1, \dots, \mathbf{a}_n$ er lineært uavhengige så har vi at $n \leq m$.

Bevis. Hvis $\mathbf{a}_1, \dots, \mathbf{a}_n$ utspenner $\mathbb{R}^m$ vet vi at alle radene i A er pivotrader. Det gir at

$$m = \text{antall rader} = \text{antall pivotrader} = \text{antall pivotsøyer} \leq \text{antall søyer} = n.$$

Hvis $\mathbf{a}_1, \dots, \mathbf{a}_n$ er lineært uavhengige vet vi at alle søylene i A er pivotsøyer. Det gir at

$$n = \text{antall søyer} = \text{antall pivotsøyer} = \text{antall pivotrader} \leq \text{antall rader} = m.$$

■

Vi definerer en *basis* for $\mathbb{R}^m$ til å være et lineært uavhengig sett med vektorer som utspenner $\mathbb{R}^m$. Fra resultatet over følger at en basis for $\mathbb{R}^m$ har nøyaktig m elementer. Vi vil betegne basiser med $\mathcal{B}$, $\mathcal{C}$, o.s.v.. $\mathbb{R}^m$ har spesielt den enkle basisen bestående av

$$\mathbf{e}_1 = \begin{pmatrix} 1 \\ 0 \\ \vdots \\ 0 \end{pmatrix}, \mathbf{e}_2 = \begin{pmatrix} 0 \\ 1 \\ \vdots \\ 0 \end{pmatrix}, \dots, \mathbf{e}_m = \begin{pmatrix} 0 \\ 0 \\ \vdots \\ 1 \end{pmatrix}.$$

Vi betegner denne med $\mathcal{E}$, og kaller den for *standardbasisen*.

Vi kan nå utvide teoremet for inverterbare matriser med enda en ekvivalens:

Proposisjon 5.4 (Teoremet for inverterbare matriser). *Følgende er ekvivalent for en $n \times n$ -matrise A :*

1. A er radekvivalent med I_n .
2. $A\mathbf{x} = \mathbf{0}$ har kun løsningen $\mathbf{x} = \mathbf{0}$.
3. $A\mathbf{x} = \mathbf{b}$ har en entydig løsning for alle høyresider $\mathbf{b}$.
4. A kan skrives som et produkt av elementære matriser.
5. A er inverterbar.
6. Søylene i A danner en basis for $\mathbb{R}^n$.

Bevis. At søylene i A danner en basis for $\mathbb{R}^n$ er det samme som at alle rader og søyler har pivotelementer. Men dette er det samme som at A er radekvivalent med I_n . ■

Eksempel 5.5. Vi har vektorene

$$\mathbf{a}_1 = \begin{pmatrix} 0 \\ 2 \\ 1 \end{pmatrix} \quad \mathbf{a}_2 = \begin{pmatrix} 1 \\ 1 \\ 2 \end{pmatrix} \quad \mathbf{a}_3 = \begin{pmatrix} 1 \\ 2 \\ 0 \end{pmatrix} \quad \mathbf{a}_4 = \begin{pmatrix} -2 \\ 3 \\ 4 \end{pmatrix}.$$

La oss først vise at disse utspenner $\mathbb{R}^3$. Vi radreduserer til trappeform:

$$\begin{pmatrix} 0 & 1 & 1 & -2 \\ 2 & 1 & 2 & 3 \\ 1 & 2 & 0 & 4 \end{pmatrix} \sim \begin{pmatrix} 1 & 2 & 0 & 4 \\ 2 & 1 & 2 & 3 \\ 0 & 1 & 1 & -2 \end{pmatrix} \sim \begin{pmatrix} 1 & 2 & 0 & 4 \\ 0 & -3 & 2 & -5 \\ 0 & 1 & 1 & -2 \end{pmatrix} \sim \begin{pmatrix} 1 & 2 & 0 & 4 \\ 0 & 1 & 1 & -2 \\ 0 & 0 & 5 & -11 \end{pmatrix}$$

Fra trappeformen er det klart at alle radene er pivotrader - vi trenger ikke bringe matrisen til redusert trappeform for å se det. Det følger at de fire vektorene utspenner $\mathbb{R}^3$. Videre ser vi at de tre første søylene er pivotsøyler, slik at disse er lineært uavhengige. Den fjerde vektoren kan uttrykkes ved de tre andre siden den ikke er en pivotsøyle. At de 4 vektorene ikke kan være lineært uavhengige kan man også slutte fra at lineær uavhengighet ville kreve at $m \geq n$, noe som ikke er tilfelle her. De tre søylene $\mathbf{a}_1$, $\mathbf{a}_2$, $\mathbf{a}_3$ er derfor både lineært uavhengige og utspenner $\mathbb{R}^3$, siden alle rader og søyler har pivotelementer i denne 3×3 -matrisen.

Hvis søylene til en matrise ikke utspenner $\mathbb{R}^m$ får vi nå en enkel prosedyre for å utvide settet med pivotsøyler til en basis for hele $\mathbb{R}^m$: Vi modifierer den reduserte trappeformen til A ved å legge til søyer som er standardbasisvektorer, slik at vi sitter igjen med en ledende ener i hver rad. Den nye matrisen har da en ledende ener i hver rad, slik at søylene danner en basis for $\mathbb{R}^m$. Radreduserer vi oss baklengs får vi m vektorer som også er en basis for $\mathbb{R}^m$, og som inneholder de opprinnelige vektorene. Dette er lettest å illustrere med et eksempel.

Eksempel 5.6. La oss se hvordan vi kan utvide

$$\mathbf{a}_1 = \begin{pmatrix} 3 \\ 2 \\ 1 \end{pmatrix} \qquad \mathbf{a}_2 = \begin{pmatrix} 6 \\ -1 \\ 3 \end{pmatrix}$$

til en basis for $\mathbb{R}^3$. Vi radreduserer

$$\begin{pmatrix} 3 & 6 \\ 2 & -1 \\ 1 & 3 \end{pmatrix} \xrightarrow{I \leftrightarrow III} \begin{pmatrix} 1 & 3 \\ 2 & -1 \\ 3 & 6 \end{pmatrix} \xrightarrow{II \sim 2I} \begin{pmatrix} 1 & 3 \\ 0 & -7 \\ 3 & 6 \end{pmatrix} \xrightarrow{III \sim 3I} \begin{pmatrix} 1 & 3 \\ 0 & -7 \\ 0 & -3 \end{pmatrix} \\ \xrightarrow{III \sim (-7)} \begin{pmatrix} 1 & 3 \\ 0 & 1 \\ 0 & -3 \end{pmatrix} \xrightarrow{III \sim 3II} \begin{pmatrix} 1 & 3 \\ 0 & 1 \\ 0 & 0 \end{pmatrix}$$

Siden begge søylene er pivotsøyler følger det at de to vektorene er lineært uavhengige. For å utvide til en basis for $\mathbb{R}^3$ legger vi til en tredje vektor i trappeformen slik at vi får en tredje pivotsøyle, og radreduserer baklengs:

$$\begin{pmatrix} 1 & 3 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix} \xrightarrow{III \sim 3II} \begin{pmatrix} 1 & 3 & 0 \\ 0 & 1 & 0 \\ 0 & -3 & 1 \end{pmatrix} \xrightarrow{II \sim (-7)} \begin{pmatrix} 1 & 3 & 0 \\ 0 & -7 & 0 \\ 0 & -3 & 1 \end{pmatrix} \xrightarrow{III \sim 3I} \begin{pmatrix} 1 & 3 & 0 \\ 0 & -7 & 0 \\ 3 & 6 & 1 \end{pmatrix} \\ \xrightarrow{II \sim 2I} \begin{pmatrix} 1 & 3 & 0 \\ 2 & -1 & 0 \\ 3 & 6 & 1 \end{pmatrix} \xrightarrow{I \leftrightarrow III} \begin{pmatrix} 3 & 6 & 1 \\ 2 & -1 & 0 \\ 1 & 3 & 0 \end{pmatrix}$$

Det følger at vi kan få en basis for $\mathbb{R}^3$ ved å legge til $\mathbf{a}_3 = (1, 0, 0)$.

5.1 Underrom

Med et *underrom* (av $\mathbb{R}^m$) mener vi en mengde $U \subseteq \mathbb{R}^m$ som er slik at

1. hvis $\mathbf{u}_1, \mathbf{u}_2 \in U$, så er også $\mathbf{u}_1 + \mathbf{u}_2 \in U$
2. hvis $\mathbf{u} \in U$, så er også $c\mathbf{u} \in U$ for alle skalarer c .

Det er klart at $\mathbb{R}^m$ spesielt også er et underrom. Det er en oppgave å vise at $\text{Sp}(\mathbf{a}_1, \mathbf{a}_2, \dots, \mathbf{a}_n)$ mer generelt er et underrom. Spesielt er

- *søylerommet* til en matrise (spennet av søylene i matrisen) et underrom (av $\mathbb{R}^m$), og
- *radrommet* til en matrise (spennet av radene i matrisen) et underrom (av $\mathbb{R}^n$).

Vi skal skrive $\text{Col}(A)$ og $\text{Row}(A)$ for søylerommet og radrommet til matrisen A . Hvis B fremkommer fra A ved elementære radoperasjoner, så har A og B samme radrom (dette er en oppgave).

Eksempel 5.7. *Nullrommet* til matrisen A , skrevet $\text{Nul}(A)$, er mengden av alle vektorer $\mathbf{x}$ som er slik at $A\mathbf{x} = \mathbf{0}$. La oss vise at dette er et underrom:

1. hvis $\mathbf{u}_1, \mathbf{u}_2 \in \text{Nul}(A)$, så er $A\mathbf{u}_1 = A\mathbf{u}_2 = \mathbf{0}$. Men da er $A(\mathbf{u}_1 + \mathbf{u}_2) = A\mathbf{u}_1 + A\mathbf{u}_2 = \mathbf{0} + \mathbf{0} = \mathbf{0}$, slik at også $\mathbf{u}_1 + \mathbf{u}_2 \in \text{Nul}(A)$.
2. hvis $\mathbf{u} \in \text{Nul}(A)$, så er $A\mathbf{u} = \mathbf{0}$. Men da er $A(c\mathbf{u}) = cA\mathbf{u} = c\mathbf{0} = \mathbf{0}$.

Det er en oppgave å vise at alle underrom kan skrives som nullrommet til en matrise.

Mer generelt definerer vi en *basis* for et underrom som et lineært uavhengig sett med vektorer som utspenner hele underrommet. Fra den reduserte trappeformen til A følger at

- Pivotradene i A (radene med en ledende ener) er en basis for $\text{Row}(A)$,
- Pivot søylene i A er en basis for $\text{Col}(A)$.

Siden basiser for $\mathbb{R}^m$ har nøyaktig m elementer, så kan ikke et underrom av $\mathbb{R}^m$ ha basiser med flere enn m elementer.

Eksempel 5.8. La oss finne basiser for radrommet, søylerommet, og nullrom-

met til matrisen $A = \begin{pmatrix} 2 & 1 & 3 \\ 1 & 3 & -1 \\ 4 & -3 & 11 \end{pmatrix}$. Radreduksjon gir først

$$A \sim \begin{pmatrix} 1 & 0 & 2 \\ 0 & 1 & -1 \\ 0 & 0 & 0 \end{pmatrix}.$$

De to første radene i den reduserte trappeformen er pivotrader, slik at $(1 \ 0 \ 2)$ og $(0 \ 1 \ -1)$ utgjør en basis for $\text{Row}(A)$.

Siden de to første søylene er pivotsøylar, så er søylene $(2, 1, 4)$ og $(1, 3, -3)$ i A en basis for $\text{Col}(A)$.

Den reduserte trappeformen sier at en vektor i nullrommet må oppfylle $x_1 + 2x_3 = 0$ og $x_2 - x_3 = 0$, slik at $x_1 = -2x_3$ og $x_2 = x_3$. Det følger at en vektor i nullrommet må være på formen

$$\begin{pmatrix} x_1 \\ x_2 \\ x_3 \end{pmatrix} = \begin{pmatrix} -2x_3 \\ x_3 \\ x_3 \end{pmatrix} = x_3 \begin{pmatrix} -2 \\ 1 \\ 1 \end{pmatrix}.$$

Det følger at $(-2, 1, 1)$ er en basis for $\text{Nul}(A)$.

I Eksempel 5.8 hadde vi én fri variabel, og et nullrom av dimensjon 1. Mer generelt er det faktisk slik at antall frie variable alltid er lik dimensjonen til nullrommet. Dette er lettest å illustrere ved et eksempel til. Anta vi har radredusert A , og kom frem til den reduserte trappeformen

$$\begin{pmatrix} 1 & 2 & 3 & 4 \\ 0 & 0 & 0 & 0 \end{pmatrix}.$$

Det er da tre frie variable, x_2 , x_3 , og x_4 . Første rad sier da at $x_1 + 2x_2 + 3x_3 + 4x_4 = 0$, som gir $x_1 = -2x_2 - 3x_3 - 4x_4$. En generell vektor i nullrommet kan derfor skrives

$$\begin{pmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{pmatrix} = \begin{pmatrix} -2x_2 - 3x_3 - 4x_4 \\ x_2 \\ x_3 \\ x_4 \end{pmatrix} = x_2 \begin{pmatrix} -2 \\ 1 \\ 0 \\ 0 \end{pmatrix} + x_3 \begin{pmatrix} -3 \\ 0 \\ 1 \\ 0 \end{pmatrix} + x_4 \begin{pmatrix} -4 \\ 0 \\ 0 \\ 1 \end{pmatrix}, \quad (5.2)$$

slik at

$$\left\{ \begin{pmatrix} -2 \\ 1 \\ 0 \\ 0 \end{pmatrix}, \begin{pmatrix} -3 \\ 0 \\ 1 \\ 0 \end{pmatrix}, \begin{pmatrix} -4 \\ 0 \\ 0 \\ 1 \end{pmatrix} \right\}$$

utspenner nullrommet. Det er også klart at disse er lineært uavhengige, slik at de også er en basis for nullrommet: Hvis den lineære kombinasjonen helt til høyre i (5.2) er 0, så sier det forrige uttrykket der at $x_2 = x_3 = x_4 = 0$. Alt i alt er dimensjonen til nullrommet 3, som også var antall frie variable.

Fra regningen over følger det mer generelt at alle nullrom kan skrives som et spenn av vektorer. Videre var det en oppgave at alle underrom kan skrives som et nullrom. Og siden ethvert spenn også gir et underrom, så kan vi tenke på spenn, nullrom, og underrom, som samme tingen. Dette kan begrunnes på flere måter - for eksempel kan man steg for steg øke et sett med vektorer med en

vektor til, så lenge du kan finne en ny vektor lineært uavhengig fra de forrige. Vi vet at denne prosessen vil stoppe senest ved m elementer. Når den stopper vet vi at underrommet er spennet av de vektorene vi stegvis har lagt til.

På samme måte kan vi vise følgende:

Proposisjon 5.9. *Alle underrom har en ortogonal basis.*

Bevis. Vi starter ved å skrive underrommet som et spenn, $\text{Sp}(\mathbf{a}_1, \dots, \mathbf{a}_n)$, der vektorene er lineært uavhengige. Anta vi har klart å finne r vektorer $\mathbf{q}_1, \dots, \mathbf{q}_r$ som er ortogonale, og slik at

$$\text{Sp}(\mathbf{q}_1, \dots, \mathbf{q}_r) = \text{Sp}(\mathbf{a}_1, \dots, \mathbf{a}_r).$$

Definer så

$$\mathbf{q}_{r+1} = \mathbf{a}_{r+1} - \sum_{i=1}^r \frac{\mathbf{a}_{r+1} \cdot \mathbf{q}_i}{\mathbf{q}_i \cdot \mathbf{q}_i} \mathbf{q}_i. \quad (5.3)$$

Vi har da at

$$\begin{aligned} \mathbf{q}_{r+1} \cdot \mathbf{q}_j &= \mathbf{a}_{r+1} \cdot \mathbf{q}_j - \sum_{i=1}^r \frac{\mathbf{a}_{r+1} \cdot \mathbf{q}_i}{\mathbf{q}_i \cdot \mathbf{q}_i} \mathbf{q}_i \cdot \mathbf{q}_j \\ &= \mathbf{a}_{r+1} \cdot \mathbf{q}_j - \frac{\mathbf{a}_{r+1} \cdot \mathbf{q}_j}{\mathbf{q}_j \cdot \mathbf{q}_j} \mathbf{q}_j \cdot \mathbf{q}_j \\ &= \mathbf{a}_{r+1} \cdot \mathbf{q}_j - \mathbf{a}_{r+1} \cdot \mathbf{q}_j = 0 \end{aligned}$$

Det følger at $\mathbf{q}_1, \dots, \mathbf{q}_{r+1}$ også er ortogonale, og det er lett å se at disse har det samme spennet som $\mathbf{a}_1, \dots, \mathbf{a}_{r+1}$. ■

Et underrom har mange forskjellige basiser, men alle slike basiser har like mange elementer:

Proposisjon 5.10. *Alle basiser for et gitt underrom har like mange elementer.*

Bevis. Anta at

$$\text{Sp}(\mathbf{v}_1, \mathbf{v}_2, \dots, \mathbf{v}_n) = \text{Sp}(\mathbf{w}_1, \mathbf{w}_2, \dots, \mathbf{w}_k)$$

der vi antar at $n > k$. Vi har en avbildning $\mathbf{F}: \mathbb{R}^n \rightarrow \mathbb{R}^k$ gitt ved at

$$\mathbf{F}(c_1, c_2, \dots, c_n) = (d_1, d_2, \dots, d_k)$$

når

$$c_1 \mathbf{v}_1 + c_2 \mathbf{v}_2 + \cdots + c_n \mathbf{v}_n = d_1 \mathbf{w}_1 + d_2 \mathbf{w}_2 + \cdots + d_n \mathbf{w}_k.$$

$\mathbf{F}$ er definert på hele $\mathbb{R}^n$, og er veldefinert. Videre er den $1 \rightarrow 1$ og lineær, og derfor gitt ved en $k \times n$ -matrise A . Siden $n > k$ må vi da ha minst en søyle uten pivotelement. Men da har vi en lineær avhengighetsrelasjon

$$x_1 \mathbf{a}_1 + x_2 \mathbf{a}_2 + \cdots + x_n \mathbf{a}_n = \mathbf{0}.$$

mellom søylene i A , slik at $A\mathbf{x} = \mathbf{0}$. Dette strider mot at $\mathbf{F}(\mathbf{x}) = A\mathbf{x}$ er $1 \rightarrow 1$, siden også $\mathbf{F}(\mathbf{0}) = A\mathbf{0} = \mathbf{0}$. ■

Antall vektorer i en basis kalles for *dimensjonen* til rommet. Når en matrise er på redusert trappeform ser vi at dimensjonen til både radrommet og søylerommet er lik antall pivotelementer. Den felles dimensjonen til $\text{Col}(A)$ og $\text{Row}(A)$ kalles for *rangen* til matrisen, skrevet $\text{rang}(A)$.

Eksempel 5.11 (Matriser med rang 1). La $\mathbf{a}$ og $\mathbf{b}$ være søylevektorer $\neq \mathbf{0}$ av lengde m og n respektive. Vi har at

$$\mathbf{a}\mathbf{b}^T = \begin{pmatrix} a_1 \mathbf{b}^T \\ a_2 \mathbf{b}^T \\ \vdots \\ a_m \mathbf{b}^T \end{pmatrix} = \begin{pmatrix} b_1 \mathbf{a} & b_2 \mathbf{a} & \cdots & b_n \mathbf{a} \end{pmatrix}.$$

Det følger at $\{\mathbf{a}\}$ er en basis for søylerommet til $\mathbf{a}\mathbf{b}^T$, og at $\{\mathbf{b}^T\}$ er en basis for radrommet til $\mathbf{a}\mathbf{b}^T$. $\mathbf{a}\mathbf{b}^T$ har altså rang 1. Det er også klart at alle matriser med rang en kan skrives på formen $\mathbf{a}\mathbf{b}^T$, der $\mathbf{a}$ og $\mathbf{b}$ er søylevektorer $\neq \mathbf{0}$.

Ved å legge multiple av første rad til de andre radene så er det klart at $\mathbf{a}\mathbf{b}^T$ kan radreduseres til $\begin{pmatrix} \mathbf{b}^T \\ \mathbf{0} \\ \vdots \\ \mathbf{0} \end{pmatrix}$. Det følger at vi har $n - 1$ frie variable når vi løser $\mathbf{a}\mathbf{b}^T \mathbf{x} = \mathbf{0}$, slik at nullrommet har dimensjon $n - 1$.

Matriser av rang en kan være nyttige byggeklosser. Den ytre formen for matriser (Proposisjon 2.13) sa for eksempel at et produkt av to matriser kunne skrives som en sum av matriser av rang en. Vi skal også komme tilbake til dette når vi snakker om diagonalisering.

Proposisjon 5.12. *La A være en $m \times n$ -matrise. Da har vi at*

$$\text{rang}(A) + \dim(\text{Nul}(A)) = n.$$

I Eksempel 5.8 fant vi både søylerommet og radrommet hadde dimensjon 2, slik at rangen også er 2. Dimensjonen til nullrommet var 1. Dette stemmer overens med proposisjonen siden $2 + 1 = 3 = n$. I eksempel 5.11 fant vi at rangen var 1, og dimensjonen til nullrommet var $n - 1$. Dette stemmer overens med proposisjonen, siden $1 + (n - 1) = n$.

Bevis. Dette følger fra at dimensjonen til nullrommet er lik antall frie variable, rangen er lik antall pivotsøyer, og at summen av antall pivotvariable og antall frie variable er lik antall søyler i matrisen. ■

Vi skal avslutte med følgende resultat:

Proposisjon 5.13. *Den reduserte trappeformen er unik.*

Bevis. Anta at A kan radreduseres til to forskjellige reduserte trappeformer C_1 og C_2 . Vi viser først at C_1 og C_2 har de samme søyler med ledende enere. Hvis dette ikke er tilfelle, så finnes det en søyleindeks k som er slik at

- søylene $1, \dots, k - 1$ i både C_1 og C_2 har ledende enere
- søyle k i C_1 har en ledende ener, mens søyle k i C_2 ikke har det.

Da kan søyle k i C_1 ikke uttrykkes ved søylene $1, \dots, k - 1$ i C_1 , mens søyle k i C_2 , kan uttrykkes ved søylene $1, \dots, k - 1$ i C_2 . Dette motsier at C_1 og C_2 har de samme lineære avhengighetsrelasjonene (de er jo radredusert fra en felles matrise). Det følger at C_1 og C_2 har de samme søyler med ledende enere. Dette betyr også at disse søylene er like, siden de ledende enerne må komme i posisjon 1,2,3, o.s.v..

Vi viser så at også søylene som ikke har ledende enere må være like. Disse kan skrives om lineærkombinasjoner av søylene som har ledende enere. Siden disse er like, og de lineære avhengighetsrelasjonene er de samme, så følger det at de andre søylene i C_1 og C_2 også er like. ■

Legg merke til at begrepet pivotsøyle ikke ble brukt for de reduserte trappeformene over. Det er faktisk først etter dette beviset at begrepet pivotsøyle for generelle matriser gir mening - siden vi nå har vist at de blir de samme uansett hvordan vi radreduserer matrisen.

Oppgaver for Kapittel 5

1. Anta at A og B er radekvivalente. Vis at $\text{Row}(A) = \text{Row}(B)$.

2. Vi har matrisen $A = \begin{pmatrix} 2 & 4 & 1 & -1 \\ 1 & -1 & 2 & 3 \\ -1 & -11 & 4 & 11 \end{pmatrix}$.

(a). Finn basiser for radrommet og søylerommet til A . Hva er rangen til A ?

(b). Finn en basis for nullrommet til A . Hva er dimensjonen til nullrommet?

3 (Eksamen 2024). Vi ser på matrisen $A = \begin{pmatrix} 2 & 8 & 6 & 1 \\ -4 & -16 & -12 & 2 \\ -1 & -4 & -3 & 1 \end{pmatrix}$

(a). Finn en basis for søylerommet til A .

(b). Utvid basisen du fant i (a) til en basis for $\mathbb{R}^3$.

4. Anta at $m > n$, og at søylene $\mathbf{a}_1, \dots, \mathbf{a}_n$ i A er lineært uavhengige. Anta også at de første n radene i A er lineært uavhengige. Forklar at

$$\{\mathbf{a}_1, \dots, \mathbf{a}_n, \mathbf{e}_{n+1}, \dots, \mathbf{e}_m\}$$

er en basis for $\mathbb{R}^m$.

5. Vis at $\text{Sp}(\mathbf{a}_1, \mathbf{a}_2, \dots, \mathbf{a}_n)$ er et underrom av $\mathbb{R}^m$

6. La A være en mengde i $\mathbb{R}^n$. Ortogonalkomplementet til A , $A^\perp$, er definert som alle vektorer i $\mathbb{R}^n$ som er ortogonale på alle vektorer i A , det vil si

$$A^\perp = \{\mathbf{x} \in \mathbb{R}^n : \mathbf{x} \cdot \mathbf{a} = 0 \forall \mathbf{a} \in A\}.$$

Vis at $A^\perp$ er et underrom av $\mathbb{R}^n$.

7. La U og V være underrom av $\mathbb{R}^n$. Vis at

$$\{\mathbf{u} + \mathbf{v} : \mathbf{u} \in U, \mathbf{v} \in V\}$$

også er et underrom av $\mathbb{R}^n$.

8. Vis at ethvert underrom kan skrives som nullrommet til en matrise.

Hint: Et generelt underrom kan skrives som $\text{Sp}(\mathbf{a}_1, \mathbf{a}_2, \dots, \mathbf{a}_n)$, der vektorene er lineært uavhengige. Utvid disse til en basis for $\mathbb{R}^m$, og la A være matrisen med disse som søyler. Se deretter på nullrommet til matrisen

$$C = \left(\begin{array}{c|c} \mathbf{0} & \mathbf{0} \\ \hline \mathbf{0} & I \end{array} \right) A^{-1}.$$

9. La vektorene $\{\mathbf{q}_i\}_{i=1}^n$ være definert ved (5.3). I denne oppgaven skal vi se at (5.3) gir opphav til en matrisefaktorisering som kalles *QR-faktoriseringen*.

(a). Vis at

$$(\mathbf{a}_1 \quad \mathbf{a}_2 \quad \cdots \quad \mathbf{a}_n) = (\mathbf{q}_1 \quad \mathbf{q}_2 \quad \cdots \quad \mathbf{q}_n) \begin{pmatrix} 1 & \frac{\mathbf{a}_2 \cdot \mathbf{q}_1}{\mathbf{q}_1 \cdot \mathbf{q}_1} & \cdots & \frac{\mathbf{a}_n \cdot \mathbf{q}_1}{\mathbf{q}_1 \cdot \mathbf{q}_1} \\ 0 & 1 & \cdots & \frac{\mathbf{a}_n \cdot \mathbf{q}_2}{\mathbf{q}_2 \cdot \mathbf{q}_2} \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & 1 \end{pmatrix} \quad (5.4)$$

(b). Forklar at følgende program regner ut de to matrisene på høyre side i (5.4). I koden er A matrisen med søyler $\mathbf{a}_i$.

```
m, n = np.shape(A)
Q = np.zeros((m,n)); Q[:,0] = A[:,0];
R = np.eye(n);
for j in range(1,n):
    R[[j-1],j:] = Q[:,[j-1]].T @ A[:,j:]/(Q[:,[j-1]].T @ Q[:,[j-1]]).item()
    Q[:,[j]] = A[:,[j]] - Q[:,[:j]]@R[:,[j]]
```

(c). Vis at

$$(\mathbf{a}_1 \quad \mathbf{a}_2 \quad \cdots \quad \mathbf{a}_n) = \begin{pmatrix} \frac{\mathbf{q}_1}{|\mathbf{q}_1|} & \frac{\mathbf{q}_2}{|\mathbf{q}_2|} & \cdots & \frac{\mathbf{q}_n}{|\mathbf{q}_n|} \end{pmatrix} \begin{pmatrix} |\mathbf{q}_1| & \frac{\mathbf{a}_2 \cdot \mathbf{q}_1}{|\mathbf{q}_1|} & \cdots & \frac{\mathbf{a}_n \cdot \mathbf{q}_1}{|\mathbf{q}_1|} \\ 0 & |\mathbf{q}_2| & \cdots & \frac{\mathbf{a}_n \cdot \mathbf{q}_2}{|\mathbf{q}_2|} \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & |\mathbf{q}_n| \end{pmatrix}$$

Dette betyr at $A = QR$ der A er matrisen med $\{\mathbf{a}_i\}_{i=1}^n$ som søyler, Q er den ortogonale matrisen med $\{\mathbf{q}_i/|\mathbf{q}_i|\}_{i=1}^n$ som søyler, og R er den øvretriangulære matrisen gitt over. Modifiser også koden fra (b) slik at de nye matrisene på høyre side blir regnet ut i stedet.

KAPITTEL 6

Determinanter

I dette kapitlet skal vi se på *determinanter*. Disse er nyttige teoretisk med en lang historie i matematikk (spesielt i flervariabel analyse), men har en ganske komplisert definisjon. Vi skal starte med den generelle definisjonen av determinant, og deretter se på hvordan denne endrer seg ved elementære radoperasjoner.

Vi skal deretter se at determinanten til en matrise sier noe om hvordan den transformerer et volum i $\mathbb{R}^n$. Vi har alle en formening om hva et volum/areal i $\mathbb{R}^3/\mathbb{R}^2$ er, men med volumer i $\mathbb{R}^n$ for $n > 3$ stiller det seg annerledes. Vi skal definere dette, og deretter se at, hvis A er en $n \times n$ -matrise, og X er en mengde i $\mathbb{R}^n$ så har mengden

$$AX = \{A\mathbf{x} : \mathbf{x} \in X\}$$

et volum, skrevet $V(AX)$, som avhenger lineært av volumet $V(X)$ til X - uansett hvordan mengden X ser ut. Vi har altså $V(AX) = kV(X)$ for en konstant k som bare avhenger av matrisen A . k kan tolkes som en *forstørrelsesfaktor* for volumer for matrisen A , og vi skal se at forstørrelsesfaktorer endrer seg ved elementære radoperasjoner på samme måte som determinanter gjør det. Fra dette utleder vi at absoluttverdien til determinanten er lik forstørrelsesfaktoren.

En av tingene vi skal vise er at determinanten til et produkt er lik produktet av determinantene. Ser vi bort fra fortegnet til determinanten så er dette kanskje enklere å overbevise seg om ved hjelp av forstørrelsesfaktorer: Hvis A_1 og A_2 har forstørrelsesfaktorer k_1 og k_2 , respektive, så har A_2X volum $k_2V(X)$, slik at A_1A_2X har volum $k_1(k_2V(X)) = k_1k_2V(X)$, slik at A_1A_2 har forstørrelsesfaktor k_1k_2 .

Vi avslutter kapitlet med å gi en nyttig geometrisk tolkning av determinanten.

6.1 Definisjon av determinanten

Determinanten er generelt definert som følger.

Definisjon 6.1. Determinanten til en $n \times n$ -matrise A defineres rekursivt som følger:

1. Determinanten til en 1×1 -matrise A er definert ved a_{11} .
2. Determinanten til en $n \times n$ -matrise A er definert ved hjelp av determinanten til $(n-1) \times (n-1)$ -submatriser av A ved

$$\begin{vmatrix} a_{11} & a_{12} & \cdots & a_{1n} \\ a_{21} & a_{22} & \cdots & a_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ a_{n1} & a_{n2} & \cdots & a_{nn} \end{vmatrix} = a_{11} \begin{vmatrix} a_{22} & a_{23} & \cdots & a_{2n} \\ a_{32} & a_{33} & \cdots & a_{3n} \\ \vdots & \vdots & \ddots & \vdots \\ a_{n2} & a_{n3} & \cdots & a_{nn} \end{vmatrix} - a_{12} \begin{vmatrix} a_{21} & a_{23} & \cdots & a_{2n} \\ a_{31} & a_{33} & \cdots & a_{3n} \\ \vdots & \vdots & \ddots & \vdots \\ a_{n1} & a_{n3} & \cdots & a_{nn} \end{vmatrix} \\ + \cdots + (-1)^{n+1} a_{1n} \begin{vmatrix} a_{21} & a_{22} & \cdots & a_{2(n-1)} \\ a_{31} & a_{32} & \cdots & a_{3(n-1)} \\ \vdots & \vdots & \ddots & \vdots \\ a_{n1} & a_{n2} & \cdots & a_{n(n-1)} \end{vmatrix}, \quad (6.1)$$

I alle submatrisene på høyre side er rad 1 i A utelatt, I den første submatrisen er i tillegg søyle 1 i A utelatt, i den andre er søyle 2 i A utelatt, o.s.v.

I tillegg til notasjonen for determinanten med vertikale streker rundt elementene i matrisen, så skriver vi også $\det(A)$ for determinanten til A . La oss skrive $A^{(ij)}$ for $(n-1) \times (n-1)$ -matrisen vi får ved å utelate rad i og søyle j i A . Størrelsene $\det(A^{(ij)})$ kalles også for *minorer*. Med denne notasjonen kan (6.1) skrives

$$\det(A) = a_{11} \det(A^{(11)}) - a_{12} \det(A^{(12)}) + a_{13} \det(A^{(13)}) - \cdots + a_{1n} (-1)^{n+1} \det(A^{(1n)}),$$

eller enda mer kompakt

$$\det(A) = \sum_{j=1}^n (-1)^{j+1} a_{1j} \det(A^{(1j)}).$$

Determinanten til en 1×1 -matrise er per definisjon det ene elementet i matrisen, og for en 2×2 -matrise er den

$$\det(A) = \begin{vmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{vmatrix} = a_{11} a_{22} - a_{12} a_{21}, \quad (6.2)$$

Determinanten til en 3×3 -matrise blir

$$\begin{vmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & a_{33} \end{vmatrix} = a_{11} \begin{vmatrix} a_{22} & a_{23} \\ a_{32} & a_{33} \end{vmatrix} - a_{12} \begin{vmatrix} a_{21} & a_{23} \\ a_{31} & a_{33} \end{vmatrix} + a_{13} \begin{vmatrix} a_{21} & a_{22} \\ a_{31} & a_{32} \end{vmatrix} \quad (6.3)$$

Eksempel 6.2 (Utrekning av determinant med definisjonen). La oss regne ut de-

terminanten til matrisen $\begin{pmatrix} 2 & 1 & 3 \\ 1 & 4 & 2 \\ 3 & 1 & -1 \end{pmatrix}$ ved hjelp av definisjonen. Vi får

$$\begin{vmatrix} 2 & 1 & 3 \\ 1 & 4 & 2 \\ 3 & 1 & -1 \end{vmatrix} = 2 \begin{vmatrix} 4 & 2 \\ 1 & -1 \end{vmatrix} - 1 \begin{vmatrix} 1 & 2 \\ 3 & -1 \end{vmatrix} + 3 \begin{vmatrix} 1 & 4 \\ 3 & 1 \end{vmatrix} \\ = 2(-4 - 2) - (-1 - 6) + 3(1 - 12) = -12 + 7 - 33 = -38.$$

Vi kan sjekke svaret ved hjelp av den innebygde funksjonen for å regne ut determinanten:

```
>> A=np.array([[2,1,3],[1,4,2],[3,1,-1]])
>> np.linalg.det(A)
-37.99999999999999
```

La $K(n)$ være antall regneoperasjoner som kreves hvis man regner ut determinanten rekursivt ved hjelp av Likning (6.1). Da er det klart at

$$K(n) = nK(n-1) + n + n - 1 \geq nK(n-1).$$

Vi skjønner fra dette at vi må ha $K(n) \geq n!$. Når n blir stor gir dette for mye regning selv for de kraftigste datamaskiner. Senere i dette kapitlet skal vi se at man kan regne ut determinanten mye mer effektivt ved hjelp av Gausseliminering. Definisjonen over er mer av teoretisk interesse - den gir ingen god implementasjon selv om den er lett å programmere, som vist under.

```
def detdef(A):
    n = np.shape(A)[0]
    if n>1:
        d=0
        for k in range(n):
            colinds = np.arange(0,n)
```

```

        colinds=np.delete(colinds, k)
        d += (-1)**k*A[0,k]*detdef( A[1:,colinds] )
    return d
else:
    return A.item()

```

At definisjonen (6.1) ikke blir brukt til å regne ut determinanten i Eksempel 6.2 kan vi kanskje gjette ved å observere at svaret kommer med fire desimaler (så svaret er ikke -38 eksakt). Hadde definisjonen vært brukt ville all regning skje med heltallsaritmetikk, og svaret burde bli -38 eksakt. Kjører du koden over vil du se at nettopp det skjer:

```

>> detdef( np.array([[2,1,3],[1,4,2],[3,1,-1]]) )
-38

```

6.2 Egenskaper til determinanten

La oss så vise egenskapene til determinanten. Vi starter med følgende.

Proposisjon 6.3. *Vi har at $\det(A) = 0$ hvis A har en rad med bare nuller, eller en søyle med bare nuller.*

Bevis. Dette er enklest å se for nullrader: Bruker vi (6.1) så mange ganger at vi kun har submatriser som starter med en nullrad, så er det klart at neste anvendelse av (6.1) gir 0. Anta i stedet at vi har en nullsøyle. For 1×1 -matriser med en nullsøyle er det klart at determinanten er 0. Anta så at vi har vist at alle $(n-1) \times (n-1)$ -matriser med en nullsøyle har determinant 0. Når vi bruker (6.1) på en $n \times n$ -matrise med en nullsøyle, så får vi bidrag fra $n-1$ $(n-1) \times (n-1)$ -submatriser med nullsøyer. Og bidraget fra den resterende søylen blir 0, siden vi har en null på tilsvarende plass i første rad som ganges inn. Det følger at $n \times n$ -matriser med en nullsøyle også har determinant 0. ■

Vi minner om at en kvadratisk matrise kalles *øvre-triangulær* hvis alt under diagonalen er 0, og *nedre-triangulær* hvis alt over diagonalen er 0.

Proposisjon 6.4. *Determinanten til en øvre- eller nedre-triangulær matrise er lik produktet av diagonalelementene.*

Bevis. Resultatet er opplagt riktig for $n = 1$. Hvis A er nedre-triangulær så sier (6.1) at $\det(A) = a_{11} \det(A^{(11)})$. Siden $A^{(11)}$ også er nedre-triangulær, og med $a_{22}, a_{33}, \dots, a_{nn}$ på diagonalen, så følger det ved induksjon at

$$\det(A) = a_{11}(a_{22}a_{33} \cdots a_{nn}) = a_{11}a_{22}a_{33} \cdots a_{nn}.$$

Anta i stedet at A er øvre-triangulær. Første søyle i minorene $A^{(12)}, A^{(13)}, \dots, A^{(1n)}$ er da alle null, og da sier Proposisjon 6.3 at

$$\det(A^{(12)}) = \det(A^{(13)}) = \cdots = \det(A^{(1n)}) = 0.$$

Bare den første minoren bidrar i (6.1), slik at $\det(A) = a_{11} \det(A^{(11)})$, som igjen ved induksjon gir at $\det(A) = a_{11}a_{22}a_{33} \cdots a_{nn}$ ■

Vi har følgende resultat for hvordan elementære radoperasjoner påvirker determinanten.

Proposisjon 6.5. *La A og B være matriser av samme dimensjon. Følgende holder:*

1. *Hvis B fremkommer fra et enkelt radbytte i A , så er $\det(B) = -\det(A)$.*
2. *Hvis B fremkommer ved å gange en rad i A med s , så er $\det(B) = s \det(A)$.*
3. *Hvis B fremkommer ved å legge s ganger rad i til rad j i A , så er $\det(B) = \det(A)$.*

Bevis. Vi viser punkt 1.. La D_{ij} være determinanten til matrisen vi får ved å fjerne rad 1 og 2 i A , samt søyle i og j i A . Vi har da at

$$\begin{aligned} \det(A) &= \sum_{i < j} ((-1)^{i+1}(-1)^j a_{1i}a_{2j} + (-1)^{j+1}(-1)^{i+1} a_{1j}a_{2i}) D_{ij} \\ &= \sum_{i < j} (-1)^{i+j} (a_{1j}a_{2i} - a_{1i}a_{2j}) D_{ij} \end{aligned}$$

Hvis B er matrisen der rad 1 og 2 i A er byttet om, så har vi at $b_{1i} = a_{2i}$ og $b_{1j} = a_{2j}$ o.s.v., og vi får

$$\begin{aligned} \det(B) &= \sum_{i < j} (-1)^{i+j} (b_{1j}b_{2i} - b_{1i}b_{2j}) D_{ij} = \sum_{i < j} (-1)^{i+j} (a_{2j}a_{1i} - a_{2i}a_{1j}) D_{ij} \\ &= - \sum_{i < j} (-1)^{i+j} (a_{2i}a_{1j} - a_{2j}a_{1i}) D_{ij} = -\det(A) \end{aligned}$$

Hvis to andre naborader byttes om så følger det ved induksjon, og definisjonen $\det(A) = \sum_{j=1}^n (-1)^{j+1} a_{1j} A_{1j}$ at determinanten også skifter fortegn. Videre hvis rad i og j ikke er naborader, så kan radbytte av disse uttrykkes ved $j - i + (j - i - 1) = 2(j - i) - 1$ naboradbytter. Siden dette er et oddetall vil determinanten også da skifte fortegn. Dermed er punkt 1. vist.

Bevisene for punkt 2 og 3 er gitt som oppgaver. ■

Hvis A har 2 like rader, så ender vi opp med samme matrise hvis disse to radene bytter plass. Som en konsekvens av punkt 1. over må dermed $\det(A) = -\det(A)$, slik at $\det(A) = 0$.

Korollar 6.6. $\det(A) = 0$ hvis A har to identiske rader

Dette resultatet er nyttig når det kommer til å bevise punkt 3. over. Vi kan også utvikle determinanten langs en hvilket som helst rad (dette er gitt som en oppgave):

Proposisjon 6.7 (Determinanter kan utvikles langs enhver rad). For alle $1 \leq i \leq n$ har vi at

$$\det(A) = \sum_{j=1}^n (-1)^{i+j} a_{ij} \det(A^{(ij)}).$$

Setter vi spesielt $A = I_n$ (som har determinant 1) og $B = E$ (der E er elementær) i Proposisjon 6.5 så får vi at

$$\det(E) = \begin{cases} -1 & \text{hvis } E \text{ er et radbytte.} \\ s & \text{hvis } E \text{ ganger en rad med } s. \\ 1 & \text{hvis } E \text{ legger til et multiplum av en rad til en annen.} \end{cases} \quad (6.4)$$

Vi kan nå vise følgende.

Proposisjon 6.8. Hvis A og B begge er kvadratiske så har vi at $\det(AB) = \det(A) \det(B)$.

Bevis. Vi kan skrive

$$A = E_1 E_2 \cdots E_k C \qquad B = F_1 F_2 \cdots F_l D$$

der $E_1, F_1, E_2, F_2, \dots$ er elementære matriser, og C og D er reduserte trappeformer. Hvis $C = D = I_n$ så er

$$\begin{aligned} \det(AB) &= \det(E_1 E_2 \cdots E_k F_1 F_2 \cdots F_l) = \det(E_1) \det(E_2) \cdots \det(E_k) \det(F_1 F_2 \cdots F_l) \\ &= \det(E_1 E_2 \cdots E_k) \det(F_1 F_2 \cdots F_l) = \det(A) \det(B). \end{aligned}$$

Anta så at en av C, D er $\neq I_n$. Det følger at en av $\det(A)$ eller $\det(B)$ er lik 0, slik at også $\det(A) \det(B) = 0$. Det er derfor nok å vise at også $\det(AB) = 0$. Det er klart at $\det(AB) = 0$ hvis og bare hvis $\det(CF_1 F_2 \cdots F_l D) = 0$. Vi ser raskt på to muligheter:

1. Hvis $C = I_n, D \neq I_n$: Da er $\det(D) = 0$ (siden D har en nullrad). Men da er

$$\det(CF_1 F_2 \cdots F_l D) = \det(F_1 F_2 \cdots F_l D) = 0.$$

2. Hvis $C \neq I_n$ så har C en nullrad. Men da har også $CF_1 F_2 \cdots F_l D$ en nullrad, og da er $\det(CF_1 F_2 \cdots F_l D) = 0$

■

Spesielt kan vi finne determinanten ved hjelp av Gausseliminering. La oss vise dette med et eksempel.

Eksempel 6.9 (Utregning av determinant med Gausseliminering). La oss verifisere determinanten vi regnet ut i Eksempel 6.2 med Gausseliminering. Vi får

$$\begin{aligned} \begin{pmatrix} 2 & 1 & 3 \\ 1 & 4 & 2 \\ 3 & 1 & -1 \end{pmatrix} &\sim \begin{pmatrix} 1 & 4 & 2 \\ 2 & 1 & 3 \\ 3 & 1 & -1 \end{pmatrix} \sim \begin{pmatrix} 1 & 4 & 2 \\ 0 & -7 & -1 \\ 0 & -11 & -7 \end{pmatrix} \sim \begin{pmatrix} 1 & 4 & 2 \\ 0 & 1 & 1/7 \\ 0 & -11 & -7 \end{pmatrix} \\ &\sim \begin{pmatrix} 1 & 4 & 2 \\ 0 & 1 & 1/7 \\ 0 & 0 & -38/7 \end{pmatrix} \end{aligned}$$

Den første radoperasjonen var et radbytte. Deretter kom to operasjoner der vi la et multiplum av rad 1 til de andre radene. Deretter skalerte vi rad 2 med $(-1/7)$, før vi la et multiplum av rad 2 til rad 3. Vi endte opp med en øvretriangulær matrise, der vi vet determinanten er produktet av diagonalelementene. Vi får da

$$\begin{vmatrix} 1 & 4 & 2 \\ 0 & 1 & 1/7 \\ 0 & 0 & -38/7 \end{vmatrix} = 1 \times (-1/7) \times (1) \times (-1) \begin{vmatrix} 2 & 1 & 3 \\ 1 & 4 & 2 \\ 3 & 1 & -1 \end{vmatrix}$$

som gir at

$$(-38/7) = (1/7) \begin{vmatrix} 2 & 1 & 3 \\ 1 & 4 & 2 \\ 3 & 1 & -1 \end{vmatrix},$$

slik at determinanten blir -38 .

For store matriser (som man ofte har i anvendelser) er det mye mer effektivt å regne ut determinanter ved Gausseliminering som over, når vi sammenligner med å bruke definisjonen av determinanten som i Eksempel 6.2: I Seksjon viste vi jo at antall operasjoner som trengs ved Gausseliminering er omtrent $2n^3/3$, mens vi så at definisjonen trengte minst $n!$ operasjoner.

Igien kan vi nå utvide teoremet for inverterbare matriser med en ekvivalens. Dette blir den endelige versjonen av dette teoremet:

Proposisjon 6.10 (Teoremet for inverterbare matriser). *Følgende er ekvivalent for en $n \times n$ -matrise A :*

1. A er radekvivalent med I_n .
2. $A\mathbf{x} = \mathbf{0}$ har kun løsningen $\mathbf{x} = \mathbf{0}$.
3. $A\mathbf{x} = \mathbf{b}$ har en entydig løsning for alle høyresider $\mathbf{b}$.
4. A kan skrives som et produkt av elementære matriser.
5. A er inverterbar.
6. Søylene i A danner en basis for $\mathbb{R}^n$.
7. $\det(A) \neq 0$

Bevis. $4. \Rightarrow 7.$: Hvis $A = E_k \cdots E_1$ så er

$$\det(A) = \det(E_k \cdots E_1) = \det(E_k) \cdots \det(E_1) \neq 0,$$

der vi brukte Proposisjon 6.8, og at elementære matriser har determinant forskjellig fra 0.

$7. \Rightarrow 1.$ Anta A ikke er radekvivalent med I_n . Da kan vi skrive $C = E_k \cdots E_1 A$, og $A = F_1 \cdots F_k C$, der $C \neq I_n$ er den reduserte trappeformen til A . Men da er $\det(C) = 0$, og det følger fra Proposisjon 6.8 at $\det(A) = 0$. ■

Proposisjon 6.11. For alle kvadratiske matriser A gjelder at $\det(A) = \det(A^T)$.

Bevis. Det er klart at for alle elementære matriser E så er $\det(E) = \det(E^T)$. Anta $A = E_1 E_2 \cdots E_k C$, der C er den reduserte trappeformen til A . Vi har at

$$\begin{aligned}\det(A^T) &= \det(C^T E_k^T E_{k-1}^T \cdots E_1^T) = \det(C^T) \det(E_k^T) \det(E_{k-1}^T) \cdots \det(E_1^T) \\ &= \det(E_1) \det(E_2) \cdots \det(E_k) \det(C) = \det(E_1 E_2 \cdots E_k C) = \det(A).\end{aligned}$$

Vi brukte her at $\det(C) = \det(C^T)$. Det er klart at dette holder hvis $C = I_n$. Hvis $C \neq I_n$ så har C en nullrad, slik at $\det(C) = 0$. Men da har C^T en nullsøyle, slik at også $\det(C^T) = 0$. ■

Fra dette kan vi utlede at vi også kan utvikle determinanten langs enhver søyle (gitt som oppgave).

Proposisjon 6.12 (Determinanter kan utvikles langs enhver søyle). For alle $1 \leq j \leq n$ har vi at

$$\det(A) = \sum_{i=1}^n (-1)^{i+j} a_{ij} \det(A^{(ij)}).$$

Fra Proposisjon 6.7 og 6.12 ser vi at minoren $\det(A^{(ij)})$ skal ha alternerende fortegn $(-1)^{i+j}$, uansett hvilken rad eller søyle vi utvikler langs. Dette kan vi også angi med et fortegnskjema:

$$\begin{pmatrix} + & - & + & - & \cdots \\ - & + & - & + & \cdots \\ + & - & + & - & \cdots \\ - & + & - & + & \cdots \\ \vdots & \vdots & \vdots & \vdots & \ddots \end{pmatrix}.$$

For praktisk bruk bør vi utvikle determinanten langs den raden/søylen som har flest nuller.

Eksempel 6.13. La oss se på 4×4 -matrisen $A = \begin{pmatrix} 2 & 0 & 2 & 3 \\ 1 & 0 & 0 & 2 \\ 3 & 1 & -1 & 3 \\ -1 & 0 & 2 & 3 \end{pmatrix}$. Vi ser fort at

den raden eller søyla med flest nuller er søyle 2. Utvikler vi langs denne får vi

$$\begin{vmatrix} 2 & 0 & 2 & 3 \\ 1 & 0 & 0 & 2 \\ 3 & 1 & -1 & 3 \\ -1 & 0 & 2 & 3 \end{vmatrix} = (-1)^{3+2} \begin{vmatrix} 2 & 2 & 3 \\ 1 & 0 & 2 \\ -1 & 2 & 3 \end{vmatrix} = - \left((-1)^{2+1} \begin{vmatrix} 2 & 3 \\ 2 & 3 \end{vmatrix} + (-1)^{2+3} 2 \begin{vmatrix} 2 & 2 \\ -1 & 2 \end{vmatrix} \right) \\ = -(-(-6-6) - 2(4+2)) = 12.$$

Her ekspanderte vi også langs rad 2, der det også var en null.

6.3 Forstørrelsesfaktorer for matriser(*)

La oss nå se på den geometriske tolkningen av determinant som en forstørrelsesfaktor. Når vi nå skal si hva vi mener med *volumet* $V(X)$ til en mengde X i $\mathbb{R}^n$ for $n > 3$, så la oss liste opp noen egenskaper $V(X)$ opplagt bør ha:

1. *enhetsboksene*

$$B = \{\mathbf{x} \in \mathbb{R}^n : 0 \leq x_i \leq 1 \forall i\}$$

bør ha volum 1.

2. Hvis X blir *translatert* med en vektor $\mathbf{r}$, så bør volumet forbli uendret. Sagt i formel: $V(X + \mathbf{r}) = V(X)$, der $X + \mathbf{r} = \{\mathbf{x} + \mathbf{r} : \mathbf{x} \in X\}$.
3. Hvis X blir *strukket ut* med en faktor r i retning $\mathbf{e}_i$ så bør volumet ganges opp med r . Sagt i formel: $V(X_{i,r}) = rV(X)$, der

$$X_{i,r} = \{\mathbf{x} : (x_1, \dots, x_{i-1}, x_i/r, x_{i+1}, \dots, x_n) \in X\}.$$

4. Hvis X_1 og X_2 er disjunkte mengder så bør $V(X_1 \cup X_2) = V(X_1) + V(X_2)$.
5. Volum bør kunne regnes ut som grunnflate ganger høyde, slik vi er vant til i $\mathbb{R}^2$ og $\mathbb{R}^3$. Men nå må vi tenke på grunnflaten som et volum i $\mathbb{R}^{n-1}$.

Bruker vi punkt 1. og 2., samt punkt 3 gjentatte ganger på en enhetsboks, så skal boksene

$$B_{\mathbf{r},\delta} = \{\mathbf{x} \in \mathbb{R}^n : r_i\delta \leq x_i \leq (r_i + 1)\delta \forall i\}$$

ha volum δ^n for alle valg av vektorer $\mathbf{r}$. Dette gir oss partisjonen

$$\mathbb{R}^n = \cup_{\mathbf{r}} B_{\mathbf{r},\delta}.$$

De fleste mengder i $\mathbb{R}^n$ kan tilnærmes vilkårlig godt ved hjelp av mange nok, og små nok, slike bokser (mer generelt kan vi definere volum som en grenseverdi

der $\delta \rightarrow 0$). La oss derfor begrense oss til områder som kan skrives som en union av endelig mange bokser. På grunn av punkt 4. må vi ha

$$V(\cup_i B_{\mathbf{r}_i, \delta}) = \sum_i \delta^n.$$

Hvis $X = \cup_i B_{\mathbf{r}_i, \delta}$, og A er inverterbar, så er det klart at $AX = A(\cup_i B_{\mathbf{r}_i, \delta}) = \cup_i A(B_{\mathbf{r}_i, \delta})$ gir en partisjon av AX i disjunkte delmengder.

Proposisjon 6.14. *La $X = \cup_i B_{\mathbf{r}_i, \delta}$ være en disjunkt union av bokser. Da er $V(AX) = V(AB)V(X)$.*

Dette betyr at $V(AB)$ fungerer som en *forstørrelsesfaktor* for A for områder X på formen $\cup_i B_{\mathbf{r}_i, \delta}$. Tanken er at, siden $V(AX) = V(AB)V(X)$ for slike X , og siden de fleste mengder kan tilnærmes vilkårlig godt med slike mengder, så vil også $V(AX) = V(AB)V(X)$ for mer generelle mengder X . Altså fungerer $V(AB)$ som en forstørrelsesfaktor mer generelt også!

Bevis. Siden $B_{\mathbf{r}, \delta} = \delta(B + \mathbf{r})$ så har vi at

$$AB_{\mathbf{r}, \delta} = A(\delta(B + \mathbf{r})) = \delta(AB + A\mathbf{r}).$$

Fra egenskap 2. og 3. følger at $V(AB_{\mathbf{r}, \delta}) = \delta^n V(AB)$. Bruker vi egenskap 4. får vi nå at

$$\begin{aligned} V(AX) &= V(A(\cup_i B_{\mathbf{r}_i, \delta})) = V(\cup_i AB_{\mathbf{r}_i, \delta}) = \sum_i V(AB_{\mathbf{r}_i, \delta}) \\ &= \sum_i \delta^n V(AB) = V(AB) \sum_i \delta^n = V(AB)V(X). \end{aligned}$$

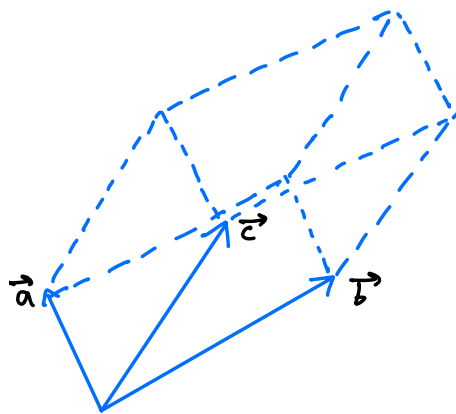
■

Siden $B = \{\sum_{i=1}^n x_i \mathbf{e}_i : 0 \leq x_i \leq 1 \forall i\}$ så er

$$AB = \left\{ \sum_{i=1}^n x_i A\mathbf{e}_i : 0 \leq x_i \leq 1 \forall i \right\} = \left\{ \sum_{i=1}^n x_i \mathbf{a}_i : 0 \leq x_i \leq 1 \forall i \right\},$$

der $\mathbf{a}_i$ er søyle i i A . Mer generelt, hvis $\mathbf{a}_1, \dots, \mathbf{a}_m$ er vektorer i $\mathbb{R}^n$, så kaller vi mengden

$$F(\mathbf{a}_1, \dots, \mathbf{a}_m) := \left\{ \sum_{i=1}^m x_i \mathbf{a}_i, 0 \leq x_i \leq 1 \forall i \right\}$$



Figur 6.1. Et parallelepiped i rommet.

for *parallelepipedet* utspent av $\{\mathbf{a}_i\}_{i=1}^m$ (parallellogrammer svarer til å sette $m = 2$). Våre forstørrelsesfaktorer er altså volumer av parallelepipeder. Figur 6.1 viser et parallelepiped utspent av tre vektorer $\mathbf{a}$, $\mathbf{b}$, og $\mathbf{c}$ i rommet.

Siden A kan skrives som $E_k \cdots E_1$, der E_i er elementære, skal vi se på hvordan volumer blir forstørret for hver slik. Det er 3 forskjellige typer elementære matriser vi må se på.

Proposisjon 6.15. Forstørrelsesfaktoren til en elementær matrise E er gitt ved

$$V(EB) = \begin{cases} 1 & \text{hvis } E \text{ er et radbytte.} \\ |s| & \text{hvis } E \text{ ganger en rad med } s. \\ 1 & \text{hvis } E \text{ legger til et multiplum av en rad til en annen.} \end{cases}$$

Bevis. Vi går gjennom de tre mulighetene.

1. Anta E bytter om rad i og j . Da er $EB = B$, slik at $V(EB) = 1$.
2. Hvis E ganger rad i med s så er $EB = \{\mathbf{x} : 0 \leq x_i \leq s, 0 \leq x_j \leq 1, j \neq i\}$, slik at $V(EB) = s$ på grunn av punkt 3..
3. Anta E er slik at c ganger rad n legges til rad $n-1$ (andre valg av rader vises tilsvarende), det vil si

$$E(x_1, \dots, x_{n-1}, x_n) = (x_1, \dots, x_{n-1} + cx_n, x_n).$$

For tverrsnittet der $x_n = k$ får vi

$$E(x_1, \dots, x_{n-1}, k) = (x_1, \dots, x_{n-1}, k) + (0, \dots, ck, 0),$$

det vil si at tverrsnittet der $x_n = k$ blir forskjøvet av E med en konstant vektor $(0, \dots, ck, 0)$, slik at alle tverrsnitt av EB på formen $x_n = k$ har samme areal som B . Dette betyr at både grunnflate og høyde ($x_n = 1$) er like for B og EB . Det følger da fra punkt 5. at $V(EB)$ er 1.

■

Samholder vi med Proposisjon 6.4 ser vi nå at $|\det(E)| = V(EB)$ for alle elementære matriser E , og fra Proposisjon 6.5 brukt gjentatte ganger følger at

$$\det(E_k \cdots E_1) = \det(E_k) \cdots \det(E_1).$$

Siden også

$$V(E_k \cdots E_1 B) = V(E_k B) \cdots V(E_1 B)$$

Så følger det fra $|\det(E_i)| = V(E_i B)$ at $|\det(A)| = V(AB)$, så lenge A kan skrives som et produkt av elementære matriser (d.v.s. at A er inverterbar). Vi har dermed vist følgende:

Proposisjon 6.16. Hvis A er inverterbar så er forstørrelsesfaktoren $V(AB)$ lik $|\det(A)|$.

For $n = 2$ sier dette spesielt at $\begin{vmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{vmatrix}$ er lik arealet til parallelogrammet utspent av $\mathbf{a}_1$ og $\mathbf{a}_2$. Hvis vi setter $\mathbf{a}_1 = |\mathbf{a}_1|(\cos\theta_1, \sin\theta_1)$, og $\mathbf{a}_2 = |\mathbf{a}_2|(\cos\theta_2, \sin\theta_2)$ så blir

$$\begin{aligned} \begin{vmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{vmatrix} &= \begin{vmatrix} |\mathbf{a}_1|\cos\theta_1 & |\mathbf{a}_1|\sin\theta_1 \\ |\mathbf{a}_2|\cos\theta_2 & |\mathbf{a}_2|\sin\theta_2 \end{vmatrix} = |\mathbf{a}_1||\mathbf{a}_2|(\cos\theta_1\sin\theta_2 - \sin\theta_1\cos\theta_2) \\ &= |\mathbf{a}_1||\mathbf{a}_2|\sin(\theta_2 - \theta_1) \end{aligned}$$

Dette utledet vi også i Oppgave 9 i Kapittel 1.

Forstørrelsesfaktoren til $A = E_k \cdots E_1$ fås nå ved å gange sammen bidragene fra de elementære matrisene.

6.4 Geometrisk tolkning av determinanten. Vektorproduktet

Setter vi

$$\mathbf{b} = (\det(A^{(11)}), -\det(A^{(12)}), \det(A^{(13)}), \dots, (-1)^{n+1} \det(A^{(1n)})), \quad (6.5)$$

så kan definisjonen av determinanten (6.1) skrives som skalarproduktet $\mathbf{a}_1 \cdot \mathbf{b}$ ($\mathbf{a}_i$ er her rad i i A). Dette gir oss følgende geometriske tolkning.

Proposisjon 6.17. *Vi har at*

1. $\mathbf{b}$ står ortogonalt på alle $\mathbf{a}_2, \dots, \mathbf{a}_n$,
2. $|\mathbf{b}| = V(F(\mathbf{a}_2, \dots, \mathbf{a}_n))$,
3. $|\mathbf{a}_1 \cdot \mathbf{b}| = V(AB) = |\det(A)|$.

Vi skal straks se på beviset for dette, men la oss først se litt nærmere på tilfellet $n = 3$. Da har vi at

$$\mathbf{b} = (a_{22}a_{33} - a_{23}a_{32}, -(a_{21}a_{33} - a_{23}a_{31}), a_{21}a_{32} - a_{22}a_{31}). \quad (6.6)$$

Dette har et eget navn:

Definisjon 6.18. *Uttrykket (6.6) kalles for vektorproduktet av $\mathbf{a}_2$ og $\mathbf{a}_3$. Vi skriver også $\mathbf{a}_2 \times \mathbf{a}_3$ for vektorproduktet av $\mathbf{a}_2$ og $\mathbf{a}_3$.*

Det er vanlig å skrive vektorproduktet som

$$\begin{vmatrix} \mathbf{e}_1 & \mathbf{e}_2 & \mathbf{e}_3 \\ a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & a_{33} \end{vmatrix} \\ = (a_{22}a_{33} - a_{23}a_{32})\mathbf{e}_1 - (a_{21}a_{33} - a_{23}a_{31})\mathbf{e}_2 + (a_{21}a_{32} - a_{22}a_{31})\mathbf{e}_3 \\ = (a_{22}a_{33} - a_{23}a_{32}, -(a_{21}a_{33} - a_{23}a_{31}), a_{21}a_{32} - a_{22}a_{31}),$$

det vil si at vi har satt inn vektorer i stedet for skalarer i første rad i matrisen.

Spesielt er $\mathbf{a}_2 \times \mathbf{a}_3$ ortogonal på $\mathbf{a}_2$ og $\mathbf{a}_3$, med lengde lik arealet til parallelogrammet utspent av de to vektorene. Disse to egenskapene karakteriserer ikke vektorproduktet fullstendig, for hvis $\mathbf{b}$ tilfredsstiller disse to egenskapene, så vil jo også $-\mathbf{b}$ gjøre det! Det viser seg imidlertid at $\mathbf{b} = \mathbf{a}_2 \times \mathbf{a}_3$ også tilfredsstiller det som kalles *høyrehåndsregelen*, noe $-\mathbf{b}$ ikke gjør:

Definisjon 6.19. La høyre pekefinger peke i samme retning som $\mathbf{a}_2$, og la langfingeren peke i samme retning som $\mathbf{a}_3$, på en slik måte at det er mindre enn 180 grader mellom de to. Tommelen vil nå peke i retning av vektorproduktet $\mathbf{a}_2 \times \mathbf{a}_3$, vinkelrett på både $\mathbf{a}_2$ og $\mathbf{a}_3$.

Proposisjon 6.20. $\mathbf{b} = \mathbf{a}_2 \times \mathbf{a}_3$ er den unike vektoren som står ortogonalt på både $\mathbf{a}_2$ og $\mathbf{a}_3$, har lengde lik arealet til parallellogrammet utspent av $\mathbf{a}_2$ og $\mathbf{a}_3$, og har retning gitt ved høyrehåndsregelen.

Vi skal ikke bevise høyrehåndsregelen. En 3×3 determinant kan altså skrives som $\det(A) = \mathbf{a}_1 \cdot (\mathbf{a}_2 \times \mathbf{a}_3)$, som på grunn av Proposisjon 6.16 også svarer til volumet av parallellepipedet utspent av $\mathbf{a}_1$, $\mathbf{a}_2$, og $\mathbf{a}_3$.

Legg merke til at $\mathbf{x} \times \mathbf{y} = \mathbf{0}$ hvis $\mathbf{x}$ og $\mathbf{y}$ er parallelle. Dette er kanskje enklest å se ved å observere at parallellogrammet de to utspenner da har areal 0. Lengden til vektorproduktet er da også 0, og da er $\mathbf{x} \times \mathbf{y} = \mathbf{0}$. Man kan også argumentere med at vektorproduktet da er en determinant der andre og tredje rad er like (etter en skalering), som er 0 på grunn av Korollar 6.6.

Eksempel 6.21. La $\mathbf{a}_1 = (1, 3, 2)$, og $\mathbf{a}_2 = (2, -1, 3)$. Da har vi at

$$\begin{aligned} \mathbf{a}_1 \times \mathbf{a}_2 &= \begin{vmatrix} \mathbf{e}_1 & \mathbf{e}_2 & \mathbf{e}_3 \\ 1 & 3 & 2 \\ 2 & -1 & 3 \end{vmatrix} \\ &= (3 \cdot 3 - 2 \cdot (-1))\mathbf{e}_1 - (1 \cdot 3 - 2 \cdot 2)\mathbf{e}_2 + (1 \cdot (-1) - 3 \cdot 2)\mathbf{e}_3 \\ &= (9 + 2, -(3 - 4), -1 - 6) = (11, 1, -7). \end{aligned}$$

Vektorproduktet mellom to vektorer kan også regnes ut ved hjelp av kommandoen `cross`:

```
>> a1 = [1,3,2]; a2 = [2,-1,3];
>> np.cross(a1,a2)
[11  1 -7]
```

Eksempel 6.22. I Eksempel 1.10 fant vi uttrykket for et plan gjennom et gitt punkt, med en gitt normalvektor. Hvis vi i stedet har tre punkter, så kan vi i stedet bruke vektorproduktet til å finne en normalvektor, og dermed et uttrykk for

planet: Hvis de tre punktene er $\mathbf{x}_0, \mathbf{x}_1$, og $\mathbf{x}_2$ så vil jo $\mathbf{x}_1 - \mathbf{x}_0$ og $\mathbf{x}_2 - \mathbf{x}_0$ være vektorer i dette planet, og da vil jo vektorproduktet av disse gi oss en normalvektor.

For å illustrere dette, ta de tre punktene $\mathbf{x}_0 = (4, -2, -1)$, $\mathbf{x}_1 = (1, 0, 0)$, og $\mathbf{x}_2 = (2, -1, 0)$. Vi får først at $\mathbf{x}_1 - \mathbf{x}_0 = (-3, 2, 1)$, og $\mathbf{x}_2 - \mathbf{x}_0 = (-2, 1, 1)$. Vektorproduktet av disse blir

$$\begin{vmatrix} \mathbf{e}_1 & \mathbf{e}_2 & \mathbf{e}_3 \\ -3 & 2 & 1 \\ -2 & 1 & 1 \end{vmatrix} = (2-1)\mathbf{e}_1 - (-3+2)\mathbf{e}_2 + (-3+4)\mathbf{e}_3 = \mathbf{e}_1 + \mathbf{e}_2 + \mathbf{e}_3 = (1, 1, 1).$$

Fra dette følger det at planet kan skrives på formen $x + y + z = 1$.

Bevis. (for Proposisjon 6.17) Vi viser først punkt 1.. Definer matrisen

$$A_0 = \begin{pmatrix} \mathbf{a}_1 \\ \mathbf{a}_2 \\ \mathbf{a}_3 \\ \vdots \\ \mathbf{a}_n \end{pmatrix},$$

der $\mathbf{a}_i$ er satt inn både i rad 1 og i . Siden en matrise med to like rader har determinant 0 (Korollar 6.6), så har vi at

$$0 = \det(A_0) = \mathbf{a}_i \cdot \mathbf{b} \text{ for } i = 2, \dots, n,$$

slik at $\mathbf{b}$ er ortogonal på alle $\{\mathbf{a}_i\}_{i=2}^n$.

For å se at $|\mathbf{b}| = V(F(\mathbf{a}_2, \dots, \mathbf{a}_n))$ (punkt 2.), observer først at projeksjonen av $\mathbf{a}_1$ ned på $\mathbf{b}$ er $\frac{\mathbf{a}_1 \cdot \mathbf{b}}{|\mathbf{b}|^2} \mathbf{b}$, som har lengde $\frac{|\mathbf{a}_1 \cdot \mathbf{b}|}{|\mathbf{b}|}$. Dette er lik avstanden fra $\mathbf{a}_1$ til planet utspent av $\mathbf{a}_2, \dots, \mathbf{a}_n$. Siden volum er lik grunnflate ganger høyde får vi at

$$V(AB) = V(F(\mathbf{a}_1, \dots, \mathbf{a}_n)) = \frac{|\mathbf{a}_1 \cdot \mathbf{b}|}{|\mathbf{b}|} V(F(\mathbf{a}_2, \dots, \mathbf{a}_n)).$$

Siden $|\det(A)| = V(AB)$ (Proposisjon 6.16) følger at

$$|\mathbf{a}_1 \cdot \mathbf{b}| = \frac{|\mathbf{a}_1 \cdot \mathbf{b}|}{|\mathbf{b}|} V(F(\mathbf{a}_2, \dots, \mathbf{a}_n))$$

slik at $|\mathbf{b}| = V(F(\mathbf{a}_2, \dots, \mathbf{a}_n))$.

Punkt 3. følger rett fra definisjonen (6.1) av determinanten og sammenhengen vi har vist mellom volum og determinant. ■

Resultatet over er kanskje lettest å tolke på (den opplagt sanne) formen

$$|\mathbf{a}_1 \cdot \mathbf{b}| = \frac{|\mathbf{a}_1 \cdot \mathbf{b}|}{|\mathbf{b}|} |\mathbf{b}|.$$

På venstre side står da volumet til parallellepipedet utspent av $\mathbf{a}_1, \mathbf{a}_2, \dots, \mathbf{a}_n$ ($V(AB)$). På høyre side er $\frac{|\mathbf{a}_1 \cdot \mathbf{b}|}{|\mathbf{b}|}$ høyden over planet utspent av $\mathbf{a}_2, \mathbf{a}_3, \dots, \mathbf{a}_n$, mens $|\mathbf{b}|$ er overflaten til parallellepipedet utspent av $\mathbf{a}_2, \mathbf{a}_3, \dots, \mathbf{a}_n$ ($V(F(\mathbf{a}_2, \dots, \mathbf{a}_n))$). Med andre ord: volum er lik grunnflate ganget med høyde!

Oppgaver for Kapittel 6

1. Regn ut

(a). $\begin{vmatrix} 4 & -1 \\ 2 & 3 \end{vmatrix}$

(b). $\begin{vmatrix} 2 & 0 & 1 \\ 3 & -1 & 7 \\ 0 & 2 & 5 \end{vmatrix}$

(c). $(2, 0, 4) \times (-1, 2, 1)$.

2. La $\mathbf{x}$, $\mathbf{y}$, og $\mathbf{z}$ være vektorer i $\mathbb{R}^3$. Vis at

1. $\mathbf{x} \times \mathbf{y} = -\mathbf{y} \times \mathbf{x}$.

2. For alle skalarer s så er $\mathbf{x} \times (s\mathbf{y}) = (s\mathbf{x}) \times \mathbf{y} = s(\mathbf{x} \times \mathbf{y})$.

3. $\mathbf{x} \times (\mathbf{y} + \mathbf{z}) = \mathbf{x} \times \mathbf{y} + \mathbf{x} \times \mathbf{z}$.

3. La $\mathbf{x}$ være en gitt vektor i $\mathbb{R}^3$. Fra 2. og 3. i forrige oppgave følger det at avbildningen $\mathbf{y} \rightarrow \mathbf{x} \times \mathbf{y}$ (som går fra $\mathbb{R}^3$ til $\mathbb{R}^3$) er lineær. Hva er matrisen til denne lineære avbildningen?

4 (Prøveeksamen 2024). Regn ut determinanten til

$$A = \begin{pmatrix} 2 & -1 & 1 \\ 0 & 1 & 4 \\ 1 & -2 & -6 \end{pmatrix}.$$

på to forskjellige måter: Ved hjelp av stegene i Gausseliminasjon (disse skrev du ned i Oppgave 1 i Seksjon 4), og ved å utvikle determinanten langs første søyle.

5. Vis at determinanten til en vandermondematrix med generatorer $x_0, \dots, x_n$ er

$$\prod_{i>j} (x_i - x_j).$$

Hint: skriv ned alle radoperasjonene som endrer determinanten i Eksempel 4.7.

6. Bevis Proposisjon 6.7.

7. Bevis punkt 2 og 3 av Proposisjon 6.5.

Hint: Bruk punkt 1 av Proposisjon 6.5 og Korollar 6.6.

8. Anta A er inverterbar. Vis at $\det(A^{-1}) = 1/\det(A)$.

9. Anta U er ortogonal, det vil si at U er reell, og at $U^T U = I_n$. Vis at $\det(U)$ er 1 eller -1 .

10. Bevis Proposisjon 6.12.

11. La A og B være kvadratiske matriser, og sett $D = \left(\begin{array}{c|c} A & C \\ \hline \mathbf{0} & B \end{array} \right)$, der $\mathbf{0}$ og C har dimensjoner slik at D blir kvadratisk. Vis at $\det(D) = \det(A)\det(B)$.

12 (**Cramer's regel**). Anta A er inverterbar, og at $\mathbf{x}$ løser systemet $A\mathbf{x} = \mathbf{b}$. La også $A_i(\mathbf{b})$ være matrisen A med søyle i erstattet med $\mathbf{b}$.

(a). Vis at

$$A \begin{pmatrix} 1 & 0 & \cdots & 0 & x_1 & 0 & \cdots & 0 & 0 \\ 0 & 1 & \cdots & 0 & x_2 & 0 & \cdots & 0 & 0 \\ \vdots & \vdots & \ddots & \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & \cdots & 0 & x_{n-1} & 0 & \cdots & 1 & 0 \\ 0 & 0 & \cdots & 0 & x_n & 0 & \cdots & 0 & 1 \end{pmatrix} = A_i(\mathbf{b})$$

Vektoren $\mathbf{x}$ kommer i søyle i .

(b). Forklar at determinanten til den store matrisen over er x_i . Utled fra dette at $x_i = \det(A_i(\mathbf{b}))/\det(A)$.

Dette gjør at vi kan uttrykke løsningen av et likningssystem ved hjelp av determinanter. Dette kalles også for *Cramer's regel*.

(c). La $A = \begin{pmatrix} a & b \\ c & d \end{pmatrix}$ være en generell inverterbar 2×2 -matrise. Bruk Cramer's

regel til å vise at $A^{-1} = \frac{1}{ad-bc} \begin{pmatrix} d & -b \\ -c & a \end{pmatrix}$

KAPITTEL 7

Eigenverdier og egenvektorer

I dette kapitlet skal vi jobbe med egenvektorer og eigenverdier, som vi definerer som følger.

Definisjon 7.1. La A være en $n \times n$ -matrise. λ kalles en eigenverdi for A hvis det finnes en vektor $\mathbf{x} \neq \mathbf{0} \in \mathbb{R}^n$ som tilfredsstiller $A\mathbf{x} = \lambda\mathbf{x}$. $\mathbf{x}$ kalles da for en tilhørende egenvektor.

λ er en eigenverdi hvis $(A - \lambda I)\mathbf{x} = \mathbf{0}$ har en løsning $\mathbf{x} \neq \mathbf{0}$, som er det samme som at $A - \lambda I$ ikke er inverterbar, som igjen er det samme som at

$$\det(A - \lambda I) = 0. \quad (7.1)$$

Dette kalles den *karakteristiske likningen* til A . Bruker vi den rekursive definisjonen av determinanten gir dette et n 'te grads polynom i λ , kalt det *karakteristiske polynomet* til A . Eigenverdiene kan altså regnes ut ved å finne nullpunkter i et polynom. I prinsippet kan man også finne eigenverdiene ved å radredusere $A - \lambda I$, men problemet er at de faktiske radoperasjonene vi da gjør vil avhenge av den konkrete verdien for λ . Derfor vil vi holde oss til den rekursive definisjonen av determinanten her ¹

Når vi har funnet eigenverdiene så kan vi finne en egenvektor tilhørende en gitt λ ved å løse systemet $(A - \lambda I)\mathbf{x} = \mathbf{0}$, som vi kan gjøre ved å radredusere $A - \lambda I$. Egenvektoren du finner er ikke unike - vi kan gange en egenvektor med et tall, og den nye vektoren er da også en egenvektor. Av og til spør man om det generelle

¹I praksis finner datamaskinen eigenverdier ved hjelp av en helt annen algoritme.

uttrykket for en egenvektor tilhørende en gitt egenverdi. Andre ganger spør man kun om en spesiell egenvektor. Det er da vanlig å velge en som er på enklest mulig form, typisk med heltallige komponenter og med minst mulige tall i seg. Noen eksempler vil gjøre dette klarere.

Eksempel 7.2. La oss først se på den lineære avbildningen T som speiler om linjen $y = 0$ i $\mathbb{R}^2$. Vi har at $T(\mathbf{e}_1) = \mathbf{e}_1$ og $T(\mathbf{e}_2) = -\mathbf{e}_2$ (vi ser allerede her at $\mathbf{e}_1$ og $\mathbf{e}_2$ er egenvektorer), slik at matrisen til T er $A = \begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix}$. Den karakteristiske likningen er $(1 - \lambda)(-1 - \lambda) = 0$, slik at egenverdiene er $\lambda_1 = -1$, $\lambda_2 = 1$.

Egenvektor for $\lambda_1 = -1$:

$$A + I = \begin{pmatrix} 2 & 0 \\ 0 & 0 \end{pmatrix} \sim \begin{pmatrix} 1 & 0 \\ 0 & 0 \end{pmatrix}$$

Den første raden her sier at en egenvektor må oppfylle $x_1 = 0$. Den generelle egenvektoren er derfor $(0, x_2)$ med x_2 en fri variabel. Vi får spesielt den enkle egenvektoren $(0, 1)$.

Egenvektor for $\lambda_2 = 1$:

$$A - I = \begin{pmatrix} 0 & 0 \\ 0 & -2 \end{pmatrix} \sim \begin{pmatrix} 0 & 1 \\ 0 & 0 \end{pmatrix}$$

Den første raden her sier at en egenvektor må oppfylle $x_2 = 0$. Spesielt har vi egenvektoren $(1, 0)$.

Eksempel 7.3. La oss ta for oss et litt større eksempel. Vi setter

$$A = \begin{pmatrix} -32 & -98 & -165 \\ 4 & 24 & 32 \\ 8 & 16 & 32 \end{pmatrix}.$$

Vi får først at

$$\begin{aligned} \det(A - \lambda I) &= \begin{vmatrix} -32 - \lambda & -98 & -165 \\ 4 & 24 - \lambda & 32 \\ 8 & 16 & 32 - \lambda \end{vmatrix} \\ &= (-32 - \lambda) \begin{vmatrix} 24 - \lambda & 32 \\ 16 & 32 - \lambda \end{vmatrix} + 98 \begin{vmatrix} 4 & 32 \\ 8 & 32 - \lambda \end{vmatrix} - 165 \begin{vmatrix} 4 & 24 - \lambda \\ 8 & 16 \end{vmatrix} \\ &= (-32 - \lambda)(\lambda^2 - 56\lambda + 256) + 98(-4\lambda - 128) - 165(8\lambda - 128) \\ &= (-\lambda^3 + 24\lambda^2 + 1536\lambda - 8192) + (-392\lambda - 12544) + (-1320\lambda + 21120) \\ &= -\lambda^3 + 24\lambda^2 - 176\lambda + 384. \end{aligned}$$

De fleste av oss husker ikke formelen for å løse en tredjegradslikning, så her er det ikke opplagt hvordan vi skal fortsette. Man kan finne røttene til et polynom ved hjelp av kommandoen `roots`. Denne tar koeffisientene til polynomet som input:

```
>> np.roots(np.array([-1,24,-176,384]))
[12.  8.  4.]
```

Alternativt kan vi finne noen av røttene ved å prøve oss frem. Da vil vi sannsynligvis først finne roten $\lambda_1 = 4$. De to andre røttene kan man finne ved først å gjøre polynomdivisjon

$$\begin{array}{r}
 (-\lambda^3 + 24\lambda^2 - 176\lambda + 384) : (-\lambda + 4) = \lambda^2 - 20\lambda + 96 \\
 -(-\lambda^3 + 4\lambda^2) \\
 \hline
 20\lambda^2 - 176\lambda \\
 -(20\lambda^2 - 80\lambda) \\
 \hline
 -96\lambda + 384 \\
 -(-96\lambda + 384) \\
 \hline
 0,
 \end{array}$$

slik at

$$-\lambda^3 + 24\lambda^2 - 176\lambda + 384 = (-\lambda + 4)(\lambda^2 - 20\lambda + 96),$$

og deretter bruke formelen for andregradslikningen for å finne røttene til $\lambda^2 - 20\lambda + 96 = 0$. Dette gir

$$\lambda = \frac{20 \pm \sqrt{400 - 384}}{2} = \frac{20 \pm 4}{2} = 10 \pm 2,$$

slik at vi også får røttene $\lambda_2 = 8$ og $\lambda_3 = 12$. La oss så finne egenvektorene

- For $\lambda_1 = 4$ får vi

$$\begin{aligned}
 A - \lambda_1 I &= \begin{pmatrix} -36 & -98 & -165 \\ 4 & 20 & 32 \\ 8 & 16 & 28 \end{pmatrix} \sim \begin{pmatrix} 1 & 5 & 8 \\ -36 & -98 & -165 \\ 8 & 16 & 28 \end{pmatrix} \sim \begin{pmatrix} 1 & 5 & 8 \\ 0 & 82 & 123 \\ 0 & -24 & -36 \end{pmatrix} \\
 &\sim \begin{pmatrix} 1 & 5 & 8 \\ 0 & 1 & 3/2 \\ 0 & 0 & 0 \end{pmatrix} \sim \begin{pmatrix} 1 & 0 & 1/2 \\ 0 & 1 & 3/2 \\ 0 & 0 & 0 \end{pmatrix}
 \end{aligned}$$

Den første raden sier at $x_1 + \frac{1}{2}x_3 = 0$, slik at $x_1 = -\frac{1}{2}x_3$. Den andre raden sier at $x_2 + \frac{3}{2}x_3 = 0$, slik at $x_2 = -\frac{3}{2}x_3$. Den generelle egenvektoren blir dermed

$$(x_1, x_2, x_3) = \left(-\frac{1}{2}x_3, -\frac{3}{2}x_3, x_3\right) = x_3 \left(-\frac{1}{2}, -\frac{3}{2}, 1\right)$$

Setter vi $x_3 = 2$ får vi den spesielle egenvektoren $\mathbf{v}_1 = (-1, -3, 2)$.

- For $\lambda_2 = 8$ får vi

$$\begin{aligned} A - \lambda_2 I &= \begin{pmatrix} -40 & -98 & -165 \\ 4 & 16 & 32 \\ 8 & 16 & 24 \end{pmatrix} \sim \begin{pmatrix} 1 & 4 & 8 \\ -40 & -98 & -165 \\ 8 & 16 & 24 \end{pmatrix} \sim \begin{pmatrix} 1 & 4 & 8 \\ 0 & 62 & 155 \\ 0 & -16 & -40 \end{pmatrix} \\ &\sim \begin{pmatrix} 1 & 4 & 8 \\ 0 & 1 & 5/2 \\ 0 & 0 & 0 \end{pmatrix} \sim \begin{pmatrix} 1 & 0 & -2 \\ 0 & 1 & 5/2 \\ 0 & 0 & 0 \end{pmatrix} \end{aligned}$$

Her får vi $\mathbf{v}_2 = (4, -5, 2)$ som en spesiell egenvektor.

- For $\lambda_3 = 12$ får vi

$$\begin{aligned} A - \lambda_3 I &= \begin{pmatrix} -44 & -98 & -165 \\ 4 & 12 & 32 \\ 8 & 16 & 20 \end{pmatrix} \sim \begin{pmatrix} 1 & 3 & 8 \\ -44 & -98 & -165 \\ 8 & 16 & 20 \end{pmatrix} \sim \begin{pmatrix} 1 & 3 & 8 \\ 0 & 34 & 187 \\ 0 & -8 & -44 \end{pmatrix} \\ &\sim \begin{pmatrix} 1 & 3 & 8 \\ 0 & 1 & 11/2 \\ 0 & 0 & 0 \end{pmatrix} \sim \begin{pmatrix} 1 & 0 & -17/2 \\ 0 & 1 & 11/2 \\ 0 & 0 & 0 \end{pmatrix} \end{aligned}$$

slik at $\mathbf{v}_3 = (17, -11, 2)$ er en spesiell egenvektor.

La oss verifisere at vi ikke har noen regnefeil her:

$$\begin{aligned} \bullet \begin{pmatrix} -32 & -98 & -165 \\ 4 & 24 & 32 \\ 8 & 16 & 32 \end{pmatrix} \begin{pmatrix} -1 \\ -3 \\ 2 \end{pmatrix} &= \begin{pmatrix} 32 + 294 - 330 \\ -4 - 72 + 64 \\ -8 - 48 + 64 \end{pmatrix} = \begin{pmatrix} -4 \\ -12 \\ 8 \end{pmatrix} = 4 \begin{pmatrix} -1 \\ -3 \\ 2 \end{pmatrix} \\ \bullet \begin{pmatrix} -32 & -98 & -165 \\ 4 & 24 & 32 \\ 8 & 16 & 32 \end{pmatrix} \begin{pmatrix} 4 \\ -5 \\ 2 \end{pmatrix} &= \begin{pmatrix} -128 + 490 - 330 \\ 16 - 120 + 64 \\ 32 - 80 + 64 \end{pmatrix} = \begin{pmatrix} 32 \\ -40 \\ 16 \end{pmatrix} = 8 \begin{pmatrix} 4 \\ -5 \\ 2 \end{pmatrix} \\ \bullet \begin{pmatrix} -32 & -98 & -165 \\ 4 & 24 & 32 \\ 8 & 16 & 32 \end{pmatrix} \begin{pmatrix} 17 \\ -11 \\ 2 \end{pmatrix} &= \begin{pmatrix} -544 + 1078 - 330 \\ 68 - 264 + 64 \\ 136 - 176 + 64 \end{pmatrix} = \begin{pmatrix} 204 \\ -132 \\ 24 \end{pmatrix} = 12 \begin{pmatrix} 17 \\ -11 \\ 2 \end{pmatrix}. \end{aligned}$$

Eksemplet over er konstruert slik at det lar seg løse uten altfor stygge tall. I praksis vil ikke egenverdiene være hele tall eller brøker, og da vil resten av utregningen bli rammet av avrundingsfeil. I praktisk bruk vil vi stort sett aldri gjøre utregninger som over for hånd - det er for tidkrevende, og det er veldig lett å gjøre regnefeil. I stedet kan vi bruke kommandoen eig. Denne tar en matrise som input, og returnerer to matriser:

- en matrise med egenvektorene til inputmatrisen som søyler,
- en diagonalmatrise der egenverdiene til inputmatrisen står diagonalen (matchende egenvektor finner man i tilsvarende søyle i den første matrisen).

Tester vi eig på eksemplet over får vi

```
>> A = np.array([[-32,-98,-165],[4,24,32],[8,16,32]])
>> D,P = np.linalg.eig(A)
[12.  8.  4.]
[[ 0.83550442 -0.59628479 -0.26726124]
 [-0.54062051  0.74535599 -0.80178373]
 [ 0.09829464 -0.2981424  0.53452248]]
```

Vi observerer flere ting her. For det første returneres desimaltall, ikke hele tall. Dette skyldes blant annet at en numerisk algoritme for å finne røttene i det karakteristiske polynomet blir brukt. For det andre, se på følgende:

```
>> P.T@P
[[ 1.          -0.93045911  0.26270347]
 [-0.93045911  1.          -0.5976143 ]
 [ 0.26270347 -0.5976143  1.          ]]
```

$P^T P$ returnerer alle mulige skalarprodukter mellom søylene i P . Dette viser derfor at søylene i P er skalert slik at de har lengde 1, og samtidig at egenvektorene ikke er ortogonale (siden det ikke er nuller utenfor diagonalen). Regner du ut den reduserte trappeformen til P vil du få identitetsmatrisen, slik at egenvektorene i det minste er lineært uavhengige. Dette skal vi straks bevise at gjelder mer generelt for egenvektorer tilhørende forskjellige egenverdier.

Vi vil jo gjerne få tak i de eksakte verdiene for egenverdiene og egenvektorene også. Dette kan vi få til ved å konvertere til symbolske størrelser:

```
>> A = sym.Matrix([[-32,-98,-165],[4,24,32],[8,16,32]])
>> A.eigenvals()
{12: 1, 8: 1, 4: 1}
>> A.eigenvects()
[(4, 1, [Matrix([[-1/2],[-3/2],[ 1]])]),
 (8, 1, [Matrix([[ 2],[-5/2],[ 1]])]),
 (12, 1, [Matrix([[ 17/2],[-11/2],[ 1]])])]
```

Vi ser at komponentene i egenvektorene er brøker, og det er ingen desimaltall. Vi ser nå de eksakt samme egenverdiene som vi fikk i eksemplet, og hvis vi ganger opp P med 2 så får vi de samme egenvektorene.

Tenk deg at vi har en stor matrise A og en vektor $\mathbf{x}_0$, og at vi skal regne ut $A^{100}\mathbf{x}_0$. En direkte utregning vil da være krevende, men tenk deg at vi allerede har regnet ut egenvektorene $\mathbf{v}_i$ og egenverdiene λ_i til A , og at egenvektorene utgjør en basis for $\mathbb{R}^n$. Da kan vi finne verdier c_i slik at $\mathbf{x}_0 = \sum_{i=1}^n c_i \mathbf{v}_i$. Fra dette kan vi regne ut

$$A^{100}\mathbf{x}_0 = \sum_{i=1}^n c_i \lambda_i^{100} \mathbf{v}_i.$$

Hvis egenverdiene her er ≤ 1 så kan vi få en grenseverdi her. La oss se på et eksempel rundt dette.

Eksempel 7.4. Vi vil studere utviklingen av en sykdom i en befolkning, og lar x_n være antall syke etter n dager, og y_n antall friske etter n dager. Vi antar at det er 1000 mennesker til sammen, slik at $x_n + y_n = 1000$ for alle n . Vi antar også

1. Hvis en er syk i dag, så er det 40% sjanse for at han er syk også i morgen.
2. Hvis en er frisk i dag så er det 20% sjanse for at han er syk i morgen.

La oss først finne en overgangsmatrise A slik at $\begin{pmatrix} x_{n+1} \\ y_{n+1} \end{pmatrix} = A \begin{pmatrix} x_n \\ y_n \end{pmatrix}$.

- Antall syke på en dag er 40% av de som var syke dagen før, pluss 20% av de som friske dagen før. Dette gir at $x_{n+1} = 0.4x_n + 0.2y_n$.
- De resterende $0.6x_n + 0.8y_n$ svarer til de som blir syke.

Dette gir at $A = \begin{pmatrix} 0.4 & 0.2 \\ 0.6 & 0.8 \end{pmatrix}$. La oss så finne egenverdiene og egenvektorene til A :

$$\det(A - \lambda I) = \begin{vmatrix} 0.4 - \lambda & 0.2 \\ 0.6 & 0.8 - \lambda \end{vmatrix} = (0.4 - \lambda)(0.8 - \lambda) - 0.12 = \lambda^2 - 1.2\lambda + 0.2.$$

Røttene til dette polynomet er $\lambda = \frac{1.2 \pm \sqrt{1.44 - 0.8}}{2} = 0.6 \pm 0.4$, slik at egenverdiene er $\lambda_1 = 0.2$ og $\lambda_2 = 1$.

- For $\lambda_1 = 0.2$ får vi

$$A - 0.2I = \begin{pmatrix} 0.2 & 0.2 \\ 0.6 & 0.6 \end{pmatrix} \sim \begin{pmatrix} 1 & 1 \\ 0 & 0 \end{pmatrix}$$

En generell egenvektor får vi ved å sette $x_1 = -x_2$. Setter vi $x_2 = 1$ får vi den spesielle egenvektoren $\mathbf{v}_1 = (-1, 1)$.

- For $\lambda_2 = 1$ får vi

$$A - I = \begin{pmatrix} -0.6 & 0.2 \\ 0.6 & -0.2 \end{pmatrix} \sim \begin{pmatrix} 1 & -1/3 \\ 0 & 0 \end{pmatrix}$$

En generell egenvektor får vi ved å sette $x_1 = x_2/3$. Setter vi $x_2 = 3$ får vi den spesielle egenvektoren $\mathbf{v}_2 = (1, 3)$.

La nå $\mathbf{x}_0 = \begin{pmatrix} 0 \\ 1000 \end{pmatrix}$ være starttilstanden der alle er friske. La oss finne grenseverdien $\lim_{n \rightarrow \infty} \mathbf{x}_n$. Vi uttrykker så $\mathbf{x}_0$ med $\mathbf{v}_1$ og $\mathbf{v}_2$:

$$(\mathbf{v}_1 \quad \mathbf{v}_2 \quad \mathbf{x}_0) = \begin{pmatrix} -1 & 1 & 0 \\ 1 & 3 & 1000 \end{pmatrix} \sim \begin{pmatrix} 1 & 3 & 1000 \\ 0 & 4 & 1000 \end{pmatrix} \sim \begin{pmatrix} 1 & 0 & 250 \\ 0 & 1 & 250 \end{pmatrix},$$

slik at $\mathbf{x}_0 = 250\mathbf{v}_1 + 250\mathbf{v}_2$. Vi får til slutt

$$\mathbf{x}_n = A^n \mathbf{x}_0 = 250 \cdot 0.2^n \mathbf{v}_1 + 250 \cdot 1^n \mathbf{v}_2 = 250 \begin{pmatrix} 1 - 0.2^n \\ 3 + 0.2^n \end{pmatrix}.$$

Det følger at $\lim_{n \rightarrow \infty} \mathbf{x}_n = (250, 750)$. Dette er en *likevektstilstand*, det vil si at

$$A \begin{pmatrix} 250 \\ 750 \end{pmatrix} = \begin{pmatrix} 0.4 & 0.2 \\ 0.6 & 0.8 \end{pmatrix} \begin{pmatrix} 250 \\ 750 \end{pmatrix} = \begin{pmatrix} 100 + 150 \\ 150 + 600 \end{pmatrix} = \begin{pmatrix} 250 \\ 750 \end{pmatrix}.$$

Matrisene i eksemplene over hadde distinkte egenverdier, og det viste seg at $\mathbb{R}^n$ da hadde en basis av egenvektorer. Det viser seg dette også gjelder mer generelt:

Proposisjon 7.5. La $\mathbf{v}_1, \dots, \mathbf{v}_k$ være egenvektorer som svarer til forskjellige egenverdier. Da er $\mathbf{v}_1, \dots, \mathbf{v}_k$ lineært uavhengige.

Bevis. Anta for motsigelse at $\mathbf{v}_1, \dots, \mathbf{v}_k$ er lineært avhengige. Da finnes det en lineær avhengighetsrelasjon

$$c_1 \mathbf{v}_1 + c_2 \mathbf{v}_2 + \dots + c_k \mathbf{v}_k = \mathbf{0}.$$

Da finnes det også en slik relasjon med færrest mulig ikkenuller, la oss si $r \geq 2$ ikkenuller. Ved evt. å stokke om på vektorene kan vi anta at de r første vektorene har koeffisient forskjellig fra 0, det vil si at

$$c_1 \mathbf{v}_1 + c_2 \mathbf{v}_2 + \dots + c_r \mathbf{v}_r = \mathbf{0},$$

der alle $c_1, \dots, c_r$ er $\neq 0$. Ganger vi denne med $-\lambda_1$ får vi

$$-c_1\lambda_1\mathbf{v}_1 - c_2\lambda_1\mathbf{v}_2 - \dots - c_r\lambda_1\mathbf{v}_r = \mathbf{0}, \quad (7.2)$$

Vi har også at

$$A(c_1\mathbf{v}_1 + c_2\mathbf{v}_2 + \dots + c_r\mathbf{v}_r) = c_1\lambda_1\mathbf{v}_1 + c_2\lambda_2\mathbf{v}_2 + \dots + c_r\lambda_r\mathbf{v}_r = \mathbf{0} \quad (7.3)$$

Legger vi sammen 7.2 og 7.3 får vi at

$$c_2(\lambda_2 - \lambda_1)\mathbf{v}_2 + c_3(\lambda_3 - \lambda_1)\mathbf{v}_3 + \dots + c_r(\lambda_r - \lambda_1)\mathbf{v}_r = \mathbf{0}.$$

Dette har gitt oss en enda kortere lineær avhengighetsrelasjon, som gir oss motsigelsen vi var ute etter. ■

Hvis en matrise ikke har distinkte egenverdier så kan to ting skje:

1. $\mathbb{R}^n$ har ikke en basis av egenvektorer for A .
2. $\mathbb{R}^n$ har en basis av egenvektorer for A .

La oss vise at begge ting kan inntreffe.

Eksempel 7.6. Vi ser på $A = \begin{pmatrix} 1 & -2 \\ 2 & -3 \end{pmatrix}$. Da er

$$\det(A - \lambda I) = \begin{vmatrix} 1 - \lambda & -2 \\ 2 & -3 - \lambda \end{vmatrix} = (1 - \lambda)(-3 - \lambda) + 4 = \lambda^2 + 2\lambda + 1 = (\lambda + 1)^2.$$

$\lambda = -1$ er derfor eneste egenverdi. Egenvektorene for denne finner vi ved radredusering:

$$A + I = \begin{pmatrix} 2 & -2 \\ 2 & -2 \end{pmatrix} \sim \begin{pmatrix} 1 & -1 \\ 0 & 0 \end{pmatrix},$$

slik at $\mathbf{v} = (1, 1)$ er en spesiell egenvektor. Denne utspenner ikke $\mathbb{R}^2$, slik at 1. over er tilfelle.

Eksempel 7.7. La oss sette $A = I_2$. Det karakteristiskepolynomet blir nå $(1 - \lambda)^2$, slik at $\lambda = 1$ er eneste egenverdi. Siden $A - I = 0$ så er både x_1 og x_2 frie variable, slik at alle vektorer er egenvektorer. Spesielt utgjør $\{\mathbf{e}_1, \mathbf{e}_2\}$ en basis av egenvektorer, slik at 2. over er tilfelle.

En matrise kan også ha komplekse egenverdier og egenvektorer.

Eksempel 7.8. Vi ser på $A = \begin{pmatrix} 2 & -8 \\ 2 & 2 \end{pmatrix}$. Da er

$$\det(A - \lambda I) = \begin{vmatrix} 2 - \lambda & -8 \\ 2 & 2 - \lambda \end{vmatrix} = (2 - \lambda)^2 + 16,$$

som er 0 når $2 - \lambda = \pm 4i$, det vil si når $\lambda = 2 \pm 4i$.

Så finner vi en egenvektor for $\lambda_1 = 2 + 4i$. Vi har at

$$A - (2 + 4i)I = \begin{pmatrix} -4i & -8 \\ 2 & -4i \end{pmatrix} \sim \begin{pmatrix} 1 & -2i \\ 0 & 0 \end{pmatrix}.$$

Dette gir $\mathbf{v}_1 = (2i, 1)$ som spesiell egenvektor.

Egenvektor for $\lambda_1 = 2 - 4i$. Vi har at

$$A - (2 - 4i)I = \begin{pmatrix} 4i & -8 \\ 2 & 4i \end{pmatrix} \sim \begin{pmatrix} 1 & 2i \\ 0 & 0 \end{pmatrix}.$$

Dette gir $\mathbf{v}_1 = (-2i, 1)$ som spesiell egenvektor.

I eksemplet over så vi at egenverdiene og egenvektorene var konjugerte av hverandre. Dette gjelder for alle reelle matriser. For å se dette, anta λ er en egenverdi med tilhørende egenvektor $\mathbf{x}$. Da er

$$A\mathbf{x} = \lambda\mathbf{x} \Rightarrow \overline{A\mathbf{x}} = \overline{\lambda\mathbf{x}} \Rightarrow \overline{A}\overline{\mathbf{x}} = \overline{\lambda}\overline{\mathbf{x}} = \lambda\overline{\mathbf{x}}.$$

Dette viser at $\overline{\lambda}$ også er en egenverdi, med $\overline{\mathbf{x}}$ som tilhørende egenvektor.

Eksempel 7.9. La oss se på rotasjonsmatrisen $A = \begin{pmatrix} \cos\theta & -\sin\theta \\ \sin\theta & \cos\theta \end{pmatrix}$. Vi finner først egenverdiene.

$$\det(A - \lambda I) = \begin{vmatrix} \cos\theta - \lambda & -\sin\theta \\ \sin\theta & \cos\theta - \lambda \end{vmatrix} = (\cos\theta - \lambda)^2 + \sin^2\theta,$$

som er 0 når $\lambda = \cos\theta \pm i\sin\theta$. Vi finner så egenvektoren for $\lambda = \cos\theta + i\sin\theta$ ved å radredusere.

$$A - (\cos\theta + i\sin\theta)I = \begin{pmatrix} -i\sin\theta & -\sin\theta \\ \sin\theta & -i\sin\theta \end{pmatrix} \sim \begin{pmatrix} 1 & -i \\ 0 & 0 \end{pmatrix},$$

slik at tilhørende egenvektor blir $(i, 1)$. Siden rotasjonsmatrisen er reell følger det også at $\lambda = (-i, 1)$ er en egenvektor for egenverdien $\cos\theta - i\sin\theta$.

Oppgaver for Kapittel 7

1. Vi definerer $A = \begin{pmatrix} 0.7 & 0.6 \\ 0.3 & 0.4 \end{pmatrix}$.

(a). Finn egenvektorer/egenverdier til A .

(b). Skriv vektoren $\mathbf{x}_0 = (1, 5)$ som en sum av egenvektorer, og regn ut grenseverdien $\lim_{n \rightarrow \infty} A^n \mathbf{x}_0$.

2. Finn egenverdiene og egenvektorene til matrisen $A = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 3 & 1 \\ 0 & 1 & 3 \end{pmatrix}$

3. Finn egenverdier med tilhørende egenvektorer for matrisen $A = \begin{pmatrix} 0 & -4 \\ 1 & 0 \end{pmatrix}$.

4. Finn egenverdier med tilhørende egenvektorer for $A = \begin{pmatrix} 6 & 2 \\ 0 & 6 \end{pmatrix}$. Finnes det en basis for $\mathbb{R}^2$ som består av egenvektorer for A ?

5. Anta både $\mathbf{x}_1$ og $\mathbf{x}_2$ er egenvektorer for A for samme egenverdi λ . Vis at $\mathbf{x}_1 + \mathbf{x}_2$ også er en egenvektor for A med egenverdi λ . Vi vet også at $c\mathbf{x}_1$ er en egenvektor for A med egenverdi λ . Det følger fra dette at alle egenvektorer for A med egenverdi λ er et underrom av $\mathbb{R}^n$. Dette underrommet kalles også for et *egenrom*.

6. La A være en kvadratisk matrise. Vis at A og A^T har de samme egenverdiene.

7. La A være en inverterbar matrise med egenverdier λ_i . Vis at A^{-1} har $1/\lambda_i$ som egenverdier. Forklar også at A og A^{-1} har de samme egenvektorene.

8. Anta at $\mathbf{v}$ er en egenvektor for både A og B . Vis at $\mathbf{v}$ er en egenvektor for $A + B$. Hva kan du si om de tilhørende egenverdiene?

KAPITTEL 8

Diagonalisering. Systemer av differensiallikninger

Anta at $n \times n$ -matrisen A er réell, og har egenvektorer $\mathbf{p}_1, \mathbf{p}_2, \dots, \mathbf{p}_k$ med tilhørende egenverdier $\lambda_1, \lambda_2, \dots, \lambda_k$. Siden $A\mathbf{p}_i = \lambda_i\mathbf{p}_i$ for $i = 1, \dots, k$ så har vi at

$$\begin{aligned} A(\mathbf{p}_1 \quad \mathbf{p}_2 \quad \cdots \quad \mathbf{p}_k) &= (A\mathbf{p}_1 \quad A\mathbf{p}_2 \quad \cdots \quad A\mathbf{p}_k) \\ &= (\lambda_1\mathbf{p}_1 \quad \lambda_2\mathbf{p}_2 \quad \cdots \quad \lambda_k\mathbf{p}_k) \\ &= (\mathbf{p}_1 \quad \mathbf{p}_2 \quad \cdots \quad \mathbf{p}_k) \begin{pmatrix} \lambda_1 & 0 & \cdots & 0 \\ 0 & \lambda_2 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & \lambda_k \end{pmatrix}. \end{aligned}$$

Vi kan også skrive dette som $AP = PD$, der $P = (\mathbf{p}_1 \quad \mathbf{p}_2 \quad \cdots \quad \mathbf{p}_k)$, og D er $k \times k$ diagonalmatrisen med $\lambda_1, \lambda_2, \dots, \lambda_k$ på diagonalen. Hvis A har n lineært uavhengige egenvektorer så blir matrisen P en inverterbar $n \times n$ -matrise. Ganger vi med P^{-1} til høyre på begge sider får vi at

$$A = PDP^{-1}.$$

Matriser som lar seg faktorisere slik (med D diagonal) kalles *diagonaliserbare*. Fra dette er det klart at matriser som har n lineært uavhengige egenvektorer er diagonaliserbare. Omvendt, hvis $A = PDP^{-1}$, så kan vi skrive $AP = PD$, som sier at $\{\mathbf{p}_i\}_{i=1}^n$ er lineært uavhengige egenvektorer for A , med $\{\lambda_i\}_{i=1}^n$ som tilhørende egenverdier. Vi har derfor vist følgende:

Proposisjon 8.1. En matrise A er diagonaliserbar hvis og bare hvis $\mathbb{C}^n$ har en basis bestående av egenvektorer for A .

Mens A antas reell så må vi i en diagonalisering tillate at P og D kan være komplekse - reelle matriser kan jo ha komplekse egenverdier, og komplekse egenverdier gir opphav til komplekse egenvektorer. Dette er grunnen til at vi over skrev $\mathbb{C}^n$, og ikke $\mathbb{R}^n$.

Hvorfor er det ønskelig at en matrise er diagonaliserbar? Med en diagonaliserbar matrise er det veldig lett å regne ut potenser, siden $A^k = PD^kP^{-1}$ (det er mindre arbeid å regne ut D^k enn A^k for store matriser A). Mer generelt er det lett å vise at

$$p(A) = P \begin{pmatrix} p(\lambda_1) & 0 & \cdots & 0 \\ 0 & p(\lambda_2) & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & p(\lambda_n) \end{pmatrix} P^{-1}$$

for ethvert polynom p .

Eksempel 8.2. I Eksempel 7.6 så vi at matrisen $A = \begin{pmatrix} 1 & -2 \\ 2 & -3 \end{pmatrix}$ ikke hadde en basis av egenvektorer. Den er derfor heller ikke diagonaliserbar.

Eksempel 8.3. I Eksempel 7.8 fant vi at matrisen $A = \begin{pmatrix} 2 & -8 \\ 2 & 2 \end{pmatrix}$ hadde egenverdier $\lambda_1 = 2 + 4i$ og $\lambda_2 = 2 - 4i$ med tilhørende egenvektorer $\mathbf{p}_1 = (2i, 1)$ og $\mathbf{p}_2 = (-2i, 1)$. Vi kan altså sette $P = \begin{pmatrix} 2i & -2i \\ 1 & 1 \end{pmatrix}$. Vi regner ut

$$P^{-1} = \frac{1}{2i+2i} \begin{pmatrix} 1 & 2i \\ -1 & 2i \end{pmatrix} = \frac{1}{4} \begin{pmatrix} -i & 2 \\ i & 2 \end{pmatrix}.$$

En diagonalisering av A blir dermed

$$A = PDP^{-1} = \frac{1}{4} \begin{pmatrix} 2i & -2i \\ 1 & 1 \end{pmatrix} \begin{pmatrix} 2+4i & 0 \\ 0 & 2-4i \end{pmatrix} \begin{pmatrix} -i & 2 \\ i & 2 \end{pmatrix}.$$

Hvis vi skriver $2 + 4i$ på polarform $re^{i\theta}$ så blir $(2 + 4i)^k = r^k e^{ik\theta}$ og $(2 - 4i)^k = r^k e^{-ik\theta}$. Dette gir

$$A^k = PD^kP^{-1} = \frac{r^k}{4} \begin{pmatrix} 2i & -2i \\ 1 & 1 \end{pmatrix} \begin{pmatrix} e^{ik\theta} & 0 \\ 0 & e^{-ik\theta} \end{pmatrix} \begin{pmatrix} -i & 2 \\ i & 2 \end{pmatrix}.$$

8.1 Diagonalisering og systemer av førsteordens likninger

Med et *lineært førsteordens system av differensiallikninger med konstante koeffisienter* mener vi et system på formen

$$\begin{aligned}x_1'(t) &= a_{11}x_1(t) + a_{12}x_2(t) + \cdots + a_{1n}x_n(t) \\x_2'(t) &= a_{21}x_1(t) + a_{22}x_2(t) + \cdots + a_{2n}x_n(t) \\&\vdots \\x_n'(t) &= a_{n1}x_1(t) + a_{n2}x_2(t) + \cdots + a_{nn}x_n(t),\end{aligned}\tag{8.1}$$

der $x_1(t), x_2(t), \dots, x_n(t)$ er ukjente funksjoner som vi antar er deriverbare. Skriver vi

$$\mathbf{x}(t) = (x_1(t), x_2(t), \dots, x_n(t)) \quad \mathbf{x}'(t) = (x_1'(t), x_2'(t), \dots, x_n'(t)),$$

og definerer matrisen A til å ha elementer a_{ij} , så kan dette skrives mer kompakt som

$$\mathbf{x}'(t) = A\mathbf{x}(t).$$

Diagonalisering er veldig nyttig når det kommer til å løse slike systemer:

Proposisjon 8.4. Anta A er diagonaliserbar, med $A = PDP^{-1}$. Da har vi at $\mathbf{x}(t)$ løser systemet $\mathbf{x}'(t) = A\mathbf{x}(t)$ hvis og bare hvis $\mathbf{y}(t) = P^{-1}\mathbf{x}(t)$ løser systemet $\mathbf{y}'(t) = D\mathbf{y}(t)$.

Bevis. Anta først at $\mathbf{x}(t)$ løser $\mathbf{x}'(t) = A\mathbf{x}(t)$. Med $\mathbf{y}(t) = P^{-1}\mathbf{x}(t)$ får vi da at

$$\mathbf{y}'(t) = (P^{-1}\mathbf{x})' = P^{-1}\mathbf{x}' = P^{-1}A\mathbf{x}(t) = P^{-1}AP\mathbf{y}(t) = D\mathbf{y}(t),$$

slik at $\mathbf{y}(t)$ løser $\mathbf{y}'(t) = D\mathbf{y}(t)$. Anta så at $\mathbf{y}(t)$ løser $\mathbf{y}'(t) = D\mathbf{y}(t)$. Vi får da

$$\mathbf{x}'(t) = (P\mathbf{y}(t))' = P\mathbf{y}'(t) = PD\mathbf{y}(t) = PDP^{-1}\mathbf{x}(t) = A\mathbf{x}(t),$$

slik at $\mathbf{x}(t)$ løser $\mathbf{x}'(t) = A\mathbf{x}(t)$. ■

Poenget er at systemet $\mathbf{y}'(t) = D\mathbf{y}(t)$ kan splittes opp i

$$y_1'(t) = \lambda_1 y_1(t) \quad y_2'(t) = \lambda_2 y_2(t) \quad \cdots \quad y_n'(t) = \lambda_n y_n(t)$$

slik at komponentfunksjonene kan finnes uavhengig av hverandre som $y_i(t) = C_i e^{\lambda_i t}$, der C_i er en konstant som kan være kompleks, og

$$\mathbf{y}(t) = (C_1 e^{\lambda_1 t}, C_2 e^{\lambda_2 t}, \dots, C_n e^{\lambda_n t}).$$

Det er interessant å tolke løsninger på forskjellige systemer i tilfellet der A er 2×2 , siden vi da kan plote x_1 mot x_2 i planet. Generell løsning når $n = 2$ er

$$\mathbf{x}(t) = P\mathbf{y}(t) = (\mathbf{p}_1 \quad \mathbf{p}_2) \begin{pmatrix} C_1 e^{\lambda_1 t} \\ C_2 e^{\lambda_2 t} \end{pmatrix} = C_1 e^{\lambda_1 t} \mathbf{p}_1 + C_2 e^{\lambda_2 t} \mathbf{p}_2,$$

La oss først se på tilfellet der A har reelle egenverdier. Husk at det da ikke er sikkert at A er diagonaliserbar når A har like egenverdier. Vi har følgende resultat.

Proposisjon 8.5. *Anta A har reelle egenverdier og at $\mathbf{x}(t_0)$ er kjent, og reell. Da finnes det unike reelle konstanter C_1 og C_2 slik at*

$$\mathbf{x}(t) = C_1 e^{\lambda_1 t} \mathbf{p}_1 + C_2 e^{\lambda_2 t} \mathbf{p}_2. \quad (8.2)$$

Omvendt er det klart at (8.2) er reell når C_1 og C_2 er reelle.

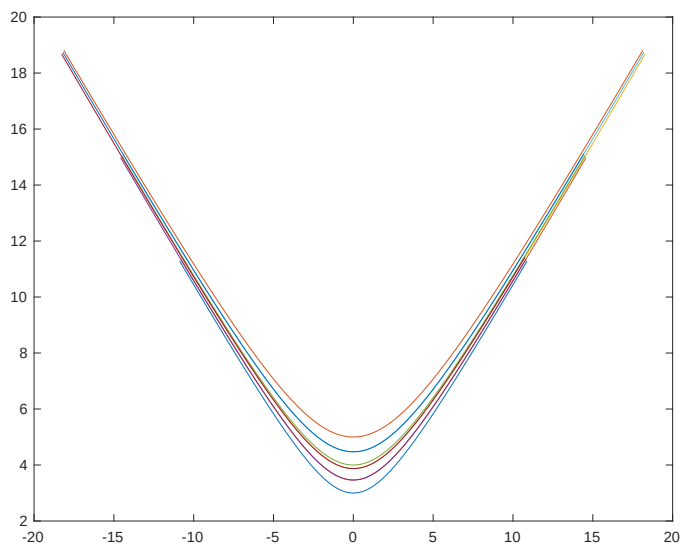
Bevis. Systemet $\mathbf{x}(t_0) = (\mathbf{p}_1 \quad \mathbf{p}_2) \begin{pmatrix} C_1 e^{\lambda_1 t_0} \\ C_2 e^{\lambda_2 t_0} \end{pmatrix}$ er i dette tilfellet reelt. Siden P er invertierbar følger det at også løsningen $\begin{pmatrix} C_1 e^{\lambda_1 t_0} \\ C_2 e^{\lambda_2 t_0} \end{pmatrix}$ er unik og reell. Men da er også C_1 og C_2 unike og reelle. ■

Eksempel 8.6 (Eksempel med reelle egenverdier). Sett $A = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}$. A har egenverdier $\lambda_1 = -1$ og $\lambda_2 = 1$, med tilhørende egenvektorer $\mathbf{p}_1 = (-1, 1)$ og $\mathbf{p}_2 = (1, 1)$. Den generelle løsningen (8.2) blir dermed

$$C_1 e^{-t} \begin{pmatrix} -1 \\ 1 \end{pmatrix} + C_2 e^t \begin{pmatrix} 1 \\ 1 \end{pmatrix} = \begin{pmatrix} -C_1 e^{-t} + C_2 e^t \\ C_1 e^{-t} + C_2 e^t \end{pmatrix},$$

og det er klart at disse har $f(x) = x$ og $f(x) = -x$ som asymptoter når $t \rightarrow \infty$ og $t \rightarrow -\infty$. Vi kan plote flere løsninger $C_1 e^{\lambda_1 t} \mathbf{p}_1 + C_2 e^{\lambda_2 t} \mathbf{p}_2$ i samme system ved å variere C_1 og C_2 . I Figur 8.1 viser løsningene vi får ved å velge $C_i \in \{1.5, 2, 2.5\}$ (ni løsninger).

La oss så se på tilfellet der A har komplekse egenverdier. Vi vet da at A er diagonaliserbar, og at egenverdier og egenvektorer kommer i konjugerte par.



Figur 8.1. Plott av ni løsninger for systemet med matrise $A = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}$.

Proposisjon 8.7. Anta A har komplekse egenverdier λ og $\bar{\lambda}$ (med tilhørende egenvektorer $\mathbf{p}$ og $\bar{\mathbf{p}}$). Anta også at $\mathbf{x}(t_0)$ er kjent, og reell. Da finnes det en konstant C (som kan være kompleks) slik at

$$\mathbf{x}(t) = 2\Re(Ce^{\lambda t}\mathbf{p}). \quad (8.3)$$

Omvendt er det klart at (8.3) er reell for alle verdier av C .

Bevis. Vi har at

$$\mathbf{x}(t) = P\mathbf{y}(t) = (\mathbf{p} \quad \bar{\mathbf{p}}) \begin{pmatrix} C_1 e^{\lambda t} \\ C_2 e^{\bar{\lambda} t} \end{pmatrix}$$

for konstanter C_1 og C_2 . Setter vi $t = t_0$ får vi

$$\mathbf{x}(t_0) = (\mathbf{p} \quad \bar{\mathbf{p}}) \begin{pmatrix} C_1 e^{\lambda t_0} \\ C_2 e^{\bar{\lambda} t_0} \end{pmatrix} \quad (8.4)$$

$$\mathbf{x}(t_0) = \overline{\mathbf{x}(t_0)} = (\bar{\mathbf{p}} \quad \mathbf{p}) \begin{pmatrix} \overline{C_1} e^{\bar{\lambda} t_0} \\ \overline{C_2} e^{\lambda t_0} \end{pmatrix} = (\mathbf{p} \quad \bar{\mathbf{p}}) \begin{pmatrix} \overline{C_2} e^{\lambda t_0} \\ \overline{C_1} e^{\bar{\lambda} t_0} \end{pmatrix}, \quad (8.5)$$

der vi byttet om rekkefølgen i de to første komponentene. Siden P er inverterbar får vi ved å sammenligne (8.4) og (8.5) at $C_2 = \overline{C_1}$, og løsningen blir

$$\mathbf{x}(t) = \begin{pmatrix} \mathbf{p} & \overline{\mathbf{p}} \end{pmatrix} \begin{pmatrix} C_1 e^{\lambda t} \\ \overline{C_1} e^{\bar{\lambda} t} \end{pmatrix} = C_1 e^{\lambda t} \mathbf{p} + \overline{C_1} e^{\bar{\lambda} t} \overline{\mathbf{p}} = 2\Re(C_1 e^{\lambda t} \mathbf{p}),$$

slik at vi kan sette $C = C_1$. ■

Uttrykket $2\Re(C_1 e^{\lambda t} \mathbf{p})$ involverer regning med komplekse tall, selv om svaret blir reelt. Det er imidlertid lett å skrive det om til ikke å bruke komplekse tall. La $\mathbf{c}$ og $\mathbf{d}$ være realdel og imaginærdel til den komplekse vektoren $C\mathbf{p}$, og la $\lambda = a + ib$. Da er $C\mathbf{p} = \mathbf{c} + i\mathbf{d}$ og $e^{\lambda t} = e^{at}(\cos(bt) + i\sin(bt))$, og vi får

$$\Re(Ce^{\lambda t} \mathbf{p}) = \Re(e^{at}(\mathbf{c} + i\mathbf{d})(\cos(bt) + i\sin(bt))) = e^{at}(\cos(bt)\mathbf{c} - \sin(bt)\mathbf{d}),$$

slik at

$$\mathbf{x}(t) = 2e^{at}(\cos(bt)\mathbf{c} - \sin(bt)\mathbf{d}). \quad (8.6)$$

Fra denne formen er følgende klart:

1. For $a > 0$: $x(t) \rightarrow \infty$ når $t \rightarrow \infty$ (spiralbevegelse vekk fra origo),
2. For $a < 0$: $x(t) \rightarrow 0$ når $t \rightarrow \infty$ (spiralbevegelse inn mot origo),
3. For $a = 0$: $x(t)$ er periodisk.

Siden $\cos(bt)$ og $\sin(bt)$ er periodiske med periode $2\pi/b$, så skal et plott over $[0, 2\pi/b]$ vise en arm av spiralbevegelsen i 1. og 2., eller en hel periode i 3..

Følgende kode antar at A er reell og har komplekse egenverdier, og at vi har en initialbetingelse $x(t_0) = x_0$. Koden regner ut egenverdiene og egenvektorene til A , løser systemet

$$\begin{pmatrix} \mathbf{p} & \overline{\mathbf{p}} \end{pmatrix} \begin{pmatrix} C e^{\lambda t_0} \\ \overline{C} e^{\bar{\lambda} t_0} \end{pmatrix} = \mathbf{x}_0, \quad (8.7)$$

finner C , regner ut $C\mathbf{p} = \mathbf{c} + i\mathbf{d}$, og deretter løsningen (8.6), som til slutt plottes:

```
D,P=np.linalg.eig(A)
C = np.linalg.solve(P, x0)
C = C[0]/np.exp(D[0]*t0)
c = np.real(C*P[:, [0]])
d = np.imag(C*P[:, [0]])
a = np.real(D[0])
b = np.imag(D[0])
```

```
t=np.linspace(t0,t0+4*np.pi/b,100).reshape((1,-1))
res = ( c@(np.cos(b*t)*np.exp(a*t)) - d@(np.sin(b*t)*np.exp(a*t)) ) * 2
plt.plot(res[0,:],res[1,:])
```

Siden vi plottet over $[t_0, t_0 + 4\pi/b]$, så skal plottet vise to armer i eventuelle spiralbevegelser.

Eksempel 8.8 (Eksempler med komplekse egenverdier). La oss se på de tre matrisene

$$A_1 = \begin{pmatrix} 0.1 & 1 \\ -1 & 0.1 \end{pmatrix} \quad A_2 = \begin{pmatrix} -0.1 & 1 \\ -1 & -0.1 \end{pmatrix} \quad A_3 = \begin{pmatrix} 0 & 1 \\ -1 & 0 \end{pmatrix}$$

Disse har egenverdier $0.1 \pm i$, $-0.1 \pm i$, og $\pm i$, respektive. Vi har altså at $a > 0$, $a < 0$, og $a = 0$, respektive, slik at løsningene på systemet over vil demonstrere de tre mulighetene nevnt over. Spesielt vil den første løsningen gi en spiralbevegelse vekk fra origo, den andre en spiralbevegelse inn mot origo. Løsningen er plottet i Figur 8.2. Vi har brukt initialbetingelsen $\mathbf{x}(2) = (2, 3)$, slik at alle tre løsningene starter i punktet $(2, 3)$.

Man kan også plote t mot $x_i(t)$. Dette gjør at man kan visualisere alle komponentene i samme plott, også når $n > 2$. I Figur 8.3 plotter vi kun den første komponenten $x_1(t)$, over det samme tidsintervallet og med samme initialbetingelse som over. Siden $c_1 \cos(bt) - d_1 \sin(bt)$ har amplitude $\sqrt{c_1^2 + d_1^2}$ (Oppgave 2) så har vi begrensninger

$$-2\sqrt{c_1^2 + d_1^2}e^{at} \leq x_1(t) \leq 2\sqrt{c_1^2 + d_1^2}e^{at} \quad (8.8)$$

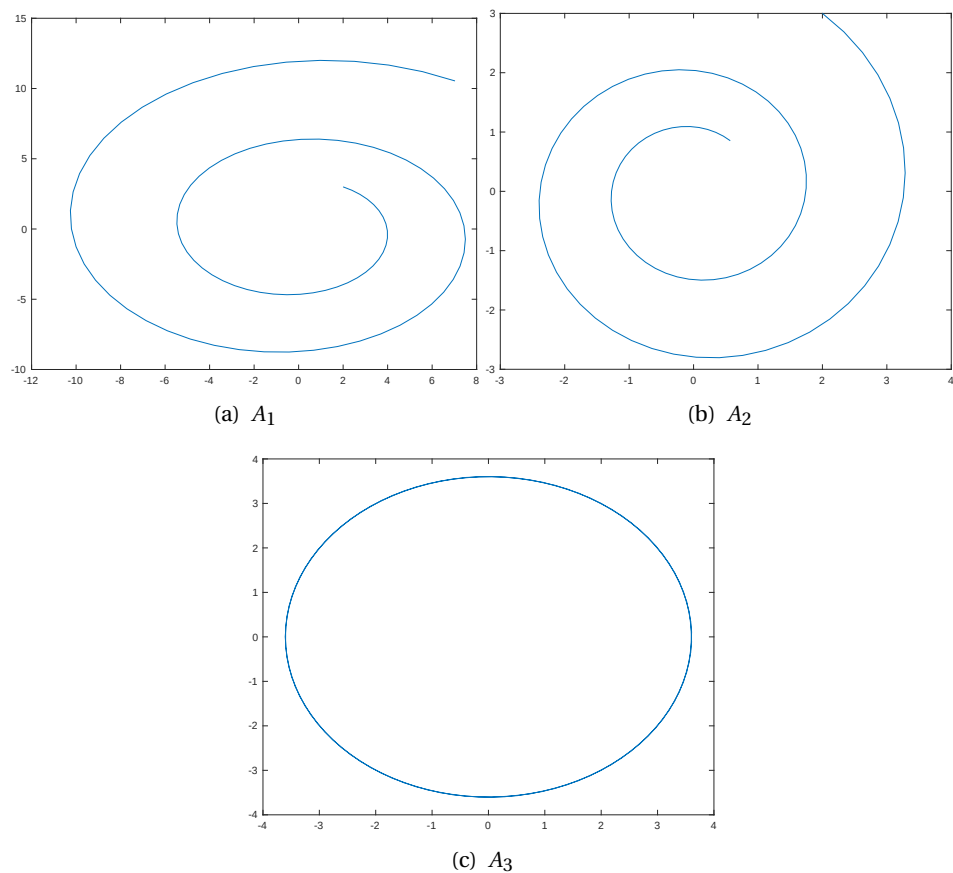
Disse er også tatt med i plottet.

Det viser seg at vi kan sette opp løsningen for systemer med $n > 2$, basert på det vi gjorde over. I det generelle tilfellet har vi en blanding av reelle og komplekse egenverdier og egenvektorer:

$$\lambda_1, \overline{\lambda_1}, \dots, \lambda_m, \overline{\lambda_m}, \lambda_{2m+1}, \dots, \lambda_n$$

$$\mathbf{p}_1, \overline{\mathbf{p}_1}, \dots, \mathbf{p}_m, \overline{\mathbf{p}_m}, \mathbf{p}_{2m+1}, \dots, \mathbf{p}_n,$$

der vi listet opp de reelle til slutt. Oppgave 3 viser hvordan resonnementet blir for det tilfellet der vi har to komplekse og en reell egenverdi.



Figur 8.2. Løsningene på systemene i Eksempel 8.8.

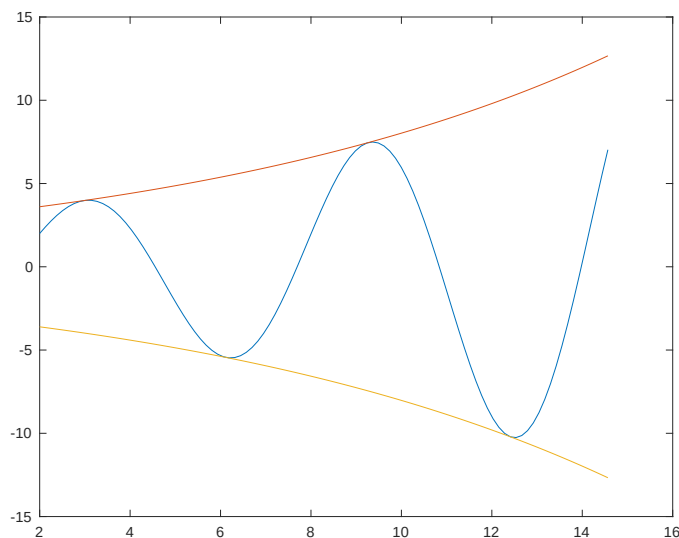
8.2 Høyereordens lineære differensiallikninger

En type likning beslektet med systemene (8.1) er *høyereordens lineære differensiallikninger med konstante koeffisienter*. Disse er på formen

$$y^{(n)}(t) + a_1 y^{(n-1)}(t) + a_2 y^{(n-2)}(t) + \cdots + a_n y(t) = 0, \quad (8.9)$$

der $y(t)$ er den ukjente funksjonen. Vi skal begrense oss til andreordens likninger, og vil skrive disse på formen

$$y'' + py' + qy = 0$$



Figur 8.3. Plott av $x_1(t)$ for systemet med matrise A_1 . Øvre og nedre grense fra (8.8) er også vist.

der p og q er konstanter. Ved å sette $x_0 = y$, $x_1 = y'$ kan vi skrive dette som det førsteordens lineære systemet

$$\begin{aligned} x_0' &= x_1 \\ x_1' &= -qx_0 - px_1 \end{aligned}$$

som har koeffisientmatrise $A = \begin{pmatrix} 0 & 1 \\ -q & -p \end{pmatrix}$. Vi har at

$$\det(A - \lambda I) = \lambda(\lambda + p) + q = \lambda^2 + p\lambda + q.$$

La oss anta at egenverdiene λ_1 og λ_2 er forskjellige (dette er det samme som at $p^2 \neq 4q$). Da er A diagonaliserbar, og vi har at

$$A - \lambda I_n = \begin{pmatrix} -\lambda & 1 \\ -q & -\lambda - p \end{pmatrix} \sim \begin{pmatrix} -\lambda & 1 \\ 0 & 0 \end{pmatrix},$$

slik at $(1, \lambda_1)$ og $(1, \lambda_2)$ er to spesielle egenvektorer. Vi kan dermed sette

$$P = \begin{pmatrix} 1 & 1 \\ \lambda_1 & \lambda_2 \end{pmatrix}.$$

La oss først anta reelle egenverdier. Vi har vist at den generelle løsningen er

$$\begin{pmatrix} y(t) \\ y'(t) \end{pmatrix} = \begin{pmatrix} 1 & 1 \\ \lambda_1 & \lambda_2 \end{pmatrix} \begin{pmatrix} C_1 e^{\lambda_1 t} \\ C_2 e^{\lambda_2 t} \end{pmatrix}, \quad (8.10)$$

der man kan finne C_1 og C_2 ved hjelp av initialverdier $y(t_0)$, $y'(t_0)$.

Eksempel 8.9 (Eksempel med distinkte reelle egenverdier). La oss se på likningen

$$y'' - 5y' + 6y = 0,$$

med initialbetingelser $y(0) = 1$, $y'(0) = 0$. Egenverdiene til A finner vi ved å løse den karakteristiske likningen $\lambda^2 - 5\lambda + 6 = 0$, som gir $\lambda_1 = 2$, $\lambda_2 = 3$. (8.10) innsatt $t = 0$ gir

$$\begin{pmatrix} 1 \\ 0 \end{pmatrix} = \begin{pmatrix} 1 & 1 \\ 2 & 3 \end{pmatrix} \begin{pmatrix} C_1 \\ C_2 \end{pmatrix},$$

som har løsning $C_1 = 3$, $C_2 = -2$. Løsningen blir derfor

$$y(t) = C_1 e^{\lambda_1 t} + C_2 e^{\lambda_2 t} = 3e^{2t} - 2e^{3t}.$$

Tilfellet med komplekse egenverdier er litt mer komplisert. Vi skal vise følgende

Proposisjon 8.10.

$$y(t) = 2e^{at} (\cos(bt)c_1 - \sin(bt)d_1), \quad (8.11)$$

der c_1 og d_1 er vilkårlig valgte reelle tall (husk at $\lambda = a + ib$).

Bevis. (8.6) sa at den generelle løsningen er

$$\begin{pmatrix} y(t) \\ y'(t) \end{pmatrix} = 2e^{at} (\cos(bt)\mathbf{c} - \sin(bt)\mathbf{d}), \quad (8.12)$$

der $\mathbf{c} = \Re(C\mathbf{p})$, $\mathbf{d} = \Im(C\mathbf{p})$, og der C er et vilkårlig valgt komplekst tall.

La nå c_1 , d_1 være gitte tall, og definér $C = c_1 + id_1$. Da er

$$C\mathbf{p} = \begin{pmatrix} c_1 + id_1 \\ \cdot \end{pmatrix} \quad \mathbf{c} = \Re(C\mathbf{p}) = \begin{pmatrix} c_1 \\ \cdot \end{pmatrix} \quad \mathbf{d} = \Im(C\mathbf{p}) = \begin{pmatrix} d_1 \\ \cdot \end{pmatrix},$$

der $\cdot$ indikerer et konkret tall. Evaluerer vi første rad i (8.12) med dette får vi at (8.11) er en løsning for alle valg av c_1 og d_1 , og på grunn av (8.12) så må alle løsninger være på denne formen. ■

Eksempel 8.11 (Eksempel med komplekse egenverdier). La oss se på likningen

$$y'' + 4y' + 5y = 0.$$

Den karakteristiske likningen blir nå $\lambda^2 + 4\lambda + 5 = 0$, som har løsning $\frac{-4 \pm \sqrt{16-20}}{2} = -2 \pm i$. Den generelle løsningen blir dermed

$$y(t) = 2e^{-2t} (c_1 \cos t - d_1 \sin t).$$

Vi har så langt unngått tilfellet der $\lambda_1 = \lambda_2$. I dette tilfellet har vi at $\lambda_1 = \lambda_2 = -p/2$, og at $p^2 = 4q$. Vi får da

$$A - \lambda I = \begin{pmatrix} -\lambda & 1 \\ -q & -\lambda - p \end{pmatrix} = \begin{pmatrix} p/2 & 1 \\ -p^2/4 & -p/2 \end{pmatrix} \sim \begin{pmatrix} p/2 & 1 \\ 0 & 0 \end{pmatrix}.$$

Det følger at egenrommet har dimensjon 1 slik at A ikke er diagonaliserbar. Vi kan dermed ikke bruke det vi har gjort for å finne løsningene. Det er imidlertid en oppgave å vise ved direkte innsetting at $y(t) = e^{\lambda t}$ er en løsning, og også at $y(t) = te^{\lambda t}$ er en løsning. Det følger nå at alle funksjoner på formen $y(t) = Ce^{\lambda t} + Dte^{\lambda t}$ også er løsninger, og man kan vise at det finnes en unik slik løsning som tilfredsstiller initialbetingelser på formen $y(t_0)$ og $y'(t_0)$. Vi skal ikke gå inn på hvordan vi gjettet løsningen $te^{\lambda t}$, og vi skal heller ikke bevise at alle løsninger kan uttrykkes ved hjelp av funksjonene $e^{\lambda t}$ og $te^{\lambda t}$.

Eksempel 8.12 (Eksempel med dobbelrot). La oss se på likningen $y'' + 6y' + 9y = 0$ med initialbetingelsene $y(0) = 4$, $y'(0) = -13$. Den karakteristiske likningen blir $\lambda^2 + 6\lambda + 9 = 0$, som har løsning $\lambda_1 = \lambda_2 = -3$. Den generelle løsningen blir dermed $y(t) = Ce^{-3t} + Dte^{-3t}$, og vi har også at $y'(t) = -3Ce^{-3t} + D(1 - 3t)e^{-3t}$. Initialbetingelsene gir da likningene

$$\begin{aligned} C &= 4 \\ -3C + D &= -13, \end{aligned}$$

som har løsning $C = 4$, $D = -1$. Løsningen blir dermed $y(t) = 4e^{-3t} - te^{-3t}$.

Oppgaver for Kapittel 8

1 (Eksamen 2024). La A være matrisen $A = \begin{pmatrix} 0.7 & 0.8 \\ 0.3 & 0.2 \end{pmatrix}$.

(a). Finn egenverdiene og egenvektorene til A .

(b). Skriv ned en diagonalisering av matrisen A . Har A en ortonormal basis av egenvektorer?

2. La b , c , og d være reelle tall. Vis at $|c \cos(bt) - d \sin(bt)| \leq \sqrt{c^2 + d^2}$ for alle t . For hvilke t har du likhet her?

3. Vi er gitt systemet

$$\begin{aligned} x_1'(t) &= x_1(t) + x_2(t) \\ x_2'(t) &= -x_1(t) + x_2(t) \\ x_3'(t) &= x_3(t) \end{aligned}$$

(a). Finn den generelle løsningen av systemet.

(b). Finn løsningen av systemet gitt initialbetingelsen $\mathbf{x}(0) = (1, 1, 1)$. Hva skjer med løsningen når $t \rightarrow -\infty$ og når $t \rightarrow \infty$?

4. Finn løsningen av

(a). $y'' - 5y' + 4y = 0$ med initialbetingelsene $y(0) = 2$, $y'(0) = 5$.

(b). $y'' + 2y' + 2y = 0$.

(c). $y'' - 4y' + 4y = 0$ med initialbetingelsene $y(0) = 3$, $y'(0) = 8$.

5. Vi skal se på likningen $y'' + py' + qy$, og antar at den karakteristiske likningen til denne har en dobbelrot λ .

(a). Vis at $y(t) = e^{\lambda t}$ løser likningen.

(b). Vis at $y(t) = te^{\lambda t}$ løser likningen (du vil her få bruk for at $\lambda_1 = \lambda_2 = -p/2$, og at $p^2 = 4q$ når vi har en dobbelrot).

(c). Vis at det finnes et unikt valg av C og D slik at $y(t) = Ce^{\lambda t} + Dte^{\lambda t}$ tilfredsstiller initialbetingelser $y(t_0)$ og $y'(t_0)$.

6. Anta at A er diagonaliserbar, slik at vi kan skrive $A = PDP^{-1}$. Siden

$$PD = (\lambda_1 \mathbf{p}_1 \quad \lambda_2 \mathbf{p}_2 \quad \cdots \quad \lambda_n \mathbf{p}_n)$$

så gir den ytre formen for matrisemultiplikasjon at

$$PDP^{-1} = \sum_{k=1}^n \lambda_k P_{:,k} (P^{-1})_{k,:}. \quad (8.13)$$

Dette splitter opp A i en sum av enklere matriser $Q_k = P_{:,k} (P^{-1})_{k,:}$. Vi kan altså skrive $A = \sum_{k=1}^n \lambda_k Q_k$. I denne oppgaven skal vi se nærmere på matrisene Q_k .

(a). Vis at $Q_k^2 = Q_k$. Matriser som oppfyller dette kalles også *idempotente*.

(b). Vis at $Q_r Q_s = 0$ når $r \neq s$.

(c). Vis at $p(A)$ kan skrives som en lineær kombinasjon av $\{Q_k\}_{k=1}^n$, når p er et polynom.

(d). Vis at Q_k kan skrives som et polynom i A .

KAPITTEL 9

Ortogonal diagonaliserbare matriser. Kondisjonstall

Når vi skal løse systemet $A\mathbf{x} = \mathbf{b}$ på en datamaskin, så vil matrisen A og høyresiden $\mathbf{b}$ typisk være gjenstand for målingsfeil eller avrundingsfeil. Datamaskinen vil derfor egentlig løse et problem på formen

$$(A + E)\mathbf{y} = \mathbf{b} + \mathbf{e},$$

der $E/\mathbf{e}$ er en matrise/vektor med små elementer, og som representerer en feil fra den faktiske koeffisientmatrisen/høyresiden. I det kommende skal vi for enkelhets skyld anta at vi ikke har avrundingsfeil i koeffisientmatrisen, slik at $E = 0$. Vi skal stille oss følgende spørsmål: Hvordan forplanter feilen $\mathbf{e}$ i høyresiden seg til løsningen av systemet? Er det noen tilfeller der selv en liten feil $\mathbf{e}$ vil propagerer til en stor feil mellom løsningene $\mathbf{x}$ og $\mathbf{y}$? En slik oppførsel er helt klart ikke ønskelig, siden dette vil gjøre oss veldig sårbare for små feil i systemet.

For å illustrere at en slik oppførsel faktisk er mulig, sett

$$A = \frac{1}{2} \begin{pmatrix} 1 + \epsilon & 1 - \epsilon \\ 1 - \epsilon & 1 + \epsilon \end{pmatrix},$$

der ϵ er et tall vi skal eksperimentere med. La oss se på systemet $A\mathbf{x} = \mathbf{b}$, der $\mathbf{b} = \begin{pmatrix} 1 \\ 1 \end{pmatrix}$. Det er lett å vise at den eksakte løsningen er $\mathbf{x} = \begin{pmatrix} 1 \\ 1 \end{pmatrix}$ for alle ϵ . Hvis vi har en liten feil $\mathbf{e}$, så ville vi da forvente at løsningen av $A\mathbf{y} = \mathbf{b} + \mathbf{e}$ er nær $\mathbf{x} = (1, 1)$, uansett hva ϵ måtte være. La oss sjekke dette med $\mathbf{e} = 10^{-3} \begin{pmatrix} 1 \\ -1 \end{pmatrix}$. I Oppgave 4 skal

du sjekke at $A\mathbf{y} = \mathbf{b} + \mathbf{e}$ har løsning

$$\mathbf{y} = \begin{pmatrix} 1 \\ 1 \end{pmatrix} + 10^{-3}\epsilon^{-1} \begin{pmatrix} 1 \\ -1 \end{pmatrix}.$$

Her skjer det noe ganske uventet: Når $\epsilon \rightarrow 0$ får vi veldig store løsninger $\mathbf{y}$, som blir lenger og lenger unna $\mathbf{x}$. Hva er det som går galt her? Den forklaringen vi skal gi er at A får et høyt *kondisjonstall* når $\epsilon \rightarrow 0$. Sagt på en litt annen måte, når $\epsilon \rightarrow 0$ vil matrisen A nærme seg den ikke-inverterbare matrisen $\frac{1}{2} \begin{pmatrix} 1 & 1 \\ 1 & 1 \end{pmatrix}$, og dette er problematisk siden et system med en ikke-inverterbar matrise ikke har unik løsning.

Oppførselen over er lettest å forklare hvis vi begrenser oss til symmetriske matriser. Vi skal i neste seksjon forklare at slike matriser er det vi kaller ortogonalt diagonaliserbare, og det er denne egenskapen som vil hjelpe oss videre.

9.1 Ortogonalt diagonaliserbare matriser

I faktoriseringen av en diagonaliserbar matrise inngår P^{-1} . Hvis P også er ortogonal (det vil si at P er reell og $P^{-1} = P^T$) så har vi at

$$A = PDP^T.$$

Reelle matriser som kan skrives slik kalles *ortogonalt diagonaliserbare*. Slike matriser må være symmetriske siden

$$A^T = (PDP^T)^T = PD^T P^T = PDP^T = A.$$

Siden $D = P^T A P$ og A og P er reelle, så er da også D reell.

Når $P^{-1} = P^T$ gir den ytre formen for matrisemultiplikasjon oss at

$$PDP^T = \sum_{k=1}^n \lambda_k P_{:k} (P_{:k})^T \quad (9.1)$$

Siden søylene i P har lengde 1, så sier dette at A kan skrives som en (vektet) sum av projeksjonsmatriser (se Eksempel 2.15).

Spektralteoremet sier at, den andre veien, alle reelle symmetriske matriser er ortogonalt diagonaliserbare. Å bevise dette er litt mer omstendelig. Først viser vi følgende.

Proposisjon 9.1. *En reell, symmetrisk $n \times n$ -matrise A har kun reelle egenverdier, og alle egenvektorer kan skrives på reell form.*

Bevis. La $\mathbf{x}, \mathbf{y} \in \mathbb{R}^n$. Med definisjonen av det komplekse skalarproduktet får vi

$$\begin{aligned} (\mathbf{Ax}) \cdot \mathbf{y} &= (\mathbf{Ax})^T \bar{\mathbf{y}} = \mathbf{x}^T A^T \bar{\mathbf{y}} \\ \mathbf{x} \cdot (\mathbf{Ay}) &= \mathbf{x}^T \overline{(\mathbf{Ay})} = \mathbf{x}^T \overline{A\mathbf{y}} = \mathbf{x}^T A \bar{\mathbf{y}}. \end{aligned}$$

Siden $A = A^T$ så har vi her likhet for alle $\mathbf{x}, \mathbf{y}$, d.v.s.

$$(\mathbf{Ax}) \cdot \mathbf{y} = \mathbf{x} \cdot (\mathbf{Ay}). \quad (9.2)$$

Hvis her $\mathbf{x} = \mathbf{y}$ og $\mathbf{Ax} = \lambda \mathbf{x}$ med $\mathbf{x} \neq \mathbf{0}$, så gir innsetting i (9.2) at

$$\lambda |\mathbf{x}|^2 = \bar{\lambda} |\mathbf{x}|^2.$$

Disse to er like hvis og bare hvis $\lambda = \bar{\lambda}$, det vil si at λ er reell. Radredusering av $A - \lambda I$ gir oss nå at alle egenvektorer kan skrives på reell form. ■

Deretter trenger vi følgende resultat, som er det vanskelige punktet.

Teorem 9.2. En reell, symmetrisk $n \times n$ -matrise kan skrives på formen

$$A = PRP^T,$$

med P ortogonal, og R øvretriangulær.

Faktoriseringen $A = PRP^T$ med P ortogonal og R øvretriangulær kalles også for en *Schur-faktorisering*. Teoremet holder mer generelt, men vi trenger bare resultatet for symmetriske matriser.

Bevis. For $n = 1$ er dette opplagt (velg $P = 1$). Vi antar nå vi har vist at alle reelle $(n-1) \times (n-1)$ -matriser har en Schur-faktorisering. La $\mathbf{x}_1$ være en egenvektor for A med egenverdi λ_1 . La $\mathbf{x}_1$ ha lengde 1. Vi kan anta at både $\mathbf{x}_1$ og λ_1 er reelle siden A er symmetrisk. Utvid $\mathbf{x}_1$ til en ortonormal basis $\{\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_n\}$ for $\mathbb{R}^n$ (se Proposisjon 5.9), og sett

$$P_1 = (\mathbf{x}_1 \quad \mathbf{x}_2 \quad \cdots \quad \mathbf{x}_n).$$

P_1 er ortogonal. Vi har at

$$P_1^T A P_1 \mathbf{e}_1 = P_1^T A \mathbf{x}_1 = \lambda_1 P_1^T \mathbf{x}_1 = \lambda_1 \mathbf{e}_1,$$

slik at $P_1^T A P_1 = \left(\begin{array}{c|c} \lambda_1 & \mathbf{a} \\ \mathbf{0} & M \end{array} \right)$, der $\mathbf{a}$ er en radvektor med $n-1$ komponenter, og M er en $(n-1) \times (n-1)$ matrise. Vi har da at

$$A = P_1 \left(\begin{array}{c|c} \lambda_1 & \mathbf{a} \\ \mathbf{0} & M \end{array} \right) P_1^T.$$

Matrisen M er symmetrisk siden A er det, og ved induksjonsantagelsen har den derfor en Schur-faktorisering $M = P_0 R_0 P_0^T$, der P_0 er ortogonal og R_0 øvretriangular. Setter vi inn dette får vi

$$\begin{aligned} A &= P_1 \left(\begin{array}{c|c} \lambda_1 & \mathbf{a} \\ \hline \mathbf{0} & P_0 R_0 P_0^T \end{array} \right) P_1^T \\ &= P_1 \left(\begin{array}{c|c} 1 & \mathbf{0} \\ \hline \mathbf{0} & P_0 \end{array} \right) \left(\begin{array}{c|c} \lambda_1 & \mathbf{a} P_0 \\ \hline \mathbf{0} & R_0 \end{array} \right) \left(\begin{array}{c|c} 1 & \mathbf{0} \\ \hline \mathbf{0} & P_0^T \end{array} \right) P_1^T \\ &= P_1 \left(\begin{array}{c|c} 1 & \mathbf{0} \\ \hline \mathbf{0} & P_0 \end{array} \right) \left(\begin{array}{c|c} \lambda_1 & \mathbf{a} P_0 \\ \hline \mathbf{0} & R_0 \end{array} \right) \left(P_1 \left(\begin{array}{c|c} 1 & \mathbf{0} \\ \hline \mathbf{0} & P_0 \end{array} \right) \right)^T. \end{aligned}$$

Siden $P_1 \left(\begin{array}{c|c} 1 & \mathbf{0} \\ \hline \mathbf{0} & P_0 \end{array} \right)$ er ortogonal (dette er en oppgave) og $\left(\begin{array}{c|c} \lambda_1 & \mathbf{a} P_0 \\ \hline \mathbf{0} & R_0 \end{array} \right)$ er øvretriangular, så er dette en Schur-faktorisering av A . ■

Hvis $A = PRP^T$ er symmetrisk så har vi at $PRP^T = PR^T P^T$, slik at $R = R^T$. Men en øvretriangular matrise som er symmetrisk må være diagonal, slik at Schur-faktoriseringen faktisk er en ortogonal diagonalisering. Reelle, symmetriske matriser er altså ortogonalt diagonaliserbare, og siden vi startet med å si at ortogonalt diagonaliserbare matriser er symmetriske, så har vi vist følgende.

Teorem 9.3 (Spektralteoremet). *En reell $n \times n$ -matrise er ortogonalt diagonaliserbar hvis og bare hvis den er symmetrisk.*

La oss oppsummere et par andre ting også. Vi vet at, hvis $A = PDP^T$, så er søyle i i P en egenvektor for A med egenverdi λ_i . Siden søylene i P er ortonormale, så følger at egenrommene er ortogonale, og dimensjonen til egenrommet er lik multiplisiteten til egenverdien.

Teorem 9.4. *En reell symmetrisk $n \times n$ -matrise*

1. *har n reelle egenverdier,*
2. *har innbyrdes ortogonale egenrom,*
3. *har dimensjonen til egenrommet for λ lik multiplisiteten til λ som rot i den karakteristiske likningen.*

Vi skal få bruk for følgende resultat

Lemma 9.5. Anta at A er reell, symmetrisk, og inverterbar, og la

$$|\lambda_1| \geq |\lambda_2| \geq \dots \geq |\lambda_n| > 0$$

være egenverdiene til A listet opp etter avtagende absoluttverdi. Da har vi at

$$|\lambda_n||\mathbf{x}| \leq |A\mathbf{x}| \leq |\lambda_1||\mathbf{x}| \quad (9.3)$$

for alle $\mathbf{x}$.

Bevis. Siden symmetriske matriser er ortogonalt diagonaliserbare kan vi skrive $A = PDP^T$, der P er ortogonal. Vi har at

$$|A\mathbf{x}|^2 = |PDP^T\mathbf{x}|^2 = \mathbf{x}^T PDP^T PDP^T \mathbf{x} = (P^T\mathbf{x})^T D^2 P^T \mathbf{x} = \mathbf{u}^T D^2 \mathbf{u},$$

der vi satt $\mathbf{u} = P^T \mathbf{x}$. Vi får så

$$\begin{aligned} \mathbf{u}^T D^2 \mathbf{u} &= \sum_{i=1}^n \lambda_i^2 u_i^2 \leq \lambda_1^2 \sum_{i=1}^n u_i^2 = \lambda_1^2 |\mathbf{u}|^2 = \lambda_1^2 |\mathbf{x}|^2 \\ \mathbf{u}^T D^2 \mathbf{u} &= \sum_{i=1}^n \lambda_i^2 u_i^2 \geq \lambda_n^2 \sum_{i=1}^n u_i^2 = \lambda_n^2 |\mathbf{u}|^2 = \lambda_n^2 |\mathbf{x}|^2, \end{aligned}$$

der vi brukte at $|\mathbf{u}|^2 = \mathbf{x}^T P P^T \mathbf{x} = \mathbf{x}^T \mathbf{x} = |\mathbf{x}|^2$. Ved å ta kvadratrøtter følger at $|\lambda_n||\mathbf{x}| \leq |A\mathbf{x}| \leq |\lambda_1||\mathbf{x}|$. ■

9.2 Relativ og absolutt feil

Diskusjonen vi startet dette kapitlet med hadde å gjøre med feilen vi får ved å løse systemer. Derfor er det viktig å være presis med rundt begrepet feil. Når vi snakker om feil mellom to størrelser er det naturlig å bruke begrepet relativ feil:

Definisjon 9.6. Anta a og b er tall. Vi kaller $\left| \frac{b-a}{a} \right|$ for den relative feilen mellom a og b . Hvis $\mathbf{a}$ og $\mathbf{b}$ mer generelt er vektorer definerer vi den relative feilen mellom disse som $\frac{|\mathbf{b}-\mathbf{a}|}{|\mathbf{a}|}$.

Relativ feil sier mer om størrelsesforholdet mellom a og b enn hva den absolute feilen $|b-a|$ gjør, siden den ikke endrer seg hvis vi skalerer tallene: Siden

$\left| \frac{cb-ca}{ca} \right| = \left| \frac{b-a}{a} \right|$ for enhver $c \neq 0$, så er den relative feilen mellom a og b identisk med den relative feilen mellom ca og cb . Det samme er ikke tilfelle for absolutt feil. Relativ feil sier også noe om hvor mange desimale tallene a og b har felles: Hvis den relative feilen er omtrent 10^{-k} , så betyr det at omtrent de k første desimale sifrene i a og b stemmer overens.

For systemene $\mathbf{Ax} = \mathbf{b}$ og $\mathbf{Ay} = \mathbf{b} + \mathbf{e}$ over har vi to forskjellige relative feil:

1. Den relative feilen i høyresiden i systemet: $|(\mathbf{b} + \mathbf{e}) - \mathbf{b}|/|\mathbf{b}| = |\mathbf{e}|/|\mathbf{b}|$.
2. Den relative feilen mellom løsningene: $|\mathbf{y} - \mathbf{x}|/|\mathbf{x}|$.

Forholdet

$$\frac{|\mathbf{y} - \mathbf{x}|/|\mathbf{x}|}{|\mathbf{e}|/|\mathbf{b}|} \quad (9.4)$$

mellom disse sier hvor mye den relative feilen i høyresiden blir propagert til løsningene. Hvis (9.4) er omtrent 10^{-n} , så betyr dette at omtrent n desimale sifre i presisjon kan bli tapt når vi løser det perturberte systemet.

I eksemplet over er den relative feilen i 1. lik $|\mathbf{e}|/|\mathbf{b}| = 10^{-3}$. For den relative feilen i 2. får vi først

$$\mathbf{y} - \mathbf{x} = \begin{pmatrix} 1 + 10^{-3}\epsilon^{-1} \\ 1 - 10^{-3}\epsilon^{-1} \end{pmatrix} - \begin{pmatrix} 1 \\ 1 \end{pmatrix} = 10^{-3}\epsilon^{-1} \begin{pmatrix} 1 \\ -1 \end{pmatrix},$$

og deretter $|\mathbf{y} - \mathbf{x}|/|\mathbf{x}| = 10^{-3}\epsilon^{-1}$. "Propageringsfaktoren" (9.4) blir dermed ϵ^{-1} . Siden dette tallet er stort når ϵ er liten så blir den relative feilen i høyresiden av systemet kraftig forstørret når vi sammenligner løsningene.

9.3 Definisjon av kondisjonstall

I denne seksjonen skal vi utlede et resultat som sier hvor stor propageringsfaktoren (9.4) kan bli. Vi skal vise følgende.

Proposisjon 9.7. Anta at A er symmetrisk og inverterbar, og la

$$|\lambda_1| \geq |\lambda_2| \geq \dots \geq |\lambda_n| > 0$$

være egenverdiene til A listet opp etter avtagende absoluttverdi. Da har vi at

$$\left| \frac{\lambda_n}{\lambda_1} \right| \frac{|\mathbf{e}|}{|\mathbf{b}|} \leq \frac{|\mathbf{y} - \mathbf{x}|}{|\mathbf{x}|} \leq \left| \frac{\lambda_1}{\lambda_n} \right| \frac{|\mathbf{e}|}{|\mathbf{b}|} \quad (9.5)$$

Videre har vi

1. likhet i høyre ulikhet når $\mathbf{e}$ er en egenvektor for λ_n , og $\mathbf{b}$ for λ_1 ,
2. likhet i venstre ulikhet når $\mathbf{e}$ er en egenvektor for λ_1 , og $\mathbf{b}$ for λ_n .

Definerer vi *kondisjonstallet* til A som $K(A) = |\lambda_1/\lambda_n|$ så kan (9.5) skrives

$$\frac{1}{K(A)} \frac{|\mathbf{e}|}{|\mathbf{b}|} \leq \frac{|\mathbf{y} - \mathbf{x}|}{|\mathbf{x}|} \leq K(A) \frac{|\mathbf{e}|}{|\mathbf{b}|} \quad (9.6)$$

I Oppgave 4 fant du at egenverdiene for A er $\epsilon < 1$ og 1 , slik at kondisjonstallet er $K(A) = \epsilon^{-1}$. Merk at

1. høyre ulikhet sier at den relative feilen i høyresiden propagerer maksimalt med faktor $K(A)$ til løsningene,
2. venstre ulikhet kan skrives om til $\frac{|\mathbf{e}|}{|\mathbf{b}|} \leq K(A) \frac{|\mathbf{y} - \mathbf{x}|}{|\mathbf{x}|}$. På denne formen sier den at den relative feilen i $\mathbf{x}$ propagerer maksimalt med en faktor $K(A)$ under matrisemultiplikasjonen $\mathbf{b} = A\mathbf{x}$.

Bevis for Proposisjon 9.7. Vi får først at $A(\mathbf{y} - \mathbf{x}) = \mathbf{e}$, så at $\mathbf{y} - \mathbf{x} = A^{-1}\mathbf{e}$. Bruker vi Lemma 9.5 på dette og på $A\mathbf{x} = \mathbf{b}$ får vi at

$$|\lambda_n||\mathbf{x}| \leq |\mathbf{b}| \leq |\lambda_1||\mathbf{x}| \quad \frac{1}{|\lambda_1|}|\mathbf{e}| \leq |\mathbf{y} - \mathbf{x}| \leq |\mathbf{e}| \frac{1}{|\lambda_n|}$$

Her brukte vi at A^{-1} også er en symmetrisk matrise, med egenverdier $1/\lambda_n, \dots, 1/\lambda_1$, listet opp etter avtagende absoluttverdi, se Oppgave 6. Kombinerer vi disse får vi at

$$\left| \frac{\lambda_n}{\lambda_1} \right| \frac{|\mathbf{e}|}{|\mathbf{b}|} \leq \frac{|\mathbf{y} - \mathbf{x}|}{|\mathbf{x}|} \leq \left| \frac{\lambda_1}{\lambda_n} \right| \frac{|\mathbf{e}|}{|\mathbf{b}|}$$

Videre er det likhet i høyre ulikhet når $\mathbf{e}$ er en egenvektor for λ_n , og $\mathbf{b}$ er en egenvektor for λ_1 , og likhet i venstre ulikhet når $\mathbf{e}$ er en egenvektor for λ_1 , og $\mathbf{b}$ er en egenvektor for λ_n . ■

Siste del av Proposisjon 9.7 sier at vi kan få likhet i begge ulikhetene ved konkrete valg av vektorer/perturbasjoner. Valget i 1. har vi allerede tatt for oss. Valget i 2. gjør vi ved å endre høyresiden til $\mathbf{b} = (1, -1)$ (egenvektor for minste egenverdi), og feilen til $\mathbf{e} = 10^{-3}(1, 1)$ (egenvektor for største egenverdi). Med

$$\mathbf{x} = \epsilon^{-1} \begin{pmatrix} 1 \\ -1 \end{pmatrix} \quad \mathbf{y} = \epsilon^{-1} \begin{pmatrix} 1 \\ -1 \end{pmatrix} + 10^{-3} \begin{pmatrix} 1 \\ 1 \end{pmatrix}.$$

får vi nå

$$\begin{aligned} A\mathbf{x} &= \epsilon^{-1} A\mathbf{v}_2 = \mathbf{v}_2 = \mathbf{b} \\ A\mathbf{y} &= \epsilon^{-1} A\mathbf{v}_2 + 10^{-3} A\mathbf{v}_1 = \mathbf{v}_2 + 10^{-3}\mathbf{v}_1 = \mathbf{b} + \mathbf{e}. \end{aligned}$$

Dette gir

$$\begin{aligned} \|\mathbf{y} - \mathbf{x}\|/\|\mathbf{x}\| &= 10^{-3}\sqrt{2}/(\epsilon^{-1}\sqrt{2}) = 10^{-3}\epsilon \\ \|\mathbf{e}\|/\|\mathbf{b}\| &= 10^{-3}\sqrt{2}/\sqrt{2} = 10^{-3}. \end{aligned}$$

Propageringsfaktoren (9.4) blir dermed ϵ , som gir likhet i venstre ulikhet. Dette valget gir dermed maksimal propagering av relativ feil under matrisemultiplikasjonen $\mathbf{b} = A\mathbf{x}$.

Eksemplet over kan virke opplagt, siden A er nær en singulær matrise når ϵ er liten. Hilbertmatrisen fra Eksempel 2.4 er et annet eksempel på en matrise med høyt kondisjonstall. Dette er ikke like opplagt, og her har man også matriser av alle størrelser. Koden under plottter kondisjonstall for en $n \times n$ -Hilbertmatrise for forskjellige verdier for n .

```
from scipy.linalg import hilbert

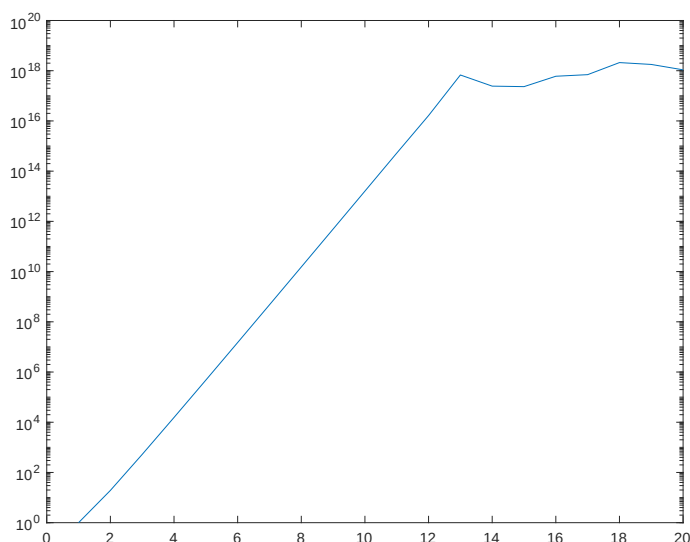
nvals = np.arange(1,21)
ch = np.zeros(20)
for k in nvals:
    ch[k-1] = np.linalg.cond(hilbert(k))
plt.semilogy(nvals,ch)
```

Vi har her brukt kommandoen $\text{cond}(A)$, som returnerer kondisjonstallet til matrisen. Langs x -aksen har vi n . Plottet er logaritmisk (se Oppgave 8 med tanke på kommandoen `semilogy`), dette gjør at vi langs y -aksen har antall riktige siffer vi mister. Vi ser at hilbertmatrisen mister omtrent ett siffer i presisjon hver gang vi øker n . Selv for små matriser og små perturbasjoner blir derfor matrisene vanskelig å bruke i numeriske beregninger.

Kondisjonstallet kan mer generelt defineres for alle matriser, ikke bare for de symmetriske. Vi skal dog ikke gå inn på den generelle definisjonen.

Oppgaver for Kapittel 9

1. Finn en ortogonal diagonalisering av matrisen $A = \begin{pmatrix} 14 & -2 \\ -2 & 11 \end{pmatrix}$.
2. Vis at $U_1 U_2$ er ortogonal når U_1 og U_2 er ortogonale.
3. Vis at $\left(\begin{array}{c|c} U_1 & \mathbf{0} \\ \hline \mathbf{0} & U_2 \end{array} \right)$ er ortogonal når U_1 og U_2 er ortogonale (det er her satt inn nuller slik at den resulterende matrisen blir kvadratisk).
4. Vi setter som før $A = \frac{1}{2} \begin{pmatrix} 1+\epsilon & 1-\epsilon \\ 1-\epsilon & 1+\epsilon \end{pmatrix}$ og $\mathbf{b} = \begin{pmatrix} 1 \\ 1 \end{pmatrix}$.



Figur 9.1. Kondisjonstall for en $n \times n$ -hilbertmatrise, for forskjellige verdier av n

(a). Finn egenverdiene til A og de tilhørende egenvektorene.

(b). Sett $\mathbf{e} = 10^{-3} \begin{pmatrix} 1 \\ -1 \end{pmatrix}$. Vis at løsningen av $\mathbf{A}\mathbf{y} = \mathbf{b} + \mathbf{e}$ er

$$\mathbf{y} = \begin{pmatrix} 1 \\ 1 \end{pmatrix} + 10^{-3} \epsilon^{-1} \begin{pmatrix} 1 \\ -1 \end{pmatrix}.$$

5 (Prøveeksamen 2024). La A være matrisen $A = \begin{pmatrix} 0.2 & 0.8 \\ 0.8 & 0.2 \end{pmatrix}$.

(a). Finn egenverdiene og egenvektorene til A .

(b). Hva er kondisjonstallet $K(A)$?

6. Anta A er en inverterbar, symmetrisk $n \times n$ -matrise, med egenverdier $\lambda_1, \lambda_2, \dots, \lambda_n$. Vis at A^{-1} også er en symmetrisk matrise, med egenverdier $1/\lambda_1, \dots, 1/\lambda_n$.

7. En $n \times n$ -matrise kalles ortogonal hvis $U^{-1} = U^T$. Forklar at kondisjonstallet til U er 1. Dette gjelder uavhengig av om U er symmetrisk eller ikke. Hva sier dette oss om slike matriser med tanke på løse et perturbert system?

8. Hvis $y = Ke^{kx}$, så sier vi at y er *eksponensiell* i x , med konstant K og eksponent k .

(a). Kommandoen `semilogx(x,y)` plotter $\log x$ mot y . Anta at `semilogx(x,y)` viser en linje. Vis at da må x være eksponensiell i y . Hvordan kan vi lese ut konstanten K og eksponenten k ?

(b). Kommandoen `semilogy(x,y)` plotter på samme måte x mot $\log y$. Anta at `semilogy(x,y)` viser en linje. Vis at da må y være eksponensiell i x . Hvordan kan vi lese ut konstanten og eksponenten nå?

Del II

Numerisk analyse

KAPITTEL 10

Interpolasjon med polynomer

En grunnleggende ide i matematikken er å tilnærme noe komplisert med noe enklere, i håp om at tilnærmingen kan fange opp noe av den viktigste informasjonen fra det kompliserte. Dette er den grunnleggende ideen bak approksimasjon med polynomer, som har vært sentral i matematikk siden oppdagelsen av Kalkulus.

Ideen med approksimasjon er ofte nyttig for å studere funksjoner. De fleste matematiske funksjoner kan ikke uttrykkes som kombinasjoner av de elementære funksjonene vi lærte om på skolen. Likevel kan så å si alle funksjoner av praktisk interesse tilnærmes vilkårlig godt med enkle funksjoner som polynomer, trigonometriske funksjoner, eller eksponensialfunksjoner. Innen beregningsvitenskap er særlig polynomer mye brukt.

Du skal allerede ha lært om approksimasjon med *Taylorpolynomer*, som gir en enkel tilnærming basert på informasjon om funksjonen i kun ett punkt, nærmere bestemt verdien av f og dens deriverte i dette punktet: Hvis $f: \mathbb{R} \rightarrow \mathbb{R}$ er en gitt funksjon som er n ganger deriverbar i et punkt $a \in \mathbb{R}$, så vet vi fra Seksjon 11.1 i *Kalkulus* at

$$p(x) = f(a) + f'(a)(x-a) + \frac{f''(a)}{2!}(x-a)^2 + \cdots + \frac{f^{(n)}(a)}{n!}(x-a)^n$$

er det entydige polynomet av grad n med de samme n første deriverte som f i a . Én måte å vise dette på er ved først å observere at ethvert polynom p av grad n kan skrives unikt som

$$p(x) = c_0 + c_1(x-a) + c_2(x-a)^2 + \cdots + c_n(x-a)^n.$$

Deriverer vi denne funksjonen k ganger i a får vi at $p^{(k)}(a) = c_k k!$, og om dette skal være lik $f^{(k)}(a)$, må vi ha $c_k = f^{(k)}(a)/k!$, som var det vi skulle vise. Merk at

likningene $p^{(k)}(a) = c_k k!$ kan skrives som det enkle diagonale systemet

$$\begin{pmatrix} 0! & 0 & 0 & 0 & \cdots & 0 \\ 0 & 1! & 0 & 0 & \cdots & 0 \\ 0 & 0 & 2! & 0 & \cdots & 0 \\ 0 & 0 & 0 & 3! & \cdots & 0 \\ \vdots & \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & 0 & \cdots & n! \end{pmatrix} \begin{pmatrix} c_0 \\ c_1 \\ c_2 \\ c_3 \\ \vdots \\ c_n \end{pmatrix} = \begin{pmatrix} f(a) \\ f'(a) \\ f''(a) \\ f'''(a) \\ \vdots \\ f^{(n)}(a) \end{pmatrix}.$$

Vi kommer tilbake til dette. *Taylors formel med restledd* sier at hvis f er $n+1$ ganger kontinuerlig deriverbar, så finnes det for enhver x , et tall $\xi \in [a, x]$, s.a.

$$f(x) = p(x) + \frac{(x-a)^{n+1}}{(n+1)!} f^{(n+1)}(\xi). \quad (10.1)$$

Dersom x er nær a , og den deriverte $f^{(n+1)}$ ikke er veldig stor, vil restleddet være lite, og p er dermed nær f . Med andre ord: Om de deriverte til f ikke er veldig store, vil taylorpolynomet p være en god approksimasjon av f , i alle fall nær a .

I dette kapitlet skal vi se på *interpolasjon*, som i stedet gir tilnærminger basert på et sett funksjonsverdier. Vi sier at funksjonen $p: [a, b] \rightarrow \mathbb{R}$ *interpolerer* f i punktene $x_0, x_1, \dots, x_n \in [a, b]$ (som vi antar er forskjellige) hvis

$$f(x_k) = p(x_k) \quad \text{for } k = 0, 1, \dots, n. \quad (10.2)$$

$x_0, x_1, \dots, x_n$ kalles *interpolasjonspunktene*. Funksjonen f kan, men behøver ikke, være kjent: Alt vi trenger fra denne er funksjonsverdiene $f(x_i)$. *polynominterpolasjon*, dreier seg om å finne et polynom p som interpolerer f . Siden betingelsen (10.2) består av $n+1$ ligninger, og et n -tegradspolynom har $n+1$ koeffisienter som kan velges fritt, gir det mening å la interpolasjonspolynomet p være av orden n .

Vi skal se på to måter for å finne interpolasjonspolynomet: Ved hjelp av kanonisk form (Seksjon 10.1), og ved hjelp av Newtonformen (Seksjon 10.2). Vi skal også se at feilen ved interpolasjon kan uttrykkes på en måte som ligner på restleddsformelen (10.1), se Seksjon 10.3. I de neste kapitlene skal vi bruke interpolasjon til å utlede numeriske metoder for de deriverte, og integraler, av funksjoner.

10.1 Interpolasjon på kanonisk form

Et generelt n -tegradspolynom kan skrives på *kanonisk form*

$$p_n(x) = c_0 + c_1 x + c_2 x^2 + \cdots + c_n x^n. \quad (10.3)$$

Interpolasjonsbetingelsene tar formen

$$\begin{aligned} f(x_0) &= c_0 + c_1 x_0 + c_2 x_0^2 + \cdots + c_n x_0^n \\ f(x_1) &= c_0 + c_1 x_1 + c_2 x_1^2 + \cdots + c_n x_1^n \\ f(x_2) &= c_0 + c_1 x_2 + c_2 x_2^2 + \cdots + c_n x_2^n \\ &\vdots \\ f(x_n) &= c_0 + c_1 x_n + c_2 x_n^2 + \cdots + c_n x_n^n. \end{aligned}$$

På matriseform kan dette skrives

$$\begin{pmatrix} 1 & x_0 & x_0^2 & \cdots & x_0^n \\ 1 & x_1 & x_1^2 & \cdots & x_1^n \\ 1 & x_2 & x_2^2 & \cdots & x_2^n \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 1 & x_n & x_n^2 & \cdots & x_n^n \end{pmatrix} \begin{pmatrix} c_0 \\ c_1 \\ c_2 \\ \vdots \\ c_n \end{pmatrix} = \begin{pmatrix} f(x_0) \\ f(x_1) \\ f(x_2) \\ \vdots \\ f(x_n) \end{pmatrix} \quad (10.4)$$

Matrisen i (10.4) er en kvadratisk vandermondematrikse. Fra Eksempel 4.7 vet vi at slike matriser er inverterbare. Derfor har (10.4) en unik løsning for $\mathbf{c}$, og vi kan konkludere at det finnes et unikt interpolerende polynom. $\mathbf{c}$ kan vi finne med koden under.

```
n = np.shape(x)[0] - 1
p = np.arange(0,n+1).reshape(1,-1)
V = x.reshape(-1,1)**p
c = np.linalg.inv(V)*f
```

Siden vi ikke er interessert i hele den inverse matrisen her, bare i løsningen på systemet $V\mathbf{c} = \mathbf{f}$, så er det mer effektivt i stedet å skrive

```
c = np.linalg.solve(V,f)
```

Etter å ha funnet $\mathbf{c}$ så kan vi gjenta det vi gjorde i Seksjon 2.4 for å plotte det interpolerende polynomet som følger.

```
x_plot = np.linspace(x[0]-0.1, x[-1]+0.1, 100).reshape((-1,1))
p = np.arange(0,n+1).reshape(1,-1)
x_plot_powers = x_plot**p
plt.plot(x_plot, x_plot_powers@c)
```

Listing 10.1. Kode for å plotte det interpolerende polynomet

Noen kommentarer om denne koden er på sin plass:

- Koden antar at vektoren x inneholder interpolasjonspunktene, og at vektoren f inneholder de tilhørende interpolasjonsverdiene. Vektorene antas å ha lik lengde.
- Den andre linja regner ut komponentvis potens, og returnerer vandermondematrixen

$$V = \begin{pmatrix} 1 & x_0 & x_0^2 & \cdots & x_0^n \\ 1 & x_1 & x_1^2 & \cdots & x_1^n \\ 1 & x_2 & x_2^2 & \cdots & x_2^n \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 1 & x_n & x_n^2 & \cdots & x_n^n \end{pmatrix}.$$

- Den tredje linja finner koeffisientene c til det interpolerende polynomet ved å løse Likning (10.4).
- De siste tre linjene sørger for selve plottingen. Først lages en søylevektor x_{plot} med plottepunkter, og som dekker interpolasjonspunktene. x_{plot} er typisk mye lengre enn vektoren x .
- En matrise $x_{\text{plot_powers}}$ regnes så ut ved å opphøye søylevektor x_{plot} til de samme potensene som på den første linja. Vi får da en ny vandermondematrix, med mange flere rader.
- På siste linje plottes det interpolerende polynomet. Koden $x_{\text{plot_powers}} * c$ gir en søylevektor der hvert element representerer en verdi av polynomet.

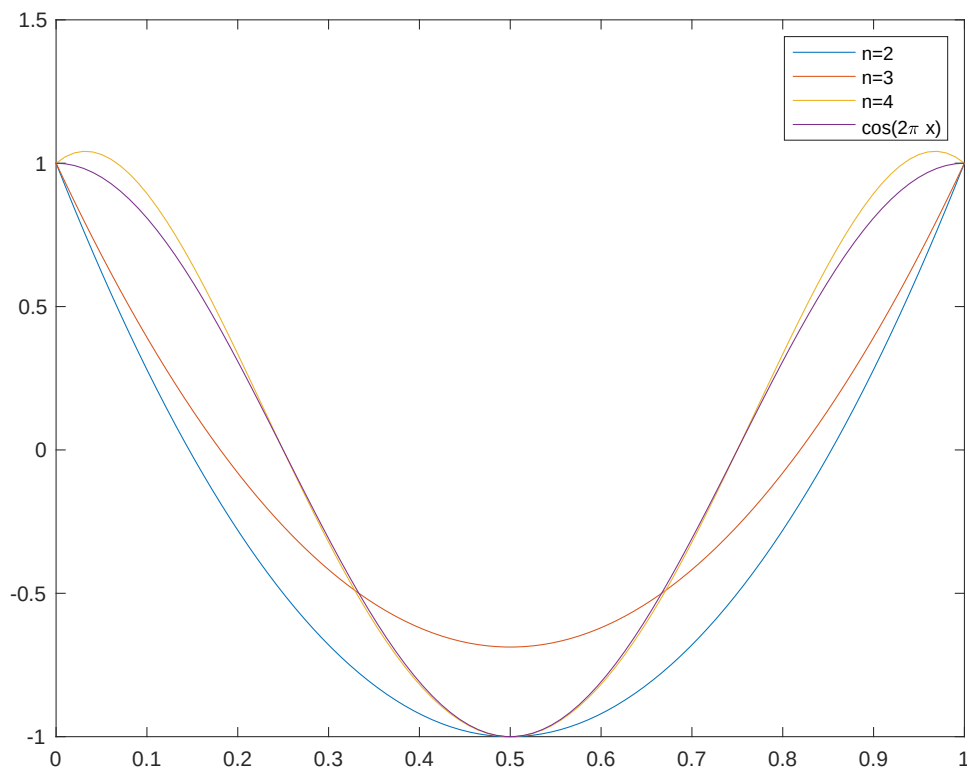
Koden over prioriterer oversiktighet fremfor effektivitet og minnebruk. Utregningen av vandermondematrixen V kan gjøres ved hjelp av ca. n^2 regneoperasjoner på datamaskinen (se Oppgave 2). I Seksjon 6.2 så vi at det krever $\approx \frac{2}{3}n^3$ operasjoner å løse et generelt system med n likninger og n ukjente. Med andre ord: Det er mer krevende å løse systemet enn å regne ut matrixen for systemet.

I Figur 10.1 har vi plottet de interpolerende polynomene p_2 , p_3 , og p_4 , for $f(x) = \cos(2\pi x)$, gitt en uniform partisjon av $[0, 1]$.

Eksempel 10.1. Anta vi vil finne et polynom som går gjennom de tre punktene $(0, 1)$, $(1, 3)$, and $(2, 2)$. Med andre ord, vi vil finne et polynom p slik at

$$p(0) = 1, \quad p(1) = 3, \quad p(2) = 2. \quad (10.5)$$

Siden det er tre punkter er det naturlig å anta et kvadratisk polynom $p(x) = c_0 +$



Figur 10.1. De interpolerende polynomene p_2 , p_3 , og p_4 , for $f(x) = \cos(2\pi x)$. f er også vist.

$c_1x + c_2x^2$. Setter vi inn i (10.5) så får vi de tre likningene

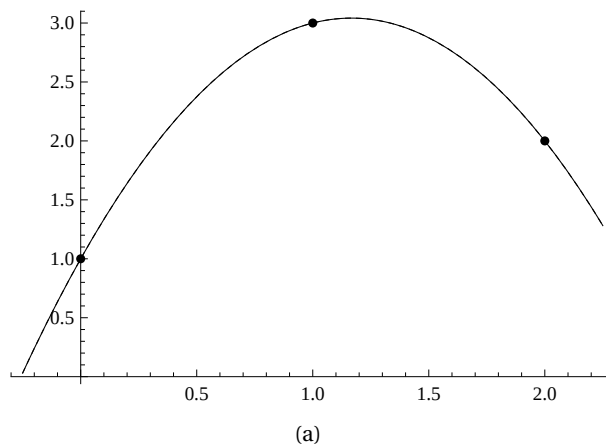
$$\begin{aligned} 1 &= p(0) = c_0, \\ 3 &= p(1) = c_0 + c_1 + c_2, \\ 2 &= p(2) = c_0 + 2c_1 + 4c_2. \end{aligned}$$

Vi løser disse og finner $c_0 = 1$, $c_1 = 7/2$, and $c_2 = -3/2$, slik at p er gitt ved

$$p(x) = 1 + \frac{7}{2}x - \frac{3}{2}x^2.$$

Et plott av dette polynomet og interpolasjonspunktene er vist i Figur 10.2. Plottet får du ved å kjøre koden i Program 10.1 etter å ha definert

```
x = np.array([[0], [1], [2]])
f = np.array([[1], [3], [2]])
```



Figur 10.2. Tre interpolasjonspunkter og det tilsvarende kvadratiske interpolasjonspolynomiet.

10.2 Interpolasjon med Newtonformen

La igjen $x_0, x_1, \dots, x_n$ være interpolasjonspunktene. Vi sier at polynomet p_n er på *newtonform* dersom det er skrevet på formen

$$p_n(x) = c_0 + c_1(x - x_0) + c_2(x - x_0)(x - x_1) + \dots + c_n(x - x_0)(x - x_1) \cdots (x - x_{n-1}). \quad (10.6)$$

Newtonformen har en del fordeler sammenlignet med den kanoniske. Interpolasjonsbetingelsene tar nå formen

$$\begin{aligned} f(x_0) &= c_0 \\ f(x_1) &= c_0 + c_1(x_1 - x_0) \\ &\vdots \\ f(x_n) &= c_0 + c_1(x_n - x_0) + \cdots + c_n(x_n - x_0) \cdots (x_n - x_{n-1}), \end{aligned} \quad (10.7)$$

eller på matriseform:

$$\begin{pmatrix} 1 & 0 & \cdots & 0 \\ 1 & x_1 - x_0 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 1 & x_n - x_0 & \cdots & (x_n - x_0) \cdots (x_n - x_{n-1}) \end{pmatrix} \begin{pmatrix} c_0 \\ c_1 \\ \vdots \\ c_n \end{pmatrix} = \begin{pmatrix} f(x_0) \\ f(x_1) \\ \vdots \\ f(x_n) \end{pmatrix}. \quad (10.8)$$

Dette er et *triangulært system*, og hvis alle interpolasjonspunktene er forskjellige vil alle elementene langs diagonalen være ulik 0. Dermed er også determinanten til matrisen ulik 0, så matrisen er inverterbar. Matriseligningen (10.8) har

dermed en unik løsning $(c_0, c_1, \dots, c_n)$. Med andre ord, vi har en unik Newton-form.

Triangulære systemer er enklere å løse enn generelle systemer. For å se dette, kall den triangulære matrisen for A , la $A = (a_{ij})_{i,j=0}^n$ være elementene i A , og la $A\mathbf{c} = \mathbf{f}$ være systemet vi skal løse. Det er klart at vi kan finne c_0 fra den første likningen, siden denne bare sier $c_0 = f_0$ (husk at $a_{00} = 1$). Anta så at vi har klart å finne $c_0, c_1, \dots, c_{k-1}$ fra de k første likningene. Likning $k+1$ sier at

$$a_{k0}c_0 + a_{k1}c_1 + \dots + a_{k(k-1)}c_{k-1} + a_{kk}c_k = f_k.$$

Vi kan skrive dette som

$$c_k = \frac{1}{a_{kk}} (f_k - a_{k0}c_0 - a_{k1}c_1 - \dots - a_{k(k-1)}c_{k-1}) \quad (10.9)$$

(diagonalelementet $a_{kk} = x_k - x_0$ er ulik 0, så dette uttrykket gir mening). Dermed kan vi finne c_k ut fra likning $k+1$ i systemet. Denne metoden for å løse et triangulært system kalles *fremoversubstitusjon*: Vi finner først c_0 , bruker denne til å finne c_1 , bruker deretter disse til å finne c_2 , og så videre.

Formelen (10.9) for å finne c_k krever k subtraksjoner og k multiplikasjoner. Summerer vi over alle k får vi

$$2 + 4 + 6 + \dots + 2(n-1) + 2n = 2(1 + 2 + \dots + n) = n(n+1) \approx n^2$$

operasjoner (vi tilnærmer med n^2 siden n^2 er det dominerende leddet her). Dette er en stor reduksjon når vi sammenligner med $\approx \frac{2}{3}n^3$ operasjoner for det kanoniske systemet. Ved å tilpasse formen på interpolasjonspolynomet, oppnår vi dermed en regnemessig reduksjon i metoden vår. Den viktige lærdommen fra newtonformen er derfor at det kan være lurt å se på andre måter å representere polynomer på også.

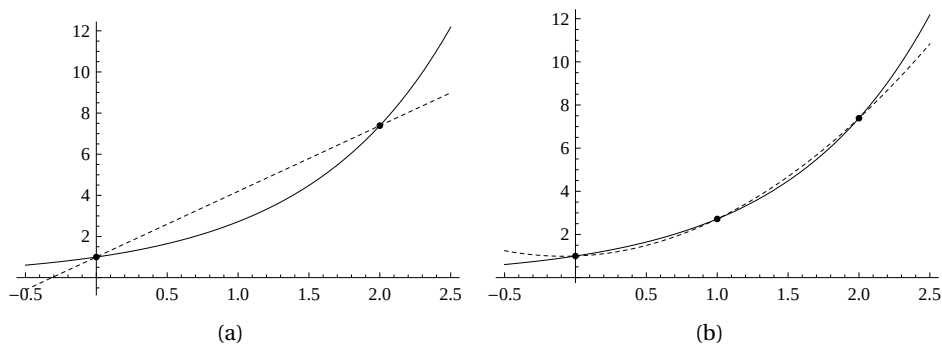
Den triangulære matrisen kan også regnes ut med n^2 operasjoner (se Oppgave 9, som også viser kode for å plote interpolasjonspolynomet via newtonformen, tilsvarende Program 10.1).

Eksempel 10.2 (Newtonformen for $n = 0$). Hvis vi bare har ett interpolasjonspunkt x_0 så blir newtonformen $p_0(x) = c_0$. Vi får at $c_0 = f(x_0)$, slik at

$$p_0(x) = f(x_0).$$

Eksempel 10.3 (Newtonformen for $n = 1$). Med to punkter x_0 og x_1 blir Newtonformen $p_1(x) = c_0 + c_1(x - x_0)$. Vi får likningene

$$\begin{aligned} f(x_0) &= c_0 \\ f(x_1) &= c_0 + c_1(x_1 - x_0) \end{aligned}$$



Figur 10.3. Interpolasjon av e^x i to punkter med en sekant (a), og i tre punkter med en parabel (b).

som har løsning

$$c_0 = f(x_0), \quad c_1 = \frac{f(x_1) - f(x_0)}{x_1 - x_0}. \quad (10.10)$$

Dette gir

$$p_1(x) = f(x_0) + \frac{f(x_1) - f(x_0)}{x_1 - x_0}(x - x_0),$$

som ikke er noe annet enn sekanten til f gjennom x_0 og x_1 (se Figur 10.3(a)). Til sammenligning ga det førsteordens Taylorpolynom til f tangenten i punktet.

Eksempel 10.4 (Newtonformen for $n = 2$). Vi legger til enda et punkt x_2 og setter opp et kvadratisk interpolasjonspolynom:

$$p_2(x) = c_0 + c_1(x - x_0) + c_2(x - x_0)(x - x_1)$$

Setter vi inn $x = x_0$ og $x = x_1$ får vi likningene

$$\begin{aligned} f(x_0) &= c_0, \\ f(x_1) &= c_0 + c_1(x_1 - x_0), \end{aligned}$$

Dette er de samme likningene vi løste for $n = 1$, så c_0 og c_1 må være gitt ved (10.10) også når $n = 2$. Setter vi inn $x = x_2$ får vi

$$f(x_2) = c_0 + c_1(x_2 - x_0) + c_2(x_2 - x_0)(x_2 - x_1),$$

som gir

$$c_2 = \frac{f(x_2) - f(x_0) - \frac{f(x_1) - f(x_0)}{x_1 - x_0}(x_2 - x_0)}{(x_2 - x_0)(x_2 - x_1)}.$$

Etter en del regning kan dette også skrives

$$c_2 = \frac{\frac{f(x_2)-f(x_1)}{x_2-x_1} - \frac{f(x_1)-f(x_0)}{x_1-x_0}}{x_2-x_0}. \quad (10.11)$$

Interpolasjonspolynomet blir nå en parabel, se Figur 10.3(b).

Mer generelt ser vi fra det triangulære systemet (10.7) at likningen som stammer fra innsettingen $x = x_k$ bare involverer punktene $(x_0, f(x_0)), (x_1, f(x_1)), \dots, (x_k, f(x_k))$. Spesielt vil c_k bare avhenge av disse punktene. Det er derfor vanlig å skrive

$$c_k = f[x_0, \dots, x_k] \quad (10.12)$$

for $k = 0, 1, \dots, n$, slik at newtonformen

$$p_n(x) = c_0 + c_1(x - x_0) + \dots + c_n(x - x_0)(x - x_1) \cdots (x - x_{n-1}) \quad (10.13)$$

kan skrives

$$\begin{aligned} p_n(x) = & f[x_0] + f[x_0, x_1](x - x_0) + f[x_0, x_1, x_2](x - x_0)(x - x_1) \\ & + \dots + f[x_0, x_1, \dots, x_n](x - x_0)(x - x_1) \cdots (x - x_{n-1}). \end{aligned} \quad (10.14)$$

Merk at interpolasjonspolynomene p_n og p_{n-1} oppfyller

$$p_n(x) = p_{n-1}(x) + f[x_0, \dots, x_n](x - x_0) \cdots (x - x_{n-1}).$$

Figur 10.4 viser det interpolerende polynomet p_n for forskjellige verdier av n for funksjonen $f(x) = \sin(x)$. Vi ser at tilnærmingen blir bedre etterhvert som vi øker graden n .

Eksempel 10.5. Vi interpolerer funksjonen $f(x) = \sqrt{x}$ i punktene $x_i = i$, for $i = 0, 1, 2, 3$. Newtonformen blir

$$p_3(x) = c_0 + c_1x + c_2x(x-1) + c_3x(x-1)(x-2),$$

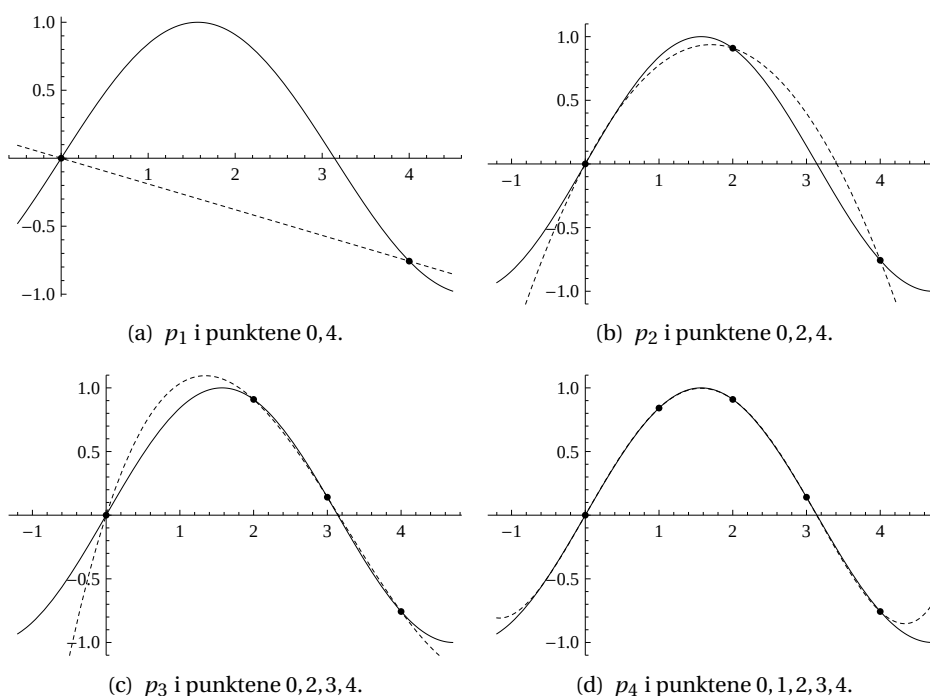
og interpolasjonsbetingelsene gir

$$\begin{aligned} 0 &= c_0, \\ 1 &= c_0 + c_1, \\ \sqrt{2} &= c_0 + 2c_1 + 2c_2, \\ \sqrt{3} &= c_0 + 3c_1 + 6c_2 + 6c_3. \end{aligned}$$

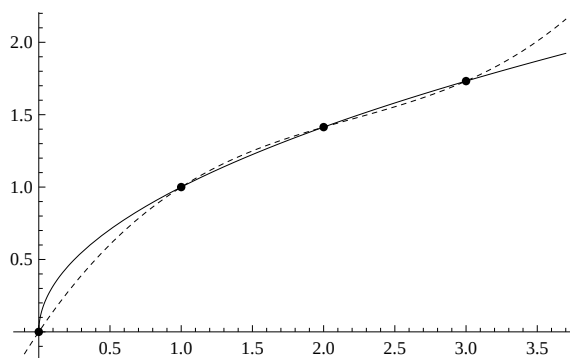
Løser vi dette ligningssystemet får vi

$$c_0 = 0, \quad c_1 = 1, \quad c_2 = -(1 - \sqrt{2}/2), \quad c_3 = (3 + \sqrt{3} - 3\sqrt{2})/6.$$

Figur 10.5 viser det interpolerende polynomet.

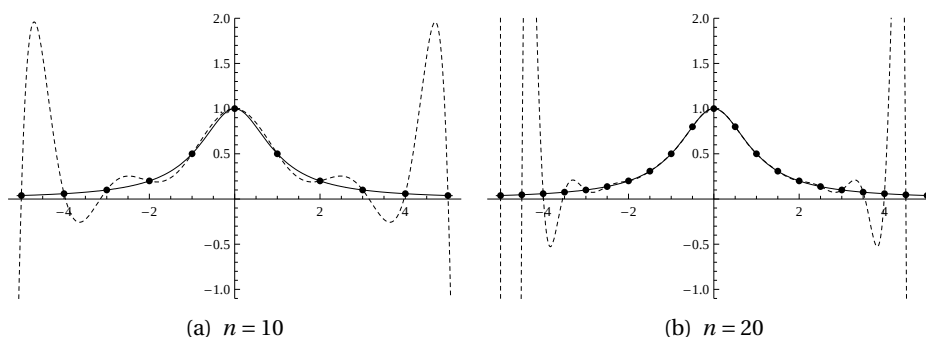


Figur 10.4. Interpolasjon av $f(x) = \sin(x)$ med det interpolerende polynomet p_n for forskjellige n .



Figur 10.5. Funksjonen $f(x) = \sqrt{x}$ (heltrukken linje) og den kubiske interpolanten i de fire punktene 0, 1, 2, og 3 (stiplet linje).

Hvis interpolasjon brukes riktig, vil den vanligvis gi en god tilnærming til den underliggende funksjonen. Etterhvert som avstanden mellom punktene avtar (enten ved å øke antall punkter eller flytte punktene nærmere hverandre), kan man forvente at tilnærmingen blir bedre. Men vi vet at det finnes funksjoner



Figur 10.6. Interpolasjon av funksjonen $f(x) = 1/(1+x^2)$ på intervallet $[-5, 5]$ med polynomer av grad n . Punktene er uniformt fordelt i intervallet i hvert tilfelle.

der taylorpolynomer er dårlige tilnærminger, og det samme kan skje for interpolasjon. Dette problemet kan vi få når den deriverte til funksjonen blir veldig stor. Under er et viktig eksempel på dette.

Eksempel 10.6. *Rungefunksjonen* er definert ved $f(x) = 1/(1+x^2)$. La oss se på denne på intervallet $[-5, 5]$. Figur 10.6 viser oss interpolanter av grad 10 og 20. Midt på intervallet blir feilen mindre når vi øker graden, men nær endepunktene på intervallet blir feilen større når vi øker graden. Vi skal undersøke hva som går galt i Seksjon 10.3.

Newtonformen og den kanoniske formen er bare to måter å skrive det interpolerende polynomet på — det finnes mange andre måter også. En av disse er *Lagrangeformen*, som vi ser på i Oppgave 6.

10.3 Interpolasjonsfeilen

I forrige seksjon så vi eksempler der det interpolerende polynomet p_n blir en bedre og bedre tilnærming til den underliggende funksjonen når n øker, og eksempler der det motsatte skjer. Vi skal nå gi en matematisk begrunnelse for hvorfor dette skjer.

Teorem 10.7. La $x_0, x_1, \dots, x_n \in [a, b]$ være distinkte interpolasjonspunkter, og anta at funksjonen $f: [a, b] \rightarrow \mathbb{R}$ er (minst) $n+1$ ganger kontinuerlig deriverbar. La p_n være det unike n -teordens interpolerende polynomet til f i punktene $x_0, x_1, \dots, x_n$. Da vil

$$|f(x) - p_n(x)| \leq \frac{M_{n+1}(b-a)^{n+1}}{(n+1)!} \quad (10.15)$$

for alle $x \in [a, b]$, der $M_{n+1} = \max_{[a,b]} |f^{(n+1)}|$.

Bevis. La $x \in [a, b]$. Vi påstår først at det finnes et punkt $y \in [a, b]$ slik at

$$f(x) - p_n(x) = \frac{f^{(n+1)}(y)}{(n+1)!} Q_n(x), \quad (10.16)$$

der Q_n er $(n+1)$ -tegradspolynomet

$$Q_n(x) = (x - x_0)(x - x_1) \cdots (x - x_n).$$

Dersom x er et av punktene $x_0, x_1, \dots, x_n$, er begge sider av (10.16) lik 0, så identiteten er automatisk sann. Anta så at x ikke er et av punktene $x_0, x_1, \dots, x_n$, og definer funksjonen

$$\phi(t) = f(t) - p_n(t) - \frac{f(x) - p_n(x)}{Q_n(x)} Q_n(t) \quad \text{for } t \in [a, b].$$

Da har ϕ nullpunkter i de $n+2$ distinkte punktene $x, x_0, x_1, \dots, x_n$ (dette er Oppgave 12). Av middelverdisetningen følger det dermed at det finnes $n+1$ punkter i intervallet (a, b) der ϕ' er lik 0. Med samme argumentasjon ser vi at det må finnes n punkter der $(\phi')' = \phi''$ er lik 0. Fortsetter vi slik, finner vi at funksjonen $\phi^{(n+1)}$ har minst ett nullpunkt $y \in (a, b)$. Siden $p_n^{(n+1)} \equiv 0$ og $Q^{(n+1)} \equiv (n+1)!$ (dette er Oppgave 13), får vi

$$0 = \phi^{(n+1)}(y) = f^{(n+1)}(y) - \frac{f(x) - p_n(x)}{Q_n(x)} (n+1)!.$$

Om vi stokker om på denne ligningen får vi nettopp (10.16).

Om vi erstatter $|f^{(n+1)}(y)|$ på høyresiden i (10.16) med dets eget maksimum, får vi

$$|f(x) - p_n(x)| \leq \frac{M_{n+1}}{(n+1)!} |Q_n(x)|.$$

Siden hvert av leddene $(x - x_0), \dots, (x - x_n)$ kan være høyst lik $b - a$ (fordi punktene $x, x_0, x_1, \dots, x_n$ alle ligger i $[a, b]$), ser vi at $|Q_n(x)| \leq (b - a)^{n+1}$ for alle x . Dette beviser (10.15). ■

Siden fakulteten $(n+1)!$ blir veldig stor når n vokser, ser vi at interpolasjonsfeilen minsker når n vokser, så lenge M_{n+1} ikke vokser for fort!

Eksempel 10.8. La oss interpolere funksjonen $f(x) = \sin(x)$ på intervallet $[a, b] = [0, 2\pi]$ (se Figur 10.4 for en illustrasjon). Vi velger $n+1$ uniformt fordelte interpolasjonspunkter $x_0, x_1, \dots, x_n$ og får n -tegradspolynomet p_n . Vi vet at de deriverte

av f er enten $\pm \cos(x)$ eller $\pm \sin(x)$, og ingen av disse funksjonene oppnår funksjonsverdier større enn 1. Derfor er $M_{n+1} \leq 1$ for alle n . Vi får dermed at

$$|f(x) - p_n(x)| \leq \frac{(2\pi)^{n+1}}{(n+1)!}.$$

Siden $(n+1)!$ vokser mye raskere enn $(2\pi)^{n+1}$ når n vokser, vil feilen gå raskt mot 0 når n vokser.

Eksempel 10.9. La f være *rungefunksjonen* fra Eksempel 10.6, gitt ved $f(x) = 1/(1+x^2)$ for $x \in [-5, 5]$. En litt teknisk utregning viser at $M_{n+1} \approx (n+1)!$ på intervallet $[-5, 5]$. Feilestimatet (10.15) sier dermed at

$$|f(x) - p_n(x)| \leq \frac{M_{n+1}(b-a)^{n+1}}{(n+1)!} \approx (b-a)^{n+1} = 10^{n+1},$$

som er et veldig stort tall når n er stor. Feilestimatet garanterer altså *ikke* at interpolasjonsfeilen er liten når n er stor, og som vi så i Eksempel 10.6, er feilen faktisk ganske stor.

Oppgaver for Kapittel 10

1. Marker følgende påstander som sanne eller usanne.

(a). Anta polynomet p_n interpolerer f i $n+1$ punkter, og at polynomet q_n interpolerer g i de samme punktene. Det interpolerende polynomet til $f+g$ i de samme punktene er da lik $p_n + q_n$.

(b). Anta polynomet p_n interpolerer f i $n+1$ punkter, og at polynomet q_n interpolerer g i de samme punktene. Det interpolerende polynomet til fg i de samme punktene er da lik $p_n q_n$.

(c). Det finnes mange polynomer av grad n som interpolerer en gitt funksjon f i $n+1$ punkter.

2. I denne oppgaven skal vi se på kode som regner ut matrisen vi trenger for å finne den kanoniske formen for interpolasjonspolynomet. I Listing 10.1 regnet vi ut vandermondematriksen med koden

```
p = np.arange(0,n+1).reshape(1,-1)
A = x**p
```

Sammenlign dette med følgende alternative kode:

```
A[:,0] = np.ones((n+1,1))
for k in range(1,n+1):
    A[:,[k]] = A[:,[k-1]]*x
```

Hvilken av de to alternativene er å foretrekke? Begrunn svaret, og hvilke antagelser du gjør.

3. I denne oppgaven skal vi se på et alternativt bevis for at interpolasjonspolynomet er unikt.

(a). Anta at det finnes to kvadratiske polynomer p_1 og p_2 som interpolerer funksjonen f i de tre punktene x_0 , x_1 og x_2 . Se på differansen $p = p_2 - p_1$. Hva er verdiene til p i interpolasjonspunktene?

(b). Bruk observasjonen i (a) til å vise at p_1 og p_2 må være like.

(c). Generaliser resultater fra (a) og (b) til polynomer av grad n .

4. En rett-fram utregning av (10.3) i et punkt x krever n multiplikasjoner og n addisjoner i tillegg til utregning av potensene ($n - 1$ multiplikasjoner). Totalt blir dette $3n - 1$ operasjoner. La oss skrive om (10.3) til

$$p_n(x) = c_0 + x(c_1 + c_2x + c_3x^2 + \cdots c_nx^{n-1}). \quad (10.17)$$

Inne i parentesene på høyre side her er det et polynom av grad $n - 1$ (som vi kaller p_{n-1}), som på samme måte kan uttrykkes med et polynom av grad $n - 2$ (p_{n-2}), o.s.v.. Vi har at

$$\begin{aligned} p_n(x) &= c_0 + xp_{n-1}(x) \\ p_{n-1}(x) &= c_1 + xp_{n-2}(x) \\ p_{n-2}(x) &= c_2 + xp_{n-3}(x) \\ &\vdots \\ p_1(x) &= c_{n-1} + xp_0(x) \\ p_0(x) &= c_n \end{aligned}$$

For å regne ut $p_n(x)$ starter vi fra siste likning, og fortsetter oppover til første likning. Denne algoritmen kalles også for *Horner's regel*.

(a). Lag en implementasjon av Horner's regel.

(b). Hvor mange aritmetiske operasjoner trenger Horner's regel?

5. Vi interpolerer funksjonen $f(x) = x^2$ med et polynom p_3 av grad 3, i punktene 0, 1, 2, og 3. Hva blir $p_3(4)$?

6. Vi har dataene

x	0	1	3	4
$f(x)$	1	0	2	1

(a). Skriv det kubiske polynomet som interpolerer i disse punktene som

$$p_3(x) = c_0(x-1)(x-3)(x-4) + c_1x(x-3)(x-4) + c_2x(x-1)(x-4) + c_3x(x-1)(x-3),$$

og finn koeffisientene fra interpolasjonsbetingelsene. Dette kalles *Lagrange-formen* til det interpolerende polynomet.

(b). Finn newtonformen til det interpolerende polynomet.

(c). Verifiser at løsningene du fant i (a) og (b) er like.

7. Interpolasjon av et sett med data.

(a). Vi har dataene

x	0	1	2
$f(x)$	2	1	0

Disse er punkter på den rette linja $y = 2 - x$. Finn newtonformen til det kvadratiske interpolerende polynomet og sammenlign. Hva er forskjellen?

(b). Anta at vi interpolerer f med polynomer av grad n i punktene $x_0, \dots, x_n$. Vis at hvis f er et polynom av grad n , så vil interpolanten p_n være identisk med f . Hvorfor forklarer dette resultatet i (a)?

8. Anta at vi har dataene

$$(0, y_0), \quad (1, y_1), \quad (2, y_2), \quad (3, y_3) \quad (10.18)$$

der $y_i = f(i)$ er funksjonsverdier av en ukjent funksjon f . I denne oppgaven skal vi finne uttrykk for tilnærminger til f i forskjellige punkter ved å bruke kubisk interpolasjon.

(a). Finn den rette linjen p_1 som interpolerer de to midterste punktene i (10.18), og bruk $p_1(3/2)$ som en tilnærming til $f(3/2)$. Vis at

$$p_1(3/2) = \frac{1}{2}(y_1 + y_2).$$

(b). Finn det kubiske polynomet p_3 som interpolerer dataene (10.18) og bruk $p_3(3/2)$ som en tilnærming til $f(3/2)$. Vis at

$$p_3(3/2) = \frac{-y_0 + 9y_1 + 9y_2 - y_3}{16}.$$

9. I denne oppgaven skal vi tilpasse plotting av interpolasjonspolynomet (Program 10.1) til newtonformen. Se på følgende kode:

```
n = np.shape(x)[0] - 1
A = np.zeros((n+1,n+1))
A[:,0] = 1;
for c in range(1,n+1):
    A[c:,c] = A[c:,c-1]*(x[c:,0]-x[c-1,0])

c = np.zeros((n+1,1))
for k in range(0,n+1):
    c[k,0] = (f[k,0] - A[k,:k]@c[0:k,0])/A[k,k]

x_plot = np.linspace(x[0]-0.1,x[-1]+0.1,100).reshape((-1,1))
x_plot_powers = np.ones(( np.shape(x_plot)[0], n+1 ))
for d in range(1,n+1):
    x_plot_powers[:,d] = x_plot_powers[:,d-1]*(x_plot[:,0]-x[d-1])
plt.plot(x_plot, x_plot_powers@c)
```

Listing 10.2. Plotting av interpolasjonspolynomet på Newtonform

Koden antar som før at interpolasjonspunktene x , med tilhørende verdier f , allerede er definert. Forklar hva som skjer i de tre løkkene. Forklar spesielt den delen som er ansvarlig for å gjøre fremoversubstitusjon. Verifiser at koden trenger omtrent $2n^2$ operasjoner for å finne newtonformen (n^2 operasjoner for å finne matrisen, og ytterligere n^2 operasjoner for å løse systemet med fremoversubstitusjon).

10. I noen sammenhenger kjenner vi kun til noen punktverdier, og noen verdier av de deriverte til en funksjon.

(a). Anta at f er deriverbar i x_1 , og at x_0, x_1 , og x_2 er forskjellige. Avgjør om det finnes et unikt kubisk polynom p_3 slik at

$$p_3(x_0) = f(x_0), \quad p_3(x_1) = f(x_1), \quad p'_3(x_1) = f'(x_1), \quad p_3(x_2) = f(x_2).$$

(b). Kan du generalisere dette resultatet til tilfeller der noen av de deriverte i interpolasjonspunktene også er kjent? Kan du sette dette i sammenheng med Taylorpolynomer?

11. Vi tilnærmer $f(x) = e^x$ med et interpolerende polynom p_n i punktene

$$0, 1/n, 2/n, \dots, (n-1)/n, 1.$$

Hvor stor må n være for at feilen $|f(x) - p_n(x)|$ skal være mindre enn 0.001 for alle $x \in [0, 1]$?

12. Vis at $\phi(t)$ i beviset for Teorem 10.15 oppfyller

$$\phi(x) = \phi(x_0) = \phi(x_1) = \dots = \phi(x_n) = 0$$

13. Vi har sett at

$$Q_n(x) = (x - x_0)(x - x_1) \cdots (x - x_n)$$

er sentral i Newtonformen, og i beviset for Teorem 10.15. I de neste kapitlene skal vi derivere og integrere det interpolerende polynomet, og vi vil derfor ha behov for å derivere og integrere Q_n .

(a). Vis at $Q^{(n+1)}/x = (n+1)!$

(b). Et spesielt tilfelle vi vil se på er når x_i 'ene er en uniform partisjon, det vil si at $x_i - x_{i-1} = h$ for alle i . Regn ut $\int_{x_0}^{x_2} Q_1(x) dx$ og $\int_{x_0}^{x_3} Q_2(x) dx$.

Hint: Delvis integrasjon kan være til hjelp her.

14. Skriv kode som lager plottet vist i Figur 10.1.

KAPITTEL 11

Numerisk derivasjon

Derivasjon er en grunnleggende operasjon i matematikk, med mange anvendelser. Derfor er det viktig å ha gode metoder for å beregne og bruke de deriverte. Du har sannsynligvis lært de grunnleggende reglene for derivasjon i skolen — symbolske metoder som passer for utregninger på papir. Disse er av begrenset verdi på datamaskiner siden mange programmeringsspråk ikke støtter symbolske beregninger.

Et annet punkt er at funksjonen vår ofte bare er kjent i et sett isolerte punkter, slik som målinger av posisjonen til en bil med en GPS-enhet. Anta at vi vil beregne bilens fart. Hvis posisjonen til bilen er kjent for alle tidspunkter (som en matematisk funksjon) så kan dette gjøres ved symbolsk derivasjon, men når posisjonen til bilen er bare kjent for enkelte tidspunkter så må vi ty til andre metoder.

Løsningen er numeriske metoder som tilnærmer den deriverte med passende kombinasjoner av kjente funksjonsverdier. La oss starte med å formulere akkurat dette som en generell strategi for numerisk derivasjon.

Strategi 11.1. Anta at f er kjent i punktene

$$\dots, a - 2h, a - h, a, a + h, a + 2h, \dots$$

(så, h er “oppløsningen” for kjente verdier av f).

1. Plukk en delmengde av $n + 1$ av disse punktene nær a , og skriv disse som

$$a + m_0h, a + m_1h, a + m_2h, \dots, a + m_nh, \quad (11.1)$$

der $m_0, \dots, m_n$ er heltallige.

2. Finn polynomet $p_n(x)$ som interpolerer f i de $n + 1$ punktene (11.1).
3. Bruk $p'_n(a)$ som en tilnærming til $f'(a)$.

Vi skal også kalle dette for *punktmetoden* basert på punktene (11.1). Punktmetoder kan brukes for høyere ordens deriverte også, ved å bruke $p_n^{(k)}(a)$ som en tilnærming til $f^{(k)}(a)$. Vi definerer *trunkeringsfeilen* i denne tilnærmingen som

$$E_k(f; a, h) = f^{(k)}(a) - p_n^{(k)}(a) \quad (11.2)$$

Dette er feilen som følger av at vi deriverer det interpolerende polynomet i stedet for funksjonen f . Målet er å få denne feilen så liten som mulig.

I det aller enkleste tilfellet baserer punktmetoder seg på kun to kjente funksjonsverdier, og vi skal i Seksjon 11.1 se hvordan dette naturlig henger sammen med definisjonen av den deriverte. I Seksjon 11.2 skal vi så se hvordan punktmetoder med flere punkter kan bli enda mer presise. Metodene for numerisk integrasjon i Kapittel 12 vil bli utledet og analysert på den samme måten som metodene for derivasjon i dette kapitlet.

I dette kapitlet vil vi skrive $M_k = \max_x |f^{(k)}(x)|$ (når f er k ganger deriverbar), der maksimum tas over et intervall som går klart fram fra konteksten. I notasjonen p_n og $E_k(f; a, h)$ vil det gå klart ut fra konteksten hva tallene m_i er, selv om de ikke er en del av notasjonen.

11.1 Newtonkvotienten

Den deriverte av f i et punkt a er definert som

$$f'(a) = \lim_{h \rightarrow 0} \frac{f(a+h) - f(a)}{h}. \quad (11.3)$$

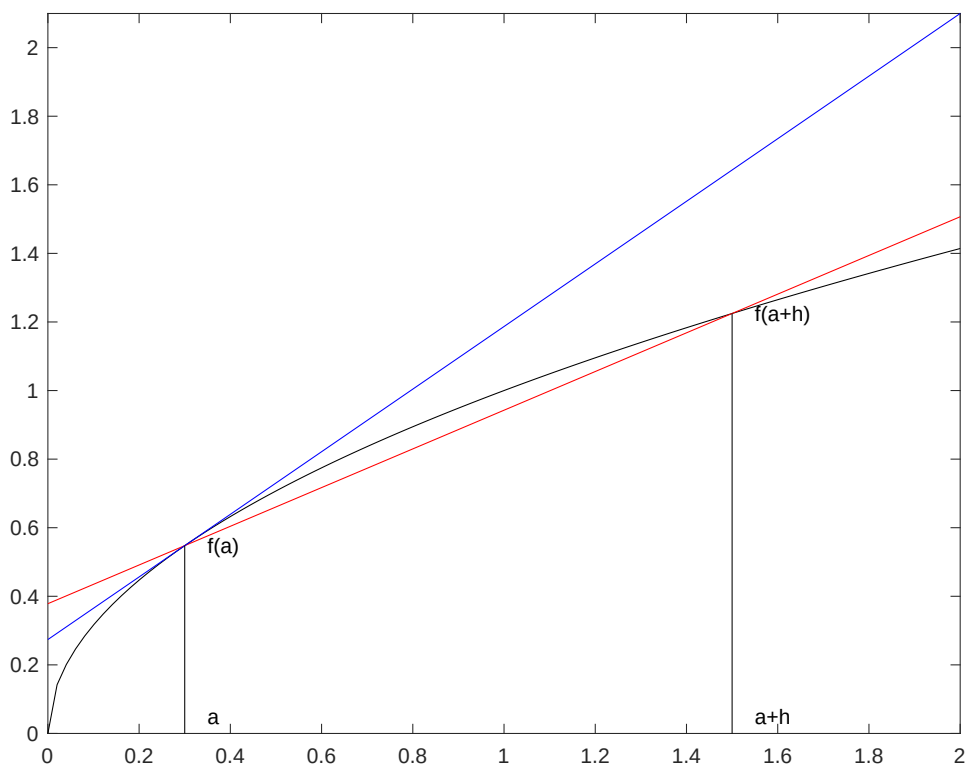
Vi vil anta at f er deriverbar i a , det vil si at denne grenseverdien eksisterer. For en fiksert $h \neq 0$ kalles tilnærmingen

$$\frac{f(a+h) - f(a)}{h} \quad (11.4)$$

til $f'(a)$ for *Newtons differenskvotient*, eller bare *newtonkvotienten*.

Bruker vi punktmetoden basert på punktene a og $a+h$ får vi den rette linjen p_1 gjennom $(a, f(a))$ og $(a+h, f(a+h))$, eller sekanten gjennom disse. Denne kan skrives

$$p_1(x) = f(a) + \frac{f(a+h) - f(a)}{h}(x-a).$$



Figur 11.1. Sekanten (rød kuve) til en funksjon (svart kurve) i punktene a og $a + h$, samt tangenten i a (blå kurve).

Den deriverte av p_1 , sekantens stigningstall, er nøyaktig høyresiden i (11.4). Newtonkvotienten faller altså sammen med punktmetoden basert på a og $a + h$ (se Figur 11.1).

Trunkeringsfeilen for newtonkvotienten er

$$E_1(f; a, h) = f'(a) - \frac{f(a+h) - f(a)}{h}.$$

La oss se nærmere på denne. Fra Taylors formel med restledd (10.1), får vi at

$$f(a+h) = f(a) + hf'(a) + \frac{h^2}{2}f''(\xi_h), \quad (11.5)$$

der $\xi_h \in [a, a+h]$. Stokker vi om på dette uttrykket får vi følgende uttrykk for trunkeringsfeilen:

$$f'(a) - \frac{f(a+h) - f(a)}{h} = -\frac{h}{2}f''(\xi_h). \quad (11.6)$$

Hvis den andrederiverte er kontinuerlig, så sier ekstremalverdisetningen at funksjonen $|f''|$ har et maksimum på intervallet $[a, a + h]$, som vi kaller M_2 . Fra dette får vi følgende feilestimat.

Proposisjon 11.2. Anta f har kontinuerlige andrederiverte nær a . Da oppfyller trunkeringsfeilen for netwonkvotienten

$$|E_1(f; a, h)| \leq M_2 \frac{h}{2}, \quad (11.7)$$

der $M_2 = \max_x |f''(x)|$.

Eksempel 11.3. La oss se på funksjonen $f(x) = \sin(x)$ i punktet $a = 0.5$. Siden den deriverte er $f'(x) = \cos(x)$, og $f'(a) \approx 0.8775825619$, er det lett å sjekke nøyaktigheten i Newtons kvotient. Prøver vi med noen verdier av h får vi

h	$(f(a+h) - f(a))/h$	$E_1(f; a, h)$
10^{-1}	0.8521693479	2.5×10^{-2}
10^{-2}	0.8751708279	2.4×10^{-3}
10^{-3}	0.8773427029	2.4×10^{-4}
10^{-4}	0.8775585892	2.4×10^{-5}
10^{-5}	0.8775801647	2.4×10^{-6}
10^{-6}	0.8775823222	2.4×10^{-7}

Vi ser at tilnærmingen blir bedre med avtagende h , som forventet. Mer presist ser vi at når h blir redusert med en faktor 10, så blir feilen redusert med samme faktor.

La oss sjekke at estimatet for feilen (11.7) stemmer overens med verdiene i tabellen. Vi har $f''(x) = -\sin(x)$, og siden $|\sin(x)| \leq 1$ for alle x , kan vi velge $M_2 = 1$. Dermed er høyre side i (11.7) lik $\frac{1}{2}h$. Tabellen tilsier at den egentlige feilen er $|E_1(f; a, h)| \approx 0.24h$, så estimatet $|E_1(f; a, h)| \leq 0.5h$ overestimerer feilen med en faktor 2. I slike feilestimer er det veldig vanlig å bomme litt, og "overdrive" hvor stor feilen kan bli. Det sentrale her er at vi har estimert noe ukjent (feilen E_1) med noe som er beregnbart (høyresiden i (11.7)).

Feilen e i våre numeriske metoder er stort sett på formen $e = Kh^k$, der h er presisjonen i metoden, og konstanten K og potensen k er ukjent. Vi sier da at y er *polynomiell* i x , med konstant K og potens k . *loglog plott* kan hjelpe oss med å se om det finnes slike sammenhenger mellom h og e , og i så fall også hjelp oss med å finne de ukjente konstantene K og k . Plotter vi h mot e så er det ikke så

lett å se hva K og k er ut fra plottet, siden kurven skyter rett til værs. Et triks kan være å ta logaritmer, noe som gir

$$\log e = \log K + k \log h.$$

Med andre ord, hvis vi plottet logaritmen til feilen mot logaritmen til h , så får vi en rett linje der

- stigningstallet til kurven gir oss potensen k , og
- skjæringen med y -aksen er i $\log K$, slik at vi kan finne K fra denne skjæringen.

Vi skriver `loglog(h, e)` hvis vi skal plote $\log h$ mot $\log e$. `loglog` følger omtrent samme syntaks som `plot`-kommandoen.

Figur 11.2 viser et `loglog`-plott av h mot trunkeringsfeilen, sammen med den øvre grensen for feilen $M_2 h/2$. Her satt vi $M_2 = 1$, slik at vi gjør den nevnte overestimeringen. Et bedre estimat ville vi fått ved å regne ut $\max_x |f''(x)|$ over et lite intervall i nærheten av a i stedet, men gjør du dette vil du se at de to kurvene blir liggende oppå hverandre. Plottet ser ut som en linje med derivert lik 1. Siden dette er et logaritmisk plott, er dette i overensstemmelse med tilnærmingen $|E_1(f; a, h)| \approx \frac{h}{2} |f''(a)|$ for trunkeringsfeilen, siden vi ved å ta logaritmer i denne får

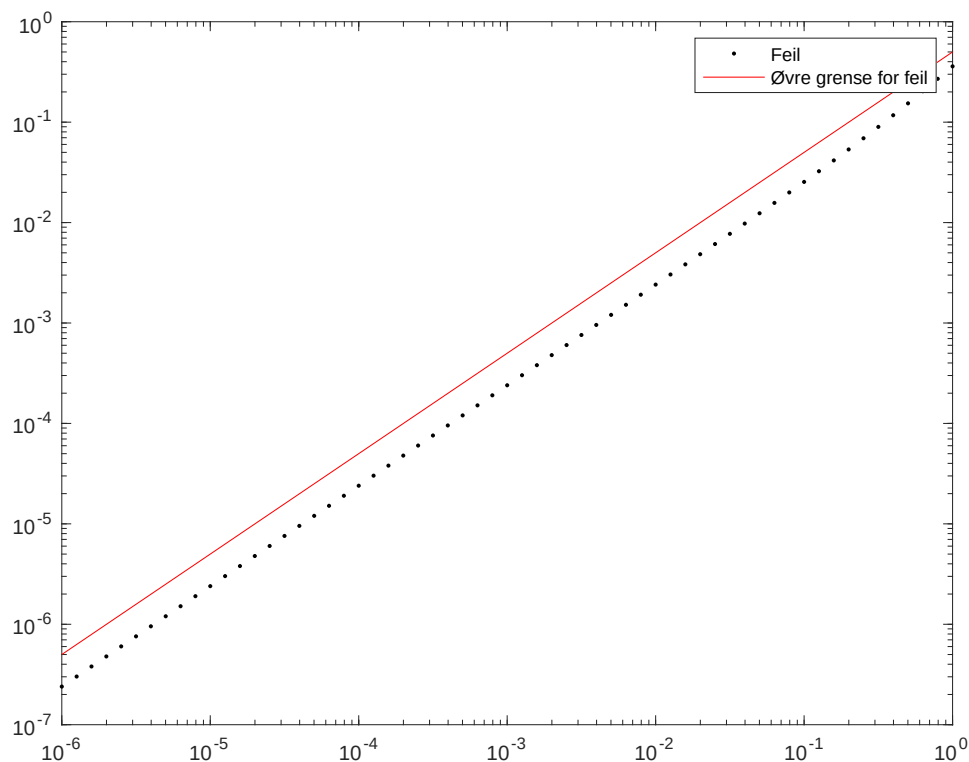
$$\begin{aligned} \log |E_1(f; a, h)| &\approx \log h + \log(0.5) + \log(\sin(0.5)) \\ &\approx \log h - 0.6203. \end{aligned}$$

Dette samsvarer bra med der linja treffer y -aksen.

11.2 Alternativer til newtonkvotienten

Vi startet dette kapitlet med newtonkvotienten, punktmetoden basert på $f(a)$ og $f(a+h)$. Hvis vi har flere punkter, byr det fort på mye regning å finne og derivere det interpolerende polynomet. Som et eksempel, se på punktmetoden basert på $f(a)$, $f(a-h)$, og $f(a+h)$. Bruker vi newtonformen til det interpolerende polynomet får vi fra (10.10) og (10.11) at

$$\begin{aligned} p_2(x) = f(a-h) &+ \frac{f(a) - f(a-h)}{h} (x - (a-h)) \\ &+ \frac{f(a+h) - 2f(a) + f(a-h)}{2h^2} (x - (a-h))(x - a). \end{aligned} \quad (11.8)$$



Figur 11.2. Trunkeringsfeilen ved bruk av newtonkvotienten for tilnærming til den deriverte til $f(x) = \sin x$ i punktet $x = 0.5$, for $h \in [10^{-6}, 10^0]$. Plottet er et logaritmisk plott.

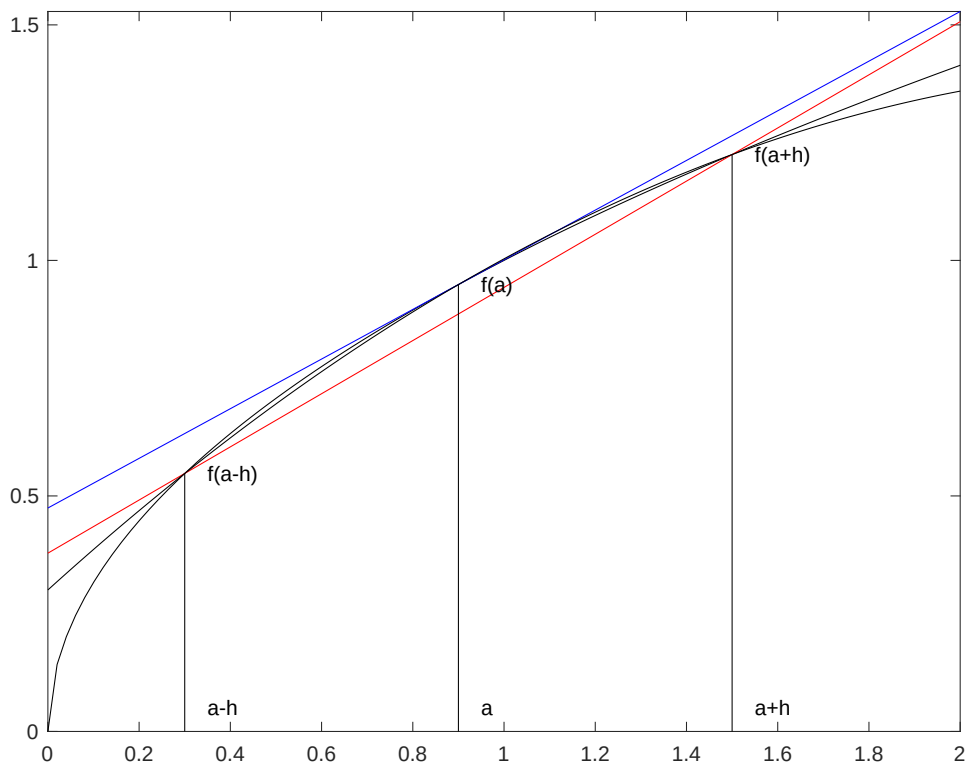
Etter litt regning finner vi at

$$p'_2(a) = \frac{f(a+h) - f(a-h)}{2h} \quad (11.9)$$

Detaljene i (11.8) og (11.9) overlater vi til Oppgave 6. Uttrykket (11.9) er stigningstallet til sekanten gjennom punktene $a-h$ og $a+h$, og kalles den *symmetriske newtonkvotienten*. Denne er illustrert i Figur 11.3,

Trunkeringsfeilen for den symmetriske Newtonkvotienten kan analyseres på samme måte som Newtonkvotienten. Vi evaluerer nå taylorpolynomet i to punkter i stedet, og tar med andregradsledd også:

$$\begin{aligned} f(a+h) &= f(a) + f'(a)h + \frac{1}{2}f''(a)h^2 + \frac{1}{6}f'''(c_1)h^3 \\ f(a-h) &= f(a) - f'(a)h + \frac{1}{2}f''(a)h^2 - \frac{1}{6}f'''(c_2)h^3, \end{aligned}$$



Figur 11.3. Sekanten i a (rød kurve) til en funksjon f (svart kurve) basert i $a-h$ og $a+h$, samt tangenten i a (blå kurve). Parabelen som interpolerer f i $a-h$, a , og $a+h$ er også vist (svart kurve)

der $c_1 \in [a, a+h]$, $c_2 \in [a-h, a]$. Trekkes disse fra hverandre får vi

$$\frac{f(a+h) - f(a-h)}{2h} - f'(a) = \frac{1}{12} h^2 (f'''(c_1) + f'''(c_2)).$$

Tar vi absoluttverdi av begge sider og bruker trekantulikheten får vi at

$$|E_1(f; a, h)| = \left| \frac{1}{12} h^2 (f'''(c_1) + f'''(c_2)) \right| \leq \frac{h^2}{12} M_3 + \frac{h^2}{12} M_3 = M_3 \frac{h^2}{6}.$$

Spesielt vil tilnærmingen være eksakt for andregradspolynomer, siden disse har tredjederivert lik 0 (slik at $M_3 = 0$). Dermed blir trunkeringsfeilen 0 for disse. Tilnærmingen er ikke eksakt for tredjegradspolynomer. Tester vi den symmetriske

Newtonkvotienten med samme verdier for h som før får vi

h	$(f(a+h) - f(a-h))/(2h)$	$E_1(f; a, h)$
10^{-1}	0.8761206554	1.5×10^{-3}
10^{-2}	0.8775679356	1.5×10^{-5}
10^{-3}	0.8775824156	1.5×10^{-7}
10^{-4}	0.8775825604	1.5×10^{-9}
10^{-5}	0.8775825619	1.8×10^{-11}
10^{-6}	0.8775825619	-7.5×10^{-12}

Når h reduseres med faktor 10, ser vi at feilen blir redusert med omtrent faktor 100. Tilsvarende faktor fra tabellen for newtonkvotienten var 10.

La oss se på et eksempel til, denne gang en punktmetode for den andrederiverte. Vi bruker samme punkter som den symmetriske Newtonkvotienten, slik at vi får samme interpolerende polynom som i (11.8). Etter litt mer regning finner vi at

$$p_2''(a) = \frac{f(a+h) - 2f(a) + f(a-h)}{h^2} \quad (11.10)$$

Som over kan vi finne trunkeringsfeilen ved å evaluere Taylorpolynomet i de samme punktene, men nå tar vi også med et tredjegradsledd,

$$\begin{aligned} f(a+h) &= f(a) + f'(a)h + \frac{1}{2}f''(a)h^2 + \frac{1}{6}f'''(a)h^3 + \frac{1}{24}f^{(iv)}(c_1)h^4 \\ f(a-h) &= f(a) - f'(a)h + \frac{1}{2}f''(a)h^2 - \frac{1}{6}f'''(a)h^3 + \frac{1}{24}f^{(iv)}(c_2)h^4, \end{aligned}$$

og i stedet for å trekke fra hverandre legger vi de nå sammen. Da får vi

$$f(a-h) - 2f(a) + f(a+h) - f''(a)h^2 = \frac{1}{24}f^{(iv)}(c_1)h^4 + \frac{1}{24}f^{(iv)}(c_2)h^4.$$

Deler vi med h^2 på begge sider får vi trunkeringsfeilen

$$|E_2f(; a, h)| = \left| \frac{f(a-h) - 2f(a) + f(a+h)}{h^2} - f''(a) \right|$$

på venstre side, og på høyre side får vi

$$M_4 \frac{h^2}{24} + M_4 \frac{h^2}{24} = M_4 \frac{h^2}{12}.$$

For punktmetoder med flere punkter kan man også gjøre som over - kombinere Taylorrekken i de kjente punktene slik at vi på venstre side får et uttrykk for trunkeringsfeilen. I Eksempel 11.6 skal vi utlede en punktmetode basert på

fire punkter. Kombinerer man Taylorrekkene i de kjente punktene for denne kan man vise at trunkeringsfeilen oppfyller $|E_1(f; a, h)| \leq \frac{M_5}{18} h^4$ (vi hopper over beviset). h^4 -faktoren gjør at trunkeringsfeilen går enda raskere mot 0 for firepunktsmetoden når h avtar.

Vi har nå estimater på trunkeringsfeilen for fire metoder

1. For Newtonkvotienten: $M_2 \frac{h}{2}$,
2. For den symmetriske Newtonkvotienten: $M_3 \frac{h^2}{6}$,
3. For firepunktsmetoden: $\frac{M_5}{18} h^4$,
4. For den andrederiverte: $M_4 \frac{h^2}{12}$.

Hvis vi bruker funksjonen $f(x) = \sin x$, samt at $M_k \approx (\sin a)^{(k)}$ nær a , så kan vi plote trunkeringsfeilestimatene mot hverandre som følger

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

a=0.5
h = 10**(-np.arange(0,6,0.1))
plt.loglog(h, np.abs(np.sin(a)*h/2), \
            h, np.abs(np.cos(a)*h**2/6), \
            h, np.abs(18*np.cos(a)*h**4), \
            h, np.abs(np.sin(a)*h**2/12))
plt.legend(['Newton', 'Symmetrisk Newton', 'Fire punkter', 'Andrederiverte'])
plt.show()
```

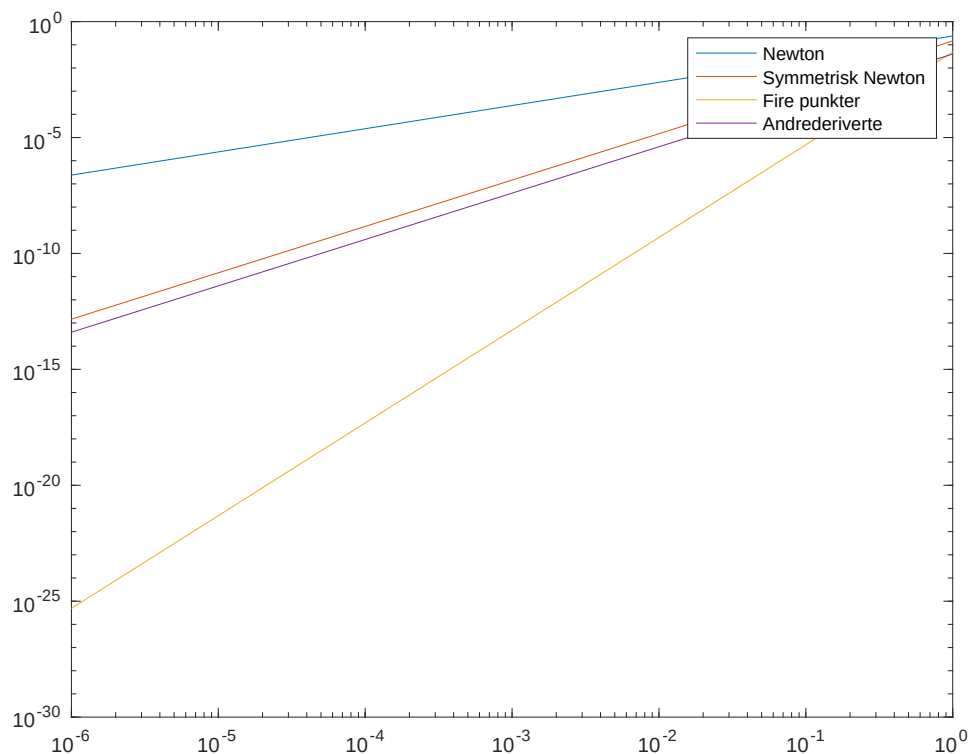
Listing 11.1. Sammenligning av trunkeringsfeil

Plottene er vist i Figur 11.4. Dette er et loglog-plott. Merk at, hvis $|E_1(f; a, h)| \approx Ch^p$, så har vi at

$$\log|E_1(f; a, h)| \approx p \log h + \log C.$$

Siden våre metoder har trunkeringsfeil proporsjonal med h , h^2 , og h^4 , så ser vi i loglog-plottet lineære kurver med stigningstall 1, 2, og 4.

Hvis vi baserer oss på enda flere punkter blir alt enda mer komplisert, og vi trenger derfor et resultat som gir oss system i kaoset. Det neste teoremet tar lineær algebra til hjelp, og sier hva punktmetoden basert på $f(a + m_0 h)$, $f(a + m_1 h), \dots, f(a + m_n h)$ blir.



Figur 11.4. loglog-plott av estimatene på trunkeringsfeilen for de fire metodene vi ser på, for $f(x) = \sin x$ i $x = 0.5$, og for $h \in [10^{-6}, 10^0]$.

Teorem 11.4. Anta at vi vet $n + 1$ funksjonsverdier $\{f(a + m_i h)\}_{i=0}^n$, der $m_0 = 0$ og $m_0, m_1, \dots, m_n$ er forskjellige heltall (vi antar altså spesielt at $f(a)$ skal bidra). La p_n være det interpolerende polynomet til f av grad $\leq n$ i punktene $\{a + m_i h\}_{i=0}^n$. Da har vi at

$$\begin{pmatrix} h^1 p_n^{(1)}(a)/1! \\ h^2 p_n^{(2)}(a)/2! \\ \vdots \\ h^n p_n^{(n)}(a)/n! \end{pmatrix} = \begin{pmatrix} m_1 & m_1^2 & \cdots & m_1^n \\ m_2 & m_2^2 & \cdots & m_2^n \\ \vdots & \vdots & \ddots & \vdots \\ m_n & m_n^2 & \cdots & m_n^n \end{pmatrix}^{-1} \begin{pmatrix} f(a + m_1 h) - f(a) \\ f(a + m_2 h) - f(a) \\ \vdots \\ f(a + m_n h) - f(a) \end{pmatrix}. \quad (11.11)$$

Det er ikke meningen at du skal huske dette resultatet! Poenget er at vi fra dette faktisk kan lese ut, og lage kode for å finne, tilnærmingene $p_n^{(k)}(a)$ til alle de deriverte! Tilnærmingene får vi ved å løse et likningssystem som involverer

en vandermondematrise.

Bevis. p_n er lik sin egen Taylorrekke av grad n om a , det vil si at

$$p_n(x) = p_n(a) + p'_n(a)(x-a) + \cdots + \frac{p_n^{(n)}(a)}{n!}(x-a)^n.$$

Evaluerer vi dette i $x = a + m_i h$ får vi at

$$f(a + m_i h) = f(a) + p'_n(a)m_i h + \cdots + \frac{p_n^{(n)}(a)}{n!} m_i^n h^n,$$

som kan skrives

$$f(a + m_i h) - f(a) = \begin{pmatrix} m_i & m_i^2 & \cdots & m_i^n \end{pmatrix} \begin{pmatrix} h p'_n(a) \\ h^2 p_n^{(2)}(a)/2! \\ \vdots \\ h^n p_n^{(n)}(a)/n! \end{pmatrix}$$

for $i = 1, 2, \dots, n$ setter vi dette sammen til

$$\begin{pmatrix} m_1 & m_1^2 & \cdots & m_1^n \\ m_2 & m_2^2 & \cdots & m_2^n \\ \vdots & \vdots & \ddots & \vdots \\ m_n & m_n h^2 & \cdots & m_n^n \end{pmatrix} \begin{pmatrix} h^1 p_n^{(1)}(a)/1! \\ h^2 p_n^{(2)}(a)/2! \\ \vdots \\ h^n p_n^{(n)}(a)/n! \end{pmatrix} = \begin{pmatrix} f(a + m_1 h) - f(a) \\ f(a + m_2 h) - f(a) \\ \vdots \\ f(a + m_n h) - f(a) \end{pmatrix},$$

Matrisen på venstre side er en inverterbar vandermondematrise. Ganger vi med den inverse på begge sider får vi uttrykket vi skulle fram til. ■

La oss se på et par eksempler.

Eksempel 11.5. La oss se på punktmetoder som baserer seg på de tre punktene a , $a - h$, og $a + h$. Dette svarer til at vi setter $n = 2$, $m_1 = -1$, $m_2 = 1$. (11.11) gir

$$\begin{aligned} \begin{pmatrix} h p'_2(a) \\ h^2 p''_2(a)/2 \end{pmatrix} &= \begin{pmatrix} m_1 & m_1^2 \\ m_2 & m_2^2 \end{pmatrix}^{-1} \begin{pmatrix} f(a-h) - f(a) \\ f(a+h) - f(a) \end{pmatrix} = \begin{pmatrix} -1 & 1 \\ 1 & 1 \end{pmatrix}^{-1} \begin{pmatrix} f(a-h) - f(a) \\ f(a+h) - f(a) \end{pmatrix} \\ &= \frac{1}{2} \begin{pmatrix} -1 & 1 \\ 1 & 1 \end{pmatrix} \begin{pmatrix} f(a-h) - f(a) \\ f(a+h) - f(a) \end{pmatrix} = \begin{pmatrix} \frac{f(a+h) - f(a-h)}{2} \\ \frac{f(a-h) - 2f(a) + f(a+h)}{2} \end{pmatrix}. \end{aligned}$$

Dette kan vi skrive om til

$$\begin{aligned} p'_2(a) &= \frac{f(a+h) - f(a-h)}{2h} \\ p''_2(a) &= \frac{f(a-h) - 2f(a) + f(a+h)}{h^2} \end{aligned}$$

Vi ser at dette faller sammen med punktmetodene fra (11.9) og (11.10).

Eksempel 11.6. Vi ser på punktmetoder som baserer seg på de fem punktene a , $a - 2h$, $a - h$, $a + h$, og $a + 2h$. Vi setter $n = 4$, $m_1 = -2$, $m_2 = -1$, $m_3 = 1$, $m_4 = 2$. (11.11) gir

$$\begin{pmatrix} h^1 p_4^{(1)}(a)/1! \\ h^2 p_4^{(2)}(a)/2! \\ h^3 p_4^{(3)}(a)/3! \\ h^4 p_4^{(4)}(a)/4! \end{pmatrix} = \begin{pmatrix} m_1 & m_1^2 & m_1^3 & m_1^4 \\ m_2 & m_2^2 & m_2^3 & m_2^4 \\ m_3 & m_3^2 & m_3^3 & m_3^4 \\ m_4 & m_4^2 & m_4^3 & m_4^4 \end{pmatrix}^{-1} \begin{pmatrix} f(a-2h) - f(a) \\ f(a-h) - f(a) \\ f(a+h) - f(a) \\ f(a+2h) - f(a) \end{pmatrix}$$

$$= \begin{pmatrix} -2 & 4 & -8 & 16 \\ -1 & 1 & -1 & 1 \\ 1 & 1 & 1 & 1 \\ 2 & 4 & 8 & 16 \end{pmatrix}^{-1} \begin{pmatrix} f(a-2h) - f(a) \\ f(a-h) - f(a) \\ f(a+h) - f(a) \\ f(a+2h) - f(a) \end{pmatrix}$$

Det er lurt å regne inversmatrisen ut symbolsk her:

```
>> V = sym.Matrix([[-2,4,-8,16], [-1,1,-1,1], [1,1,1,1], [2,4,8,16]])
>> V.inv()
Matrix([[1/12, -2/3, 2/3, -1/12],
        [-1/24, 2/3, 2/3, -1/24],
        [-1/12, 1/6, -1/6, 1/12],
        [1/24, -1/6, -1/6, 1/24]])
```

Uttrykket over blir dermed

$$\begin{pmatrix} 1/12 & -2/3 & 2/3 & -1/12 \\ -1/24 & 2/3 & 2/3 & -1/24 \\ -1/12 & 1/6 & -1/6 & 1/12 \\ 1/24 & -1/6 & -1/6 & 1/24 \end{pmatrix} \begin{pmatrix} f(a-2h) - f(a) \\ f(a-h) - f(a) \\ f(a+h) - f(a) \\ f(a+2h) - f(a) \end{pmatrix}$$

Evaluerer vi første rad får vi

$$\begin{aligned} h p_4'(a) &= \begin{pmatrix} 1/12 & -2/3 & 2/3 & -1/12 \end{pmatrix} \begin{pmatrix} f(a-2h) - f(a) \\ f(a-h) - f(a) \\ f(a+h) - f(a) \\ f(a+2h) - f(a) \end{pmatrix} \\ &= \frac{f(a-2h)}{12} - \frac{2f(a-h)}{3} + \frac{2f(a+h)}{3} - \frac{f(a+2h)}{12} - \left(\frac{1}{12} - \frac{2}{3} + \frac{2}{3} - \frac{1}{12} \right) f(a) \\ &= \frac{f(a-2h) - 8f(a-h) + 8f(a+h) - f(a+2h)}{12}. \end{aligned}$$

Deler vi med h får vi at

$$p_4'(a) = \frac{f(a-2h) - 8f(a-h) + 8f(a+h) - f(a+2h)}{12h}.$$

Oppgaver for Kapittel 11

1. Et alternativ til newtonkvotienten (11.4) er den venstresidige varianten

$$f'(a) \approx \frac{f(a) - f(a-h)}{h}.$$

Vis at vi denne faller sammen med punktmetoden basert på a og $a-h$.

2. Skriv et program som regner ut newtonkvotienten for funksjonen $f(x) = e^x$ i $a = 1$, og bruker $h = 10^{-k}$ for $k = 1, 2, \dots, 14$. Regn også ut feilen. Oppfør tilnærmingen og dens feil slik som Proposisjon 11.2 sier?

3. Da vi utledet trunkeringsfeilen (11.7), var er det ikke opplagt hva graden til taylorpolynomet i (11.5) skulle være. I denne oppgaven skal du se hva som skjer om man senker graden på taylorpolynomet.

(a). Gjør taylorutviklingen i (11.5) på nytt, men bruk et taylorpolynom av grad 2 i stedet. Prøv fra dette å utlede en formel for feilen som er tilsvarende (11.6).

(b). Gjør (a) på nytt, men bruk et taylorpolynom av grad 0, altså bare en konstant funksjon.

(c). Hvorfor kan du konkludere at det førsteordens taylorpolynom og formelen for feilen i (11.6) er best?

4. Vi tilnærmer den andrederiverte til en funksjon $f(x)$ i $x = 0$ med

$$D_2 f(0) = \frac{f(h) - 2f(0) + f(-h)}{h^2}$$

Vi antar at f er deriverbar uendelig mange ganger. Finn det korrekte alternativet i følgende flervalgsoppgaver.

- (a). Tilnærmingen $D_2 f(0)$ vil alltid være eksakt hvis $f(x)$ er

- ☐ en trigonometrisk funksjon
- ☐ en logaritmisk funksjon
- ☐ et polynom av grad 4
- ☐ et polynom av grad 3

(b). Feilen

$$|f''(0) - D_2 f(0)|$$

er begrenset av

- ☐ $\frac{h^2}{12} \max_{x \in [-h, h]} |f''(x)|$
- ☐ $\frac{h^2}{48} \max_{x \in [-h, h]} |f^{(4)}(x)|$
- ☐ $\frac{h}{4} \max_{x \in [-h, h]} |f''(x)|$
- ☐ $\frac{h^2}{12} \max_{x \in [-h, h]} |f^{(4)}(x)|$

5. Marker hver av følgende påstander som sanne eller gale

(a). Den symmetriske newtonkvotienten er eksakt for polynomer av grad 2 eller lavere.

(b). Selv om den symmetriske newtonkvotienten gir større presisjon, så er det en trade-off siden den krever mye mer regning enn newtonkvotienten.

(c). firepunktsmetoden med steglengde $h = 0.2$ vil vanligvis ha mindre feil enn den symmetriske newtonkvotienten med $h = 0.1$.

(d). firepunktsmetoden er eksakt for alle fjerdegradspolynomer, men ikke for alle femtegradspolynomer.

6. Vis at formlene (11.8) (11.9) er riktige.

7. Vis at den symmetriske newtonkvotienten i $x = a$

(a). er lik $f'(a)$ når f er en kvadratisk funksjon $f(x) = bx^2 + cx + d$.

(b). er lik gjennomsnittet av de to asymmetriske kvotientene

$$f_r(a) = \frac{f(a+h) - f(a)}{h} \quad f_l(a) = \frac{f(a) - f(a-h)}{h}. \quad (11.12)$$

8. I denne oppgaven ser vi på noen andre mulige tilnærminger til den deriverte.

(a). Finn en tilnærming til $f'(a)$ som bare bruker verdiene $f(a-h)$, $f(a)$, og $f(a+2h)$.

(b). Finn en tilnærming til $f'(a)$ som bare bruker verdiene $f(a)$, $f(a+h)$, og $f(a+2h)$.

9. Denne oppgaven illustrerer en annen måte som kan brukes til å utlede metoder for numerisk derivasjon.

(a). Anta vi vil utlede en metode for å tilnærme $f'(a)$ som er på formen

$$f'(a) \approx c_1 f(a-h) + c_2 f(a+h), \quad c_1, c_2 \in \mathbb{R}.$$

Vi vil at metoden skal være eksakt når $f(x) = 1$ og $f(x) = x$. Bruk disse betingelsene til å regne ut c_1 og c_2 .

(b). Vis at metoden i (a) er eksakt for alle polynomer av grad 1, og sammenlign den med metodene vi har utledet i dette kapitlet.

(c). Bruk metodene i (a) og (b) til å utlede en metode for å tilnærme $f''(a)$.

$$f''(a) \approx c_1 f(a-h) + c_2 f(a) + c_3 f(a+h), \quad c_1, c_2, c_3 \in \mathbb{R},$$

ved å kreve at metodene skal være eksakt for $f(x) = 1$, x , og x^2 . Kjenner du igjen metoden?

(d). Vis at metoden i (c) er eksakt for alle kubiske polynomer.

10. I denne oppgaven skal vi sammenligne den symmetriske newtonkvotienten med de to asymmetriske newtonkvotientene.

(a). Tegn grafen til funksjonen

$$f(x) = \frac{-x^2 + 10x - 5}{4}$$

over intervallet $[0, 6]$, sammen med de tre sekantene som hører til de tre tilnærmingene til de deriverte fra (a) (bruk $a = 3$ og $h = 2$). Kan du fra dette avgjøre hvilken tilnærming som er best?

(b). Regn ut de tre tilnærmingene til de deriverte i (a) numerisk for $f(x)$ med $a = 3$ og $h_1 = 0.1$ og $h_2 = 0.001$. Hva er de relative feilene?

KAPITTEL 12

Numerisk integrasjon

I numerisk derivasjon brukes kjente funksjonsverdier til å finne tilnærminger til den deriverte av en funksjon. I numerisk *integrasjon* brukes samme informasjon til å finne tilnærminger til *integralet*. Metodene er nyttige når funksjonsverdiene bare er kjent i isolerte punkter, eller når det er vanskelig eller umulig å beregne integralet for hånd.

I Kapittel 11 laget vi en generell strategi for å utlede metoder for numerisk derivasjon. Den tilsvarende strategien for numerisk integrasjon er som følger: Finn polynomet som interpolerer funksjonen i et passende sett av punkter, og bruk integralet av dette polynomet som en tilnærming til integralet av funksjonen. Trunkeringsfeilen kan da analyseres på omtrent samme måte som for numerisk derivasjon.

Vi starter dette kapitlet med å gå gjennom definisjonen av integralet (Seksjon 12.1). Deretter ser vi på den enkleste metoden for numerisk integrasjon, midtpunktsmetoden (Seksjon 12.2). I Seksjon 12.3 formulerer vi så den ovennevnte generelle strategien i litt mer detalj, og ser på noen viktige spesialtilfeller.

12.1 Hva er integralet?

Integralet av en funksjon $f: [a, b] \rightarrow \mathbb{R}$ tolker vi som arealet under grafen, der de positive verdiene til f bidrar positivt til arealet, og de negative verdiene bidrar negativt (se Figur 12.1). Standard notasjon for integralet er

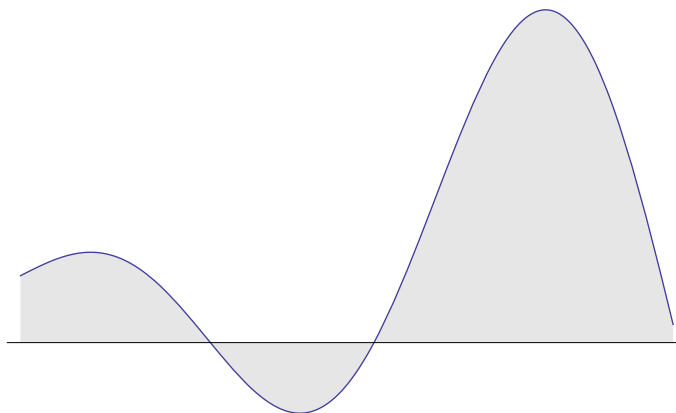
$$\int_a^b f(x) dx.$$

Før vi fortsetter må vi introdusere et begrep som vi skal bruke flere ganger i vår definisjon av integralet.

Definisjon 12.1 (Partisjon). La a og b være to reelle tall med $a < b$. En partisjon av $[a, b]$ er en samling av tall $\{x_0, x_1, \dots, x_n\}$ slik at

$$a = x_0 < x_1 < x_2 < \dots < x_{n-1} < x_n = b.$$

Partisjonen sies å være uniform dersom $x_i - x_{i-1} = (b - a)/n$ for $i = 1, \dots, n$. Avstanden mellom punktene, $h = (b - a)/n$, kalles da steglengden til partisjonen.



Figur 12.1. Arealet under grafen til en funksjon.

Om $\Pi = \{x_0, x_1, \dots, x_n\}$ er en partisjon, lar vi *størrelsen* på partisjonen være $|\Pi| = \max_{i=1, \dots, n} |x_i - x_{i-1}|$, altså den største avstanden mellom nabopunkter. Hvis Π er uniform vil $|\Pi| = (b - a)/n$.

12.1.1 Definisjonen av integralet

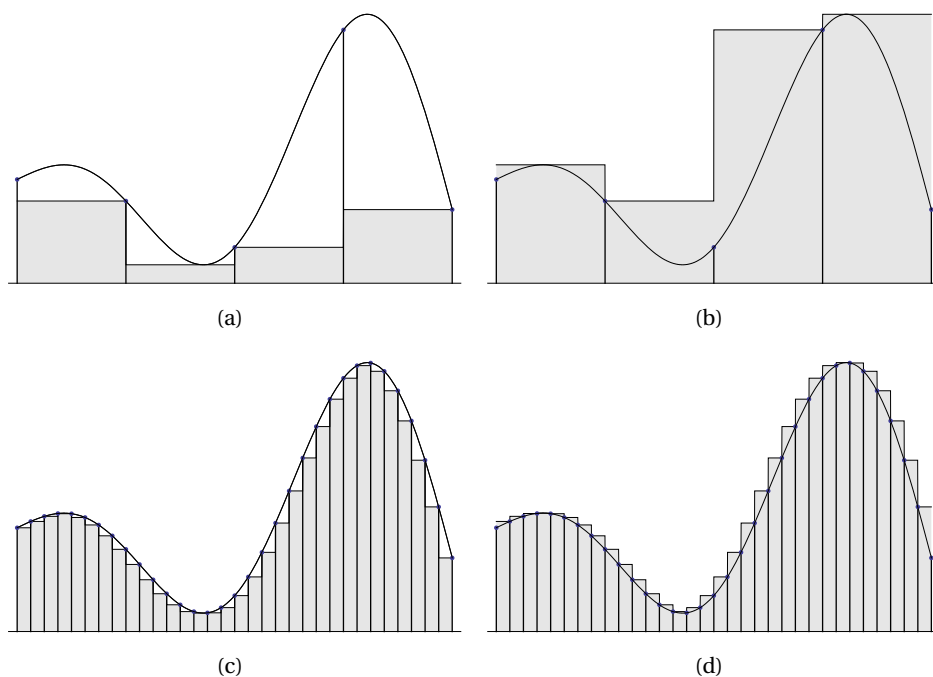
For å definere integralet bruker man såkalte *Riemann-summer*, som er numeriske tilnærminger av arealet under grafen. Vi velger en partisjon $\Pi = \{x_0, x_1, \dots, x_n\}$ av $[a, b]$, og finner maksimum og minimum til f på hvert delintervall $[x_{i-1}, x_i]$,

$$m_i = \min_{x \in [x_{i-1}, x_i]} f(x), \quad M_i = \max_{x \in [x_{i-1}, x_i]} f(x) \quad \text{for } i = 1, 2, \dots, n.$$

Dersom f er kontinuerlig, vil disse verdiene alltid eksistere, så vi antar heretter kontinuitet. Vi kan så gjøre to forskjellige tilnærminger til integralet ved å tilnærme f med to forskjellige *stegfunksjoner*, som begge er konstant på hvert delintervall $[x_{i-1}, x_i]$: Den første er lik m_i på dette delintervallet, den andre er lik M_i (se Figur 12.2). Deretter summerer vi opp arealene under de to stegfunksjonene, og ender opp med de to tilnærmingerne

$$\underline{I}(\Pi) = \sum_{i=1}^n m_i(x_i - x_{i-1}), \quad \bar{I}(\Pi) = \sum_{i=1}^n M_i(x_i - x_{i-1}). \quad (12.1)$$

Generelt er den første tilnærmingen for liten, den andre for stor. Integralet får vi nå av å la størrelsen på partisjonen gå mot 0:



Figur 12.2. Innskrevne (til venstre) og omskrevne (til høyre) stegfunksjoner for en grov partisjon (øverst) og en fin partisjon (nederst).

Definisjon 12.2 (Integralet). La f være en kontinuerlig funksjon definert på intervallet $[a, b]$. Hvis grenseverdiene

$$\lim_{|\Pi| \rightarrow 0} I(\Pi) \quad \text{og} \quad \lim_{|\Pi| \rightarrow 0} \bar{I}(\Pi)$$

(der grensen tas over følger av partisjoner med synkende størrelse) begge eksisterer og er like, sier vi at f er integrerbar, og vi definerer integralet av f over $[a, b]$ som

$$I = \int_a^b f(x) dx = \lim_{|\Pi| \rightarrow 0} I(\Pi) = \lim_{|\Pi| \rightarrow 0} \bar{I}(\Pi).$$

Det viser seg at *enhver* kontinuerlig funksjon på et lukket intervall er integrerbar. Et bevis for dette finner du i *Kalkulus*. Konstruksjonen av integralet illustreres i Figur 12.2, der vi ser hvordan de stykkevis konstante tilnærmingerne blir bedre når partisjonen blir mindre.

Definisjonen over kan brukes som en numerisk metode for å finne tilnær-

mingler til integralet: Vi velger å jobbe med enten maksimum eller minimum, velger en partisjon av $[a, b]$ som i Figur 12.2, og legger sammen arealene av rektanglene. Problemet med denne metoden er at det kan både være vanskelig og tidkrevende å finne maksimum og minimum, selv på en datamaskin. Men det viser seg at integralet har en egenskap som er veldig nyttig når det kommer til numeriske beregninger:

Teorem 12.3. *La $f: [a, b] \rightarrow \mathbb{R}$ være kontinuert, la $\{x_0, \dots, x_n\}$ være en partisjon av $[a, b]$, og la t_i være et tall i $[x_{i-1}, x_i]$ for hver $i = 1, \dots, n$. Da vil summen*

$$I = \sum_{i=1}^n f(t_i)(x_i - x_{i-1}) \quad (12.2)$$

konvergere mot integralet når størrelsen på partisjonene går mot null.

Bevis. Siden f er kontinuert, er den integrerbar, så tilnærmingene $\underline{I}$ og $\bar{I}$ konvergerer begge mot $\int_a^b f(x) dx$ når størrelsen på partisjonene går mot null. For enhver partisjon vil $\underline{I} \leq I \leq \bar{I}$, fordi $\min_{x \in [x_{i-1}, x_i]} f(x) \leq f(t_i) \leq \max_{x \in [x_{i-1}, x_i]} f(x)$ for hver $i = 1, \dots, n$. Dermed vil også I konvergere mot $\int_a^b f(x) dx$. ■

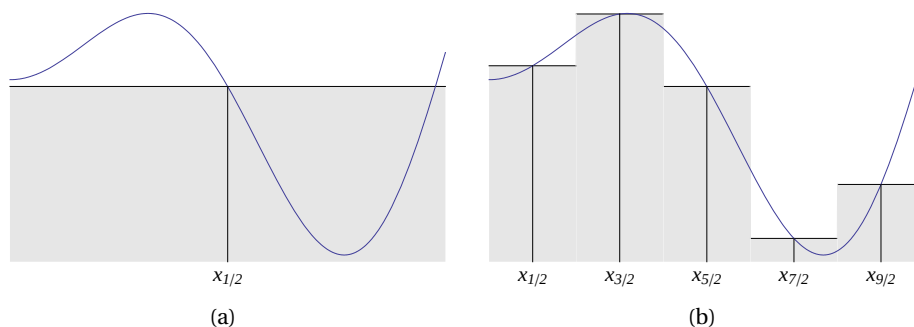
Generelt kan vi utlede metoder for numerisk integrasjon ved å dele opp intervallet $[a, b]$ i små delintervaller, tilnærme f med et polynom på hvert delintervall, integrere dette polynomet i stedet for f , og til slutt legge sammen bidragene fra hvert delintervall. Vi skal bruke denne strategien til å utlede mer avanserte metoder for numerisk integrasjon, og dette vil fungere så lenge f kan tilnærmes godt med polynomer på hvert delintervall.

12.2 Midtpunktsmetoden

Ved hjelp av Teorem 12.3 kan vi utlede praktiske, numeriske metoder for å regne ut integralet: Vi lager en partisjon av $[a, b]$, velger t_i lik x_{i-1} eller x_i , og regner ut summen (12.2). Det viser seg at et enda bedre valg er midtpunktet $t_i = (x_i + x_{i-1})/2$, som gir oss tilnærmingen

$$I \approx \sum_{i=1}^n f((x_i + x_{i-1})/2)(x_i - x_{i-1}). \quad (12.3)$$

Dette er den såkalte *midtpunktsmetoden*. Vi kan tenke på dette som en konstant tilnærming på hvert delintervall, der konstanten er verdien i midtpunktet.



Figur 12.3. Midtpunktsmetoden med ett delintervall **(a)**, og fem delintervaller **(b)**.

Algoritme 12.4. La f være en funksjon som er integrerbar på intervallet $[a, b]$, og la $\{x_i\}_{i=0}^n$ være en uniform partisjon av $[a, b]$. Midtpunktsmetoden tilnærmer integralet av f med

$$\int_a^b f(x) dx \approx I_{mid}(h) = h \sum_{i=1}^n f(x_{i-1/2}), \quad (12.4)$$

der

$$x_{i-1/2} = (x_{i-1} + x_i)/2 = a + (i - 1/2)h.$$

Metoden er illustrert i Figur 12.3 for tilfellene med ett og fem delintervaller.

Eksempel 12.5. La oss teste midtpunktsmetoden på en funksjon der vi vet svaret. Vi velger integralet

$$\int_0^1 \cos x dx = \sin 1 \approx 0.8414709848$$

Det eksakte svaret er da $\sin 1$. Vi tester metoden ved å splitte intervallet i 2^k delintervaller, for $k = 1, 2, \dots, 10$, d.v.s. vi halverer steglengden i hver iterasjon. Vi

får da

h	$I_{\text{mid}}(h)$	$I - I_{\text{mid}}(h)$
0.500000	0.85030065	-8.8×10^{-3}
0.250000	0.84366632	-2.2×10^{-3}
0.125000	0.84201907	-5.5×10^{-4}
0.062500	0.84160796	-1.4×10^{-4}
0.031250	0.84150523	-3.4×10^{-5}
0.015625	0.84147954	-8.6×10^{-6}
0.007813	0.84147312	-2.1×10^{-6}
0.003906	0.84147152	-5.3×10^{-7}
0.001953	0.84147112	-1.3×10^{-7}
0.000977	0.84147102	-3.3×10^{-8}

Vi ser at hver gang h is halveres, så blir feilen $|I - I_{\text{mid}}(h)|$ redusert omtrent med faktor 4.

12.2.1 Feilestimat for midtpunktsmetoden

Midtpunktsmetoden (12.4) er en tilnærming, og vil medføre en liten feil. Som i kapitlet om numerisk derivasjon skal vi nå *estimere* hvor stor denne feilen kan bli. Dette gjør vi i to steg. Først skal vi analysere feilen i tilfellet med bare ett delintervall. Dette kaller vi den *lokale feilen*. Deretter bruker vi dette til å finne et estimat for det generelle tilfellet med mange delintervaller. Dette kaller vi den *globale feilen*.

Lokal feilanalyse

Feilen i midtpunktsmetoden med bare ett delintervall er

$$E = \int_a^b f(x) dx - f(m)(b-a), \quad (12.5)$$

der $m = (a+b)/2$. Taylorutvikler vi $f(x)$ om midtpunktet m får vi

$$f(x) = f(m) + (x-m)f'(m) + \frac{(x-m)^2}{2}f''(\xi),$$

der ξ er et tall mellom m og x , som avhenger av x . Integrerer vi dette får vi

$$\begin{aligned} \int_a^b f(x) dx &= \int_a^b \left(f(m) + (x-m)f'(m) + \frac{(x-m)^2}{2}f''(\xi) \right) dx \\ &= f(m)(b-a) + \frac{f'(m)}{2} [(x-m)^2]_a^b + \frac{1}{2} \int_a^b (x-m)^2 f''(\xi) dx \\ &= f(m)(b-a) + \frac{1}{2} \int_a^b (x-m)^2 f''(\xi) dx, \end{aligned}$$

der leddet i midten var 0. Dette leder oss til uttrykket

$$|E| = \left| \int_a^b f(x) dx - f(m)(b-a) \right| = \frac{1}{2} \left| \int_a^b (x-m)^2 f''(\xi) dx \right|.$$

Litt mer regning på dette gir

$$\begin{aligned} |E| &= \frac{1}{2} \left| \int_a^b (x-m)^2 f''(\xi) dx \right| \leq \frac{1}{2} \int_a^b |(x-m)^2 f''(\xi)| dx \\ &= \frac{1}{2} \int_a^b (x-m)^2 |f''(\xi)| dx \\ &\leq \frac{M_2}{2} \int_a^b (x-m)^2 dx \\ &= \frac{M_2}{2} \frac{1}{3} [(x-m)^3]_a^b \\ &= \frac{M_2}{6} ((b-m)^3 - (a-m)^3) \\ &= \frac{M_2}{24} (b-a)^3, \end{aligned}$$

der $M_2 = \max_{x \in [a,b]} |f''(x)|$. Ulikheten på første linje holder fordi når vi flytter absoluttverdien innenfor integralet, så blir funksjonen vi integrerer ikke negativ overalt. Dette betyr at arealer der integranden i det opprinnelige uttrykket var negativ, nå er blitt positive, slik at det andre integralet er større enn det første.

Likheten på andre linje holder siden $(x-m)^2$ aldri er negativ, og ulikheten på tredje linje holder siden vi erstatter $|f''(\xi)|$ med sitt maksimum på intervallet $[a, b]$. På fjerde linje regner vi bare ut integralet av $(x-m)^2$. Så setter vi inn grensene. Siste linje følger av at $(b-m)^3 - (a-m)^3 = (b-a)^3/8$. Vi har dermed bevist følgende lemma.

Lemma 12.6. *La $f: [a, b] \rightarrow \mathbb{R}$ være en kontinuerlig funksjon med kontinuerlige første- og andreordens deriverte. Feilen i midtpunktsmetoden med ett delintervall er da begrenset av*

$$\left| \int_a^b f(x) dx - f(m)(b-a) \right| \leq \frac{M_2}{24} (b-a)^3,$$

der $m = (a+b)/2$.

Det viktige i Lemma 12.6 er faktoren $(b-a)^3$. Denne medfører at dersom vi halverer bredden på intervallene, så vil feilen i metoden reduseres med faktor 8.

Global feilanalyse

I forrige seksjon så vi på feilen for étt delintervall. La oss nå se på hva som skjer når vi legger sammen bidragene fra mange delintervaller. Generelt har vi en partisjon av $[a, b]$ i n delintervaller, alle med lengde $h = (b - a)/n$. Med

$$I = \int_a^b f(x) dx = \sum_{i=1}^n \int_{x_{i-1}}^{x_i} f(x) dx, \quad I_{\text{mid}}(h) = \sum_{i=1}^n f(x_{i-1/2})h,$$

blir den totale feilen i midtpunktsmetoden

$$I - I_{\text{mid}}(h) = \sum_{i=1}^n \left(\int_{x_{i-1}}^{x_i} f(x) dx - f(x_{i-1/2})h \right),$$

der uttrykket inne i parentesene er den lokale feilen på intervallet $[x_{i-1}, x_i]$. Vi har derfor

$$\begin{aligned} |I - I_{\text{mid}}(h)| &= \left| \sum_{i=1}^n \left(\int_{x_{i-1}}^{x_i} f(x) dx - f(x_{i-1/2})h \right) \right| \\ &\leq \sum_{i=1}^n \left| \int_{x_{i-1}}^{x_i} f(x) dx - f(x_{i-1/2})h \right| \\ &\leq \sum_{i=1}^n \frac{h^3}{24} M_2^{(i)} \end{aligned}$$

der $M_2^{(i)} = \max_{x \in [x_{i-1}, x_i]} |f''(x)|$. I den første ulikheten har vi brukt trekantulikheten, mens vi i den andre ulikheten har brukt Lemma 12.6. Vi kan forenkle (12.6) ytterligere ved å bruke maksimum over hele $[a, b]$, i stedet for over bare $[x_{i-1}, x_i]$: For alle i har vi at

$$M_2^{(i)} = \max_{x \in [x_{i-1}, x_i]} |f''(x)| \leq \max_{x \in [a, b]} |f''(x)| = M_2,$$

slik at (12.6) kan forenkles til

$$|I - I_{\text{mid}}(h)| \leq \sum_{i=1}^n \frac{h^3}{24} M_2^{(i)} \leq \sum_{i=1}^n \frac{h^3}{24} M_2 = \frac{h^3}{24} n M_2. \quad (12.6)$$

Siden $h = (b-a)/n$, får vi $hn = b-a$. Hvis vi setter dette inn i (12.6), får vi følgende estimat for den globale feilen.

Teorem 12.7. Anta at $f: [a, b] \rightarrow \mathbb{R}$ og dens første to deriverte er kontinuerlige. Da er feilen i midtpunktsmetoden begrenset av

$$|I - I_{mid}(h)| \leq (b - a) \frac{h^2}{24} M_2. \quad (12.7)$$

Dette bekrefter oppførselen vi så for feilen i Eksempel 12.5: Hvis h reduseres med faktor 2, så reduseres feilen med faktor $2^2 = 4$.

12.2.2 Estimering av steglengden

La oss si du ønsker å tilnærme et integral, og du ønsker at feilen i tilnærmingen ikke er større enn en øvre grense ϵ . Vi kan bruke feilestimatet (12.7) ved å kreve at

$$(b - a) \frac{h^2}{24} \max_{x \in [a, b]} M_2 \leq \epsilon$$

som vi lett kan løse for h ,

$$h \leq \sqrt{\frac{24\epsilon}{(b - a)M_2}}.$$

Dersom vi klarer å beregne (eller estimere) M_2 , maksimumsverdien til den andraderiverte, kan vi sette inn i uttrykket ovenfor, og finne en øvre grense for h som garanterer at integrasjonsfeilen blir mindre enn ϵ .

12.3 Alternativer til midtpunktsmetoden

Det finnes også mange andre metoder for numerisk integrasjon. Følgende strategi for å lage slike er en direkte parallell til Strategi 11.1 for numerisk derivasjon.

Strategi 12.8. La $n, k \geq 1$. En approksimasjon av

$$I = \int_a^b f(x) dx,$$

konstrueres som følger:

1. Del intervallet $[a, b]$ i nk like store delintervaller

$$a = x_0 < x_1 < x_2 < \cdots < x_{nk-1} < x_{nk} = b.$$

Sett $h = (b - a)/n$.

2. For hvert av delintervallene $J_i = [x_{(i-1)k}, x_{ik}]$, finn polynomet $p_{k,i}$ av grad k som interpolerer f i de $k+1$ punktene $\{x_{(i-1)k}, \dots, x_{ik}\}$.
3. Bruk $\int_{J_i} p_{k,i}(x) dx$ som en approksimasjon til $\int_{J_i} f(x) dx$, og legg sammen disse:

$$I(h) = \sum_{i=1}^n \int_{J_i} p_{k,i}(x) dx.$$

Strategi 12.8 kalles også *Newton–Cotes-metoden*. Ved hjelp av feilanalysen i Kapittel 10 kan vi estimere feilen i denne metoden:

Teorem 12.9. La $n, k \geq 1$ og la $f: [a, b] \rightarrow \mathbb{R}$ være en funksjon med $k+1$ kontinuerlige deriverte. Da vil feilen i Newton–Cotes-metoden av orden k med steglengde $h = (b-a)/n$ være høyst

$$|I - I(h)| \leq \frac{(b-a)M_{k+1}}{(k+1)!} h^{k+1}. \quad (12.8)$$

Bevis. Velg et delintervall J_i , og la $p_{k,i}$ være polynomet av orden k som interpolerer f i punktene $x_{ik}, \dots, x_{(i+1)k}$ (som beskrevet i Strategi 12.8). Fra Teorem 10.7 vet vi at for $x \in J_i$ vil

$$|f(x) - p_{k,i}(x)| \leq \frac{M_{k+1}}{(k+1)!} (x_{(i+1)k} - x_{ik})^{k+1} = \frac{M_{k+1}}{(k+1)!} h^{k+1}.$$

Dermed er

$$\begin{aligned} |I - I(h)| &= \left| \int_a^b f(x) dx - \sum_{i=1}^n \int_{J_i} p_{k,i}(x) dx \right| = \left| \sum_{i=1}^n \int_{J_i} f(x) - p_{k,i}(x) dx \right| \\ &\leq \sum_{i=1}^n \int_{J_i} |f(x) - p_{k,i}(x)| dx \\ &\leq \sum_{i=1}^n \int_{J_i} \frac{M_{k+1}}{(k+1)!} h^{k+1} dx \\ &= \frac{M_{k+1}}{(k+1)!} \sum_{i=1}^n h^{k+2} \\ &= \frac{M_{k+1}}{(k+1)!} nh^{k+2}. \end{aligned}$$

(Fra første til andre linje brukte vi trekantulikheten, så satt vi inn feilestimatet for interpolanten, og så integrerte vi konstanten $M_{k+1}/(k+1)!h^{k+1}$.) Om vi nå setter inn $h = (b-a)/n$ får vi (12.8). ■

Av feilestimatet (12.8) ser vi at for å senke feilen, kan vi øke k , senke h (det vil si øke n), eller en kombinasjon av disse. Hvilken kombinasjon som er mest effektiv kommer an på hvor stor faktoren M_{k+1} er for forskjellige k .

12.3.1 Trapesmetoden ($k = 1$)

Vi vet at linja som interpolerer f i to punkter x_{i-1} og x_i er

$$p_{1,i}(x) = f(x_{i-1}) + \frac{f(x_i) - f(x_{i-1})}{x_i - x_{i-1}}(x - x_{i-1}).$$

Bruker vi denne i Strategi 12.8 får vi *trapesmetoden*; som vi ser av Figur 12.4 tegner tilnærmingen i hvert delintervall J_i et trapes. Vi får da tilnærmingen

$$\int_{x_{i-1}}^{x_i} p_{1,i}(x) dx = \frac{f(x_{i-1}) + f(x_i)}{2} h.$$

Legger vi sammen bidragene fra alle delintervallene får vi approksimasjonen

$$I_{\text{trap}}(h) = \sum_{i=1}^n \frac{f(x_{i-1}) + f(x_i)}{2} h.$$

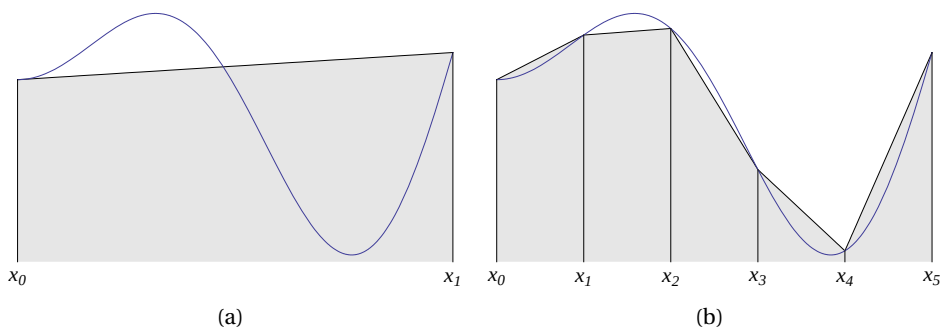
Siden x_i både er slutten på et intervall og starten på neste intervall når $i \neq 0, 1$, så kan dette også skrives

$$I_{\text{trap}}(h) = h \left(\frac{f(a) + f(b)}{2} + \sum_{i=1}^{n-1} f(x_i) \right). \quad (12.9)$$

Om vi anvender Teorem 10.7 med $k = 1$ får vi følgende feilestimat for trapesmetoden.

Teorem 12.10 (Feilestimat for trapesmetoden). La $f: [a, b] \rightarrow \mathbb{R}$ være en to ganger kontinuerlig deriverbar funksjon. Feilen i trapesmetoden med steglengde h er da begrenset av

$$|I - I_{\text{trap}}(h)| \leq \frac{(b-a)M_2}{2} h^2. \quad (12.10)$$



Figur 12.4. Trapesmetoden med ett delintervall (a) og fem delintervaller (b).

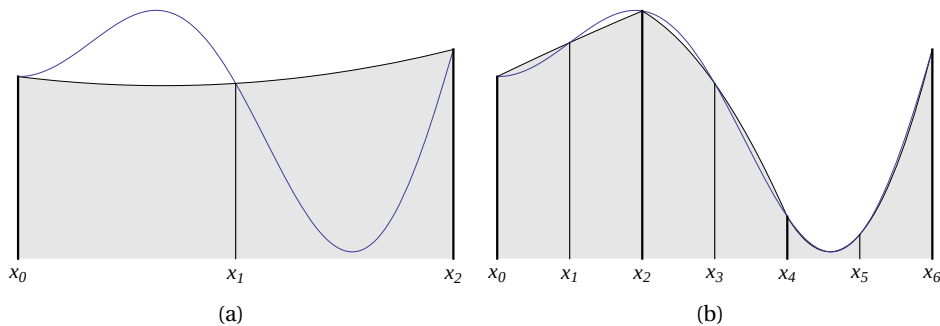
Med litt mer regning kan man vise at trapesmetoden har en feil som er høyst $\frac{(b-a)M_2}{12}h^2$, altså seks ganger lavere enn høyresiden i (12.10) (se Oppgave 13). Feil-estimatet (12.10) er altså et *overestimat*. Denne konstante faktoren på $\frac{1}{6}$ er ikke spesielt viktig; det viktige i (12.10) er

- faktoren h^2 , som gjør at hvis vi for eksempel halverer h , vil feilen gå ned med en faktor $1/4$
- faktoren M_2 , som sier at feilen blir større jo mer f oscillerer (det vil si, jo større f'' er).

Eksempel 12.11. La oss gjenta Eksempel 12.5 med trapesmetoden. Hvis vi igjen splitter intervallet i 2^k delintervaller, for $k = 1, 2, \dots, 10$, så får vi

h	$I_{\text{trap}}(h)$	$I - I_{\text{trap}}(h)$
0.500000	0.82386686	1.8×10^{-2}
0.250000	0.83708375	4.4×10^{-3}
0.125000	0.84037503	1.1×10^{-3}
0.062500	0.84119705	2.7×10^{-4}
0.031250	0.84140250	6.8×10^{-5}
0.015625	0.84145386	1.7×10^{-5}
0.007813	0.84146670	4.3×10^{-6}
0.003906	0.84146991	1.1×10^{-6}
0.001953	0.84147072	2.7×10^{-7}
0.000977	0.84147092	6.7×10^{-8}

Vi ser at hver gang steglengden halveres, så blir feilen redusert med omtrent faktor 4. Men vi ser også at selv om vi nå bruker to funksjonsverdier i hvert delintervall for å estimere integralet, så er feilen faktisk dobbelt så stor som for midt-



Figur 12.5. Simpsons metode med ett delintervall (a) og tre delintervaller (b).

punktsmetoden. Dette kan sees fra feilestimatet (12.10), siden $1/12$ er dobbelt så stor som $1/24$ i (12.7).

12.3.2 $k = 2$: Simpsons metode

For $k = 2$ tilnærmer vi med et kvadratisk polynom på hvert delintervall J_i . Metoden blir i dette tilfellet kalt *Simpsons metode*, og er illustrert i Figur 12.5. I Oppgave 16 blir du bedt å vise at Newtonformen til polynomet som interpolerer f i a , $m = (a + b)/2$, og b er

$$p_2(x) = f(a) + \frac{f(m) - f(a)}{h}(x - a) + \frac{f(a) - 2f(m) + f(b)}{2h^2}(x - a)(x - m),$$

der $h = (b - a)/2$, og at

$$\int_a^b p_2(x) dx = \frac{h}{3} (f(a) + 4f(m) + f(b)).$$

Dette er Simpsons metode med ett delintervall. Legger vi sammen bidragene fra alle delintervallene får vi

$$I_{\text{Simp}}(h) = \frac{h}{3} \left(f(a) + f(b) + 2 \sum_{i=1}^{n-1} f(x_{2i}) + 4 \sum_{i=1}^n f(x_{2i-1}) \right), \quad (12.11)$$

der nå $h = (b - a)/(2n)$. Setter vi inn $k = 2$ i Teorem 10.7, får vi følgende feilestimat for denne metoden.

Teorem 12.12 (Feilestimat for Simpsons metode). *La $f: [a, b] \rightarrow \mathbb{R}$ være en fire ganger kontinuerlig deriverbar funksjon. Feilen i Simpsons metode med steglengde h er da begrenset av*

$$|I - I_{\text{Simp}}(h)| \leq \frac{(b-a)M_3}{6} h^3. \quad (12.12)$$

Simpsons metode er mye brukt. Siden metoden baserer seg på kvadratiske polynomer, så må den være eksakt for slike. Den er faktisk eksakt for alle kubiske polynomer også (se Oppgave 14). Det finnes skarpere feilestimer for Simpsons metode enn dette (se Oppgave 15, som går et stykke utenfor pensum).

Oppgaver for Kapittel 12

1. I denne oppgaven skal vi bruke definisjonen av integralet på funksjonen $f(x) = e^x$ på intervallet $[0, 1]$.

(a). Skriv og kjør et program som regner ut $\underline{I}$ og $\bar{I}$ for en uniform partisjon på 10 delintervaller (la metoden ta antall delintervaller som argument). Hva blir feilene fra den eksakte verdien $I = e - 1 = 1.718281828$ av integralet?

Hint: I stedet for en for-løkke kan du bruke sum-kommandoen når du legger sammen bidragene fra delintervallene.

(b). Lag et program som finner hvor mange delintervaller som trengs for å regne ut $\underline{I}$ med en feil mindre enn 3×10^{-3} ?

2. Markér hver av følgende påstander som sanne eller gale.

(a). Midtpunktsmetoden er eksakt for alle polynomer av grad 1.

(b). Midtpunktsmetoden er eksakt for alle polynomer av grad 2.

(c). Den globale feilen i midtpunktsmetoden er av orden en lavere enn den lokale feilen.

(d). Når vi reduserer steglengden h i midtpunktsmetoden med faktor 3, så reduseres feilen med omtrent faktor 9.

3. Vi bruker midtpunktsmetoden med 2 delintervaller for å finne en tilnærming til integralet

$$\int_0^1 x^2 dx.$$

Hva får vi?

- ☐ 5/16
- ☐ 1/4
- ☐ 4/9
- ☐ 2/5

4. Finn en tilnærming til

$$\int_0^{\pi/2} \frac{\sin x}{1+x^2} dx = 0.526978557614\dots$$

ved hjelp av midtpunktsmetoden med 6 delintervaller.

5. (a). Test midtpunktsmetoden med 10 delintervaller på integralet

$$\int_0^1 e^x dx = e - 1.$$

(b). Finn en verdi av h som garanterer at den absolutte feilen er mindre enn 10^{-10} .

6. Gjenta forrige oppgave, men med integralet byttet ut med

$$\int_2^6 \ln x dx = \ln(11664) - 4.$$

7. Markér hver av de følgende påstandene som sanne eller gale.

- (a). Trapesmetoden er vanligvis mer nøyaktig enn midtpunktsmetoden.
- (b). Siden hvert evalueringspunkt i trapesmetoden forekommer som endepunkt i to forskjellige delintervaller, så må vi evaluere funksjonen vi integrerer to ganger i hvert punkt.
- (c). Simpsons metode krever at vi bruker et odde antall punkter.
- (d). Simpsons metode er eksakt for polynomer av grad 3 eller lavere.

8. Finn det korrekte alternativet i følgende flervalgsoppgaver.

(a). Vi bruker trapesmetoden med to delintervaller for å tilnærme integralet

$$\int_0^1 x^2 dx.$$

Hva blir resultatet?

- ☐ 1/2
- ☐ 3/8
- ☐ 5/9
- ☐ 3/5

(b). Hvilke(n) av trapes-, midtpunkts- eller Simpsons metode er mest nøyaktig for lineære polynomer?

- ☐ Trapesmetoden.
- ☐ Simpsons metode.
- ☐ Midtpunktsmetoden.
- ☐ Alle er eksakte.

(c). Vi bruker Simpsons metode for å regne ut tilnærminger til $\int_a^b f(x) dx$.
Da blir resultatet alltid eksakt hvis $f(x)$ er

- ☐ en trigonometrisk funksjon.
- ☐ en logaritmisk funksjon.
- ☐ et polynom av grad to.
- ☐ på formen $g(x)/h(x)$ der f og g er polynomer av grad to.

(d). ☐ Midtpunktsmetoden er mer nøyaktig enn Simpsons metode.

- ☐ Midtpunktsmetoden og trapesmetoden gir alltid samme feil.
- ☐ Midtpunktsmetoden gir 0 feil bare hvis $f(x) = c$ for en konstant c .
- ☐ Midtpunktsmetoden gir 0 feil hvis $f(x)$ er en vilkårlig rett linje i planet.

9. Gjenta Oppgave 4 og 5 i Seksjon 12.2 for trapesmetoden.

10. Hvor mange funksjonsevalueringer trenger du for å beregne integralet

$$\int_0^1 \frac{dx}{1+2x}$$

med feil mindre enn 10^{-10} for

- (a). trapesmetoden,
- (b). midtpunktsmetoden,
- (c). og Simpsons metode?

11. Gjenta Oppgave 4 og 5 i Seksjon 12.2 med Simpsons metode.

12. Vi vil lage en metode for numerisk integrasjon på formen

$$\int_a^b f(x) dx \approx w_1 f(a) + w_2 f((a+b)/2) + w_3 f(b),$$

der w_i skal bestemmes. Vis at, hvis metoden er eksakt for de tre polynomene $f(x) = x^i$ for $i = 0, 1, 2$, så blir metoden lik Simpsons metode.

13. La $m = \frac{a+b}{2}$ være midtpunktet på intervallet $[a, b]$, og se på følgende anvendelse av delvis integrasjon:

$$\begin{aligned} -\int_a^b (x-m)f'(x) dx &= -[(x-m)f(x)]_a^b + \int_a^b f(x) dx \\ &= -\frac{b-a}{2}f(b) - \frac{b-a}{2}f(a) + \int_a^b f(x) dx \\ &= \int_a^b f(x) dx - (b-a)\frac{f(a)+f(b)}{2}. \end{aligned}$$

Dette viser at den lokale feilen i trapesmetoden kan skrives som

$$L = -\int_a^b (x-m)f'(x) dx. \quad (12.13)$$

(a). Anvend delvis integrasjon på (12.13) for å vise at

$$L = \int_a^b \left(C + \frac{1}{2}(x-m)^2 \right) f''(x) dx. \quad (12.14)$$

der C er en konstant (vi skal komme tilbake tilbake hvordan vi skal velge C).

(b). Vis at, med $C = -(b-a)^2/8$, så har vi at $C + \frac{1}{2}(x-m)^2 \leq 0$ for $x \in [a, b]$.
Vis også at, med dette valget av C , så følger det fra (12.14) at $|L| \leq \frac{M_2(b-a)^3}{12}$.

(c). Bruk (b) til å vise at den globalen feilen i trapesmetoden er $\leq (b-a)\frac{h^2}{12}M_2$.

For å forenkle notasjonen i de neste oppgavene definerer vi $m = a + h$, og $b = a + 2h$. Vi definerer også

$$q(x) = (x - a)(x - m)(x - b),$$

og $p_2(x)$ som det kvadratiske polynomet som interpolerer f i a , m , og b .

14. I denne oppgaven skal vi se på forskjellige måter å vise at Simpsons metode er eksakt for alle polynomer av grad 3.

- (a). Verifiser at Simpsons metode er eksakt for $f(x) = x^i$ for $i = 0, 1, 2, 3$.
- (b). Bruk (a) til å vise at Simpsons metode er eksakt for ethvert kubisk polynom.
- (c). Kunne du komme frem til samme konklusjon som i (b) ved bare å betrakte feilestimatet (12.12)?
- (d). Forklar at q er anti-symmetrisk om m , og at, hvis f er et polynom av grad 3, så er $f - p_2$ et multiplum av q . Hvorfor følger eksakthet av Simpsons metode for polynomer av grad 3 fra dette?

15. I denne oppgaven skal vi bevise et skarpere feilestimat enn (12.12) for Simpsons metode. Vi lar

$$E = \int_a^b f(x) dx - \int_a^b p_2(x) dx$$

være feilen i Simpsons metode med ett intervall.

- (a). Vis at $\int_a^b xq(x) dx = -\frac{1}{120}(b-a)^5$, og at $(xq(x))^{(4)} = 4!$.
- (b). Se på den kontinuerlige funksjonen

$$\phi(x) = f(x) - p_2(x) - kq(x) - \frac{E}{\int_a^b xq(x) dx} xq(x),$$

Vis at

$$\phi^{(4)}(x) = f^{(4)}(x) + 120 \frac{4!E}{(b-a)^5}, \quad (12.15)$$

og at $\int_a^b \phi(x) dx = 0$. Til slutt, forklar at $\phi(a) = \phi(m) = \phi(b) = 0$.

- (c). Vi at det eksisterer en k slik at $\int_a^m \phi(x) dx = 0$.

For siste del av denne oppgaven lar vi k være definert som i (c).

(d). Vis at ϕ har minst fem forskjellige nullpunkter i $[a, b]$.

(e). Rolles theorem sier at, for en kontinuerlig deriverbar funksjon definert på $[a, b]$ slik at $\phi(a) = \phi(b) = 0$, så eksisterer det et punkt $c \in (a, b)$ slik at $\phi'(c) = 0$. Bruk Rolles teorem til å vise at det finnes en $\xi \in (a, b)$ slik at $\phi^{(4)}(\xi) = 0$.

(f). Forklar at, med ξ som i (e), så har vi at $E = -f^{(4)}(\xi)(b-a)^5/2880$. Det følger at feilen for ett intervall er $\leq M_4(b-a)^5/2880$. Siden vi har bidrag fra n slike, så blir den en øvre grense for feilen

$$nM_4(2h)^5/2880 = (b-a)M_4(2h)^4/2880 = M_4(b-a)h^4/180,$$

Dette er betydelig bedre enn det globale feilestimatet (12.12).

16. (a). Vis at Newtonformen til polynomet som interpolerer f i a , $m = (a+b)/2$, og b er

$$p_2(x) = f(a) + \frac{f(m) - f(a)}{h}(x-a) + \frac{f(a) - 2f(m) + f(b)}{2h^2}(x-a)(x-m),$$

der $h = (b-a)/2$.

(b). Vis at $\int_a^b p_2(x) dx = \frac{h}{3}(f(a) + 4f(m) + f(b))$.

Hint: Regn ut $\int_a^b (x-a)(x-m) dx$ først, se Oppgave 13 i Seksjon 10.3.

17. På intervallet $[x_{i-1}, x_i]$ bruker trapesmetoden kun funksjonsverdiene $f(x_{i-1})$ og $f(x_i)$. På det neste intervallet brukes verdiene $f(x_i)$ og $f(x_{i+1})$. Alle funksjonsverdiene, bortsett fra første og siste, forekommer derfor to ganger i summen til høyre i (12.9). Funksjonsevalueringer kan være kostbare — de kan for eksempel stamme fra sensorer som er dyre i bruk. Skriv om (12.9) på en måte slik at vi unngår gjentagende funksjonsevalueringer.

18. Anta at vi vet $n+1$ funksjonsverdier $\{f(a+kh)\}_{k=0}^n$. Vis at

$$\int_a^{a+nh} p_n(x) dx = hn \left(f(a) + \left(\frac{n}{2} \quad \frac{n^2}{3} \quad \dots \quad \frac{n^n}{n+1} \right) \begin{pmatrix} h^1 p_n^{(1)}(a)/1! \\ h^2 p_n^{(2)}(a)/2! \\ \vdots \\ h^n p_n^{(n)}(a)/n! \end{pmatrix} \right)$$

Vis også at, for $n=2$ og $n=3$, så får vi spesielt

$$\begin{aligned} \int_a^{a+2h} p_2(x) dx &= \frac{h}{3}(f(a) + 4f(a+h) + f(a+2h)) \\ \int_a^{a+3h} p_3(x) dx &= \frac{3h}{8}(f(a) + 3f(a+h) + 3f(a+2h) + f(a+3h)). \end{aligned}$$

Hint: Teorem 11.4 gir oss et uttrykk for $\begin{pmatrix} h^1 p_n^{(1)}(a)/1! \\ h^2 p_n^{(2)}(a)/2! \\ \vdots \\ h^n p_n^{(n)}(a)/n! \end{pmatrix}$.

APPENDIX

Solutions

Kapittel 1

Exercise 1. Vi har at

$$\begin{pmatrix} 2 \\ 1 \\ 4 \end{pmatrix} + 3 \begin{pmatrix} -1 \\ -4 \\ 3 \end{pmatrix} = \begin{pmatrix} 2 \\ 1 \\ 4 \end{pmatrix} + \begin{pmatrix} 3 \cdot (-1) \\ 3 \cdot (-4) \\ 3 \cdot 3 \end{pmatrix} = \begin{pmatrix} 2 \\ 1 \\ 4 \end{pmatrix} + \begin{pmatrix} -3 \\ -12 \\ 9 \end{pmatrix} = \begin{pmatrix} 2-3 \\ 1-12 \\ 4+9 \end{pmatrix} = \begin{pmatrix} -1 \\ -11 \\ 13 \end{pmatrix}$$

$$\begin{pmatrix} -i \\ -1 \\ 3 \end{pmatrix} + i \begin{pmatrix} -i \\ 2 \\ 2 \end{pmatrix} = \begin{pmatrix} -i \\ -1 \\ 3 \end{pmatrix} + \begin{pmatrix} i \cdot (-i) \\ i \cdot 2 \\ i \cdot 2 \end{pmatrix} = \begin{pmatrix} -i \\ -1 \\ 3 \end{pmatrix} + \begin{pmatrix} 1 \\ 2i \\ 2i \end{pmatrix} = \begin{pmatrix} -i+1 \\ -1+2i \\ 3+2i \end{pmatrix} = \begin{pmatrix} 1-i \\ -1+2i \\ 3+2i \end{pmatrix}.$$

Exercise 2. Linja kan parametriseres ved

$$\mathbf{y}(t) = (1-t) \begin{pmatrix} 1 \\ 1 \\ 1 \end{pmatrix} + t \begin{pmatrix} 2 \\ 4 \\ 3 \end{pmatrix} = \begin{pmatrix} 1-t \\ 1-t \\ 1-t \end{pmatrix} + \begin{pmatrix} 2t \\ 4t \\ 3t \end{pmatrix} = \begin{pmatrix} 1+t \\ 1+3t \\ 1+2t \end{pmatrix}.$$

Planet består av alle punkter (x, y, z) slik at $(x, y, z) - (3, -1, -1)$ står ortogonalt på $(1, 1, 1)$, det vil si at

$$0 = \left(\begin{pmatrix} x \\ y \\ z \end{pmatrix} - \begin{pmatrix} 3 \\ -1 \\ -1 \end{pmatrix} \right) \cdot \begin{pmatrix} 1 \\ 1 \\ 1 \end{pmatrix} = \begin{pmatrix} x-3 \\ y+1 \\ z+1 \end{pmatrix} \cdot \begin{pmatrix} 1 \\ 1 \\ 1 \end{pmatrix} = x-3 + y+1 + z+1 = x+y+z-1.$$

Planet består derfor av alle punkter slik at $x+y+z=1$.

Exercise 3. Vi har at

$$|(3, 4, 5)| = \sqrt{3^2 + 4^2 + 5^2} = \sqrt{9 + 16 + 25} = \sqrt{50} = 5\sqrt{2}$$

$$|(i, -i, 1, -1)| = \sqrt{|i|^2 + |-i|^2 + |1|^2 + |-1|^2} = \sqrt{1^2 + (-1)^2 + 1 + 1} = \sqrt{4} = 2.$$

Exercise 4(a).

$$(-1, 2, 4) \cdot (3, 3, -1) = -1 \cdot 3 + 2 \cdot 3 + 4 \cdot (-1) = -3 + 6 - 4 = -1.$$

Exercise 4(b).

$$\begin{aligned}(-i, 2, 3 + i) \cdot (2 - i, -4i, -2) &= -i \cdot \overline{2 - i} + 2 \cdot \overline{-4i} + (3 + i) \cdot \overline{-2} \\&= -i(2 + i) + 2 \cdot (4i) + (3 + i) \cdot (-2) \\&= -2i + 1 + 8i - 6 - 2i = -5 + 4i.\end{aligned}$$

Exercise 5. Vi regner ut

$$\mathbf{a} \cdot \mathbf{b} = (1, 2, 4) \cdot (2, 4, 2) = 2 + 8 + 8 = 18$$

$$|\mathbf{b}|^2 = 2^2 + 4^2 + 2^2 = 4 + 16 + 4 = 24$$

Dermed blir

$$\text{proj}_{\mathbf{b}} \mathbf{a} = \frac{\mathbf{a} \cdot \mathbf{b}}{|\mathbf{b}|^2} \mathbf{b} = \frac{18}{24} \begin{pmatrix} 2 \\ 4 \\ 2 \end{pmatrix} = \begin{pmatrix} 3/2 \\ 3 \\ 3/2 \end{pmatrix}.$$

Exercise 6. Alle disse regnereglene følger fra de tilsvarende regnereglene for reelle tall, når vi kombinerer med definisjonene av vektoraddisjon og multiplikasjon med en skalar. Vi tar 1.-3. i Proposisjon 1.6 først.

1. $\mathbf{a} + \mathbf{b}$ har komponenter $a_i + b_i$, og $\mathbf{b} + \mathbf{a}$ har komponenter $b_i + a_i$. Siden $a_i + b_i = b_i + a_i$ (rekkefølgen ved addisjon av tall spiller ingen rolle), så er alle komponentene like, slik at $\mathbf{a} + \mathbf{b} = \mathbf{b} + \mathbf{a}$.
2. $\mathbf{a} + \mathbf{b}$ har komponenter $a_i + b_i$, og $(\mathbf{a} + \mathbf{b}) + \mathbf{c}$ har da komponenter $(a_i + b_i) + c_i$. På samme måte har $\mathbf{a} + (\mathbf{b} + \mathbf{c})$ komponenter $a_i + (b_i + c_i)$. Siden

$$(a_i + b_i) + c_i = a_i + (b_i + c_i)$$

er en kjent regneregul for tall, så følger igjen at alle komponentene er like, slik at $(\mathbf{a} + \mathbf{b}) + \mathbf{c} = \mathbf{a} + (\mathbf{b} + \mathbf{c})$.

3. $s(\mathbf{a} + \mathbf{b})$ har komponenter $s(a_i + b_i)$, og $s\mathbf{a} + s\mathbf{b}$ har komponenter $sa_i + sb_i$. Igjen er $s(a_i + b_i) = sa_i + sb_i$ en kjent regneregul, slik at $s(\mathbf{a} + \mathbf{b}) = s\mathbf{a} + s\mathbf{b}$.

Så tar vi 1.-3. i Proposisjon 1.9.

- 1.

$$\mathbf{a} \cdot \mathbf{a} = (a_1, \dots, a_n) \cdot (a_1, \dots, a_n) = a_1 \cdot a_1 + \dots + a_n \cdot a_n = a_1^2 + \dots + a_n^2 = |\mathbf{a}|^2$$

2.

$$\begin{aligned}\mathbf{a} \cdot \mathbf{b} &= (a_1, \dots, a_n) \cdot (b_1, \dots, b_n) = a_1 b_1 + \dots + a_n b_n \\ &= b_1 a_1 + \dots + b_n a_n = (b_1, \dots, b_n) \cdot (a_1, \dots, a_n) = \mathbf{b} \cdot \mathbf{a}\end{aligned}$$

3.

$$\begin{aligned}\mathbf{c} \cdot (\mathbf{a} + \mathbf{b}) &= (c_1, \dots, c_n) \cdot ((a_1, \dots, a_n) + (b_1, \dots, b_n)) \\ &= (c_1, \dots, c_n) \cdot (a_1 + b_1, \dots, a_n + b_n) \\ &= c_1(a_1 + b_1) + \dots + c_n(a_n + b_n) \\ &= c_1 a_1 + c_1 b_1 + \dots + c_n a_n + c_n b_n \\ &= c_1 a_1 + \dots + c_n a_n + c_1 b_1 + \dots + c_n b_n \\ &= (c_1, \dots, c_n) \cdot (a_1, \dots, a_n) + (c_1, \dots, c_n) \cdot (b_1, \dots, b_n) \\ &= \mathbf{c} \cdot \mathbf{a} + \mathbf{c} \cdot \mathbf{b}.\end{aligned}$$

Det andre punktet i 3. her er helt tilsvarende.

Exercise 7. Vi har at

$$\begin{aligned}|\mathbf{s}\mathbf{a}| &= |(sa_1, \dots, sa_n)| = \sqrt{|sa_1|^2 + \dots + |sa_n|^2} \\ &= \sqrt{|s|^2|a_1|^2 + \dots + |s|^2|a_n|^2} = |s|\sqrt{|a_1|^2 + \dots + |a_n|^2} = |s||\mathbf{a}|.\end{aligned}$$

Exercise 8. Fra trekantulikheten har vi at

$$|\mathbf{a}| = |(\mathbf{a} - \mathbf{b}) + \mathbf{b}| \leq |\mathbf{a} - \mathbf{b}| + |\mathbf{b}|.$$

Fra dette får vi at $|\mathbf{a}| - |\mathbf{b}| \leq |\mathbf{a} - \mathbf{b}|$. hvis vi lar $\mathbf{a}$ og $\mathbf{b}$ bytte roller får vi også at $|\mathbf{b}| - |\mathbf{a}| \leq |\mathbf{b} - \mathbf{a}| = |\mathbf{a} - \mathbf{b}|$. Både $|\mathbf{b}| - |\mathbf{a}|$ og $|\mathbf{a}| - |\mathbf{b}|$ er altså $\leq |\mathbf{a} - \mathbf{b}|$. Men da følger det at

$$||\mathbf{a}| - |\mathbf{b}|| \leq |\mathbf{a} - \mathbf{b}|.$$

Exercise 9(a). Sett $A = \{r\mathbf{a} + s\mathbf{b} : 0 \leq r, s \leq 1\}$. Linjen $r\mathbf{a} + s\mathbf{b}$, $0 \leq s \leq 1$ beskriver en linje i parallelogrammet som forbinder punktene $r\mathbf{a}$ og $r\mathbf{a} + \mathbf{b}$. Siden dette holder for alle r er det klart at A er inneholdt i parallelogrammet.

Det er også geometrisk klart at, for ethvert punkt i parallelogrammet, så finnes et en r slik at punktet ligger på linjen gjennom $r\mathbf{a}$ og $r\mathbf{a} + \mathbf{b}$. Parallelogrammet er derfor også inneholdt i A , slik at A må være identisk med hele parallelogrammet.

Exercise 9(b). Vi kan tenke på $|\mathbf{a}|$ som grunnflaten til parallelogrammet, og $|\mathbf{b}| \sin \theta$ som høyden. Arealet er gitt ved grunnflate ganger høyde, det vil si $|\mathbf{a}||\mathbf{b}| \sin \theta$.

Exercise 10. $\mathbf{a} + \mathbf{b}$ og $\mathbf{a} - \mathbf{b}$ svarer til diagonalene til parallelogrammet utspent av $\mathbf{a}$ og $\mathbf{b}$. Siden $\mathbf{a}$ og $\mathbf{b}$ svarer til sidene i parallelogrammet, så gir loven oss en sammenheng mellom diagonalene og sidene i et parallelogram. Dette forklarer hvordan denne identiteten har fått dette navnet.

Exercise 11. Vi har at

$$|\mathbf{a} + \mathbf{b}|^2 = f(\mathbf{a} + \mathbf{b}, \mathbf{a} + \mathbf{b}) = f(\mathbf{a}, \mathbf{a}) + f(\mathbf{a}, \mathbf{b}) + f(\mathbf{b}, \mathbf{a}) + f(\mathbf{b}, \mathbf{b}) = |\mathbf{a}|^2 + 2f(\mathbf{a}, \mathbf{b}) + |\mathbf{b}|^2.$$

Det følger at

$$f(\mathbf{a}, \mathbf{b}) = \frac{1}{2}(|\mathbf{a} + \mathbf{b}|^2 - |\mathbf{a}|^2 - |\mathbf{b}|^2) = \frac{1}{2} \sum_{k=1}^n (a_k + b_k)^2 - a_k^2 - b_k^2 = \sum_{k=1}^n a_k b_k.$$

Kapittel 2

Exercise 1(a). Vi har at

$$\begin{pmatrix} 2 & 4 & 3 \\ -1 & 2 & 4 \\ 4 & 1 & -1 \end{pmatrix} - \begin{pmatrix} -1 & 2 & 8 \\ -1 & -2 & -5 \\ 2 & 4 & -2 \end{pmatrix} = \begin{pmatrix} 2 - (-1) & 4 - 2 & 3 - 8 \\ -1 - (-1) & 2 - (-2) & 4 - (-5) \\ 4 - 2 & 1 - 4 & -1 - (-2) \end{pmatrix} \begin{pmatrix} 3 & 2 & -5 \\ 0 & 4 & 9 \\ 2 & -3 & 1 \end{pmatrix}$$

Exercise 1(b). Vi har at

$$\begin{aligned} 2 \begin{pmatrix} 3 & 2 \\ -1 & -2 \end{pmatrix} + 4 \begin{pmatrix} -3 & 3 \\ -1 & -1 \end{pmatrix} &= \begin{pmatrix} 2 \cdot 3 & 2 \cdot 2 \\ 2 \cdot (-1) & 2 \cdot (-2) \end{pmatrix} + \begin{pmatrix} 4 \cdot (-3) & 4 \cdot 3 \\ 4 \cdot (-1) & 4 \cdot (-1) \end{pmatrix} \\ &= \begin{pmatrix} 6 & 4 \\ -2 & -4 \end{pmatrix} + \begin{pmatrix} -12 & 12 \\ -4 & -4 \end{pmatrix} \\ &= \begin{pmatrix} 6 - 12 & 4 + 12 \\ -2 - 4 & -4 - 4 \end{pmatrix} = \begin{pmatrix} -6 & 16 \\ -6 & -8 \end{pmatrix} \end{aligned}$$

Exercise 2.

1. $A + B$ har komponenter $a_{ij} + b_{ij}$, og $B + A$ har komponenter $b_{ij} + a_{ij}$. Igjen bruker vi at rekkefølgen på addisjon av tall ikke spiller noen rolle. Alle komponenter i $A + B$ og $B + A$ er derfor like, slik at også $A + B = B + A$.
2. $(A + B) + C$ har komponenter $(a_{ij} + b_{ij}) + c_{ij}$, og $A + (B + C)$ har komponenter $a_{ij} + (b_{ij} + c_{ij})$. Igjen kan vi bruke en kjent regneregul for tall, slik at komponentene i venstresiden og høyresiden er like, slik at $(A + B) + C = A + (B + C)$.

3. $s(A+B)$ har komponenter $s(a_{ij} + b_{ij})$, og $sA + sB$ har komponenter $sa_{ij} + sb_{ij}$. Også nå kan vi bruke en kjent regneregel til å konkludere at venstresiden og høyresiden er like.

Exercise 3.

$$\begin{pmatrix} 2 & 4 \\ 3 & -1 \\ 0 & 1 \end{pmatrix}^T = \begin{pmatrix} 2 & 3 & 0 \\ 4 & -1 & 1 \end{pmatrix}$$

$$\begin{pmatrix} 1 & 4 & 7 \\ 2 & 4 & 3 \\ -1 & -2 & 1 \end{pmatrix}^T = \begin{pmatrix} 1 & 2 & -1 \\ 4 & 4 & -2 \\ 7 & 3 & 1 \end{pmatrix}$$

Exercise 4. Koden sjekker om $(A+B)+C = A+(B+C)$ er riktig for tre gitte matriser. Hvis tallet 0 blir skrevet ut så mener maskinen at de to er like. I praksis vil faktisk ikke dette skje. På grunn av avrundingsfeil på maskinen så vil ikke maskinen regne ut nøyaktig 0.

Exercise 5. Vi har at

$$AB = \begin{pmatrix} 1 & 4 & -1 \\ -2 & -1 & 4 \end{pmatrix} \begin{pmatrix} 1 & 0 \\ 3 & 2 \\ 4 & 2 \end{pmatrix} = \begin{pmatrix} 1+12-4 & 0+8-2 \\ -2-3+16 & 0-2+8 \end{pmatrix} = \begin{pmatrix} 9 & 6 \\ 11 & 6 \end{pmatrix}$$

Alternativ 2. er derfor riktig.

Exercise 6. Vi har $A^T \in \mathbb{R}^{5 \times 3}$ og $B \in \mathbb{R}^{6 \times 3}$, så A^T har ikke like mange kolonner (3) som B har rader (6).

Exercise 7(a).

$$\begin{pmatrix} 2 & 4 & 3 \\ 1 & -1 & -2 \\ -3 & -1 & 1 \end{pmatrix} \begin{pmatrix} 2 \\ 5 \\ 3 \end{pmatrix} = \begin{pmatrix} 4+20+9 \\ 2-5-6 \\ -6-5+3 \end{pmatrix} = \begin{pmatrix} 33 \\ -9 \\ -8 \end{pmatrix}$$

Exercise 7(b).

$$\begin{pmatrix} 1 & 3 \\ 2 & 4 \end{pmatrix} \begin{pmatrix} -1 & 0 \\ 3 & 2 \end{pmatrix} = \begin{pmatrix} -1+9 & 0+6 \\ -2+12 & 0+8 \end{pmatrix} = \begin{pmatrix} 8 & 6 \\ 10 & 8 \end{pmatrix}$$

Exercise 7(c).

$$\begin{pmatrix} -1 & 2 & 4 & 1 \\ 1 & 3 & 2 & 4 \end{pmatrix} \begin{pmatrix} 2 & 1 \\ 2 & 0 \\ 1 & 0 \\ 3 & 4 \end{pmatrix} = \begin{pmatrix} -2+4+4+3 & -1+0+0+4 \\ 2+6+2+12 & 1+0+0+16 \end{pmatrix} = \begin{pmatrix} 9 & 3 \\ 22 & 17 \end{pmatrix}$$

Exercise 8. Vi viser 2. først. Komponent (i, j) i $A(B + C)$ er

$$\sum_k a_{ik}(B + C)_{kj} = \sum_k a_{ik}(b_{kj} + c_{kj}). \quad (.16)$$

Komponent (i, j) i AB er $\sum_k a_{ik}b_{kj}$, og komponent (i, j) i AC er $\sum_k a_{ik}c_{kj}$. Ved definisjonen av matriseaddisjon er da komponent (i, j) i $AB + AC$

$$\sum_k a_{ik}b_{kj} + \sum_k a_{ik}c_{kj} \quad (.17)$$

Fra kjente regneregler vet vi at (.16) og (.17) er like. $A(B + C)$ og $AB + AC$ har derfor like komponenter, slik at $A(B + C) = AB + AC$. 3. følger på samme måte.

Vi viser til slutt 4..

- Komponent (i, j) i $s(AB)$ er $s \sum_k a_{ik}b_{kj}$.
- Komponent (i, j) i $(sA)B$ er $\sum_k (sa_{ik})b_{kj}$
- Komponent (i, j) i $A(sB)$ er $\sum_k a_{ik}(sb_{kj})$.

At disse tre er like følger nå fra regnereglene for reelle tall.

Exercise 9. Vi har

$$A^2 = \begin{pmatrix} 1 & 4 \\ -2 & -1 \end{pmatrix} \begin{pmatrix} 1 & 4 \\ -2 & -1 \end{pmatrix} = \begin{pmatrix} 1-8 & 4-4 \\ -2+2 & -8+1 \end{pmatrix} = \begin{pmatrix} -7 & 0 \\ 0 & -7 \end{pmatrix}$$

$$A^3 = A^2 A = \begin{pmatrix} -7 & 0 \\ 0 & -7 \end{pmatrix} \begin{pmatrix} 1 & 4 \\ -2 & -1 \end{pmatrix} = \begin{pmatrix} -7 & -28 \\ 14 & 7 \end{pmatrix}$$

$$A^4 = A^2 A^2 = \begin{pmatrix} -7 & 0 \\ 0 & -7 \end{pmatrix} \begin{pmatrix} -7 & 0 \\ 0 & -7 \end{pmatrix} = \begin{pmatrix} 49 & 0 \\ 0 & 49 \end{pmatrix}.$$

Exercise 10. Vi har at $F(\mathbf{e}_1) = \mathbf{e}_2$, og at $F(\mathbf{e}_2) = \mathbf{e}_1$. Det følger at matrisen til A er

$$(F(\mathbf{e}_1) \quad F(\mathbf{e}_2)) = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}$$

Exercise 11. Vi har at komponent (i, j) i $(AB)^T$ er lik komponent (j, i) i AB , som er $\sum_k a_{jk}b_{ki}$.

Vi har også at komponent (i, j) i $B^T A^T$ er

$$\sum_k (B^T)_{ik} (A^T)_{kj} = \sum_k b_{ki} a_{jk} = \sum_k a_{jk} b_{ki}.$$

Det følger at matrisene $(AB)^T$ og $B^T A^T$ har likhet i alle komponenter, slik at $(AB)^T = B^T A^T$.

Kapittel 3

Exercise 1(a). $(2, 4, 3)$

Exercise 1(b). Systemet har ingen løsning.

Exercise 1(c). $\mathbf{x} = (3/2, x_3 - 1/4, x_3) = (3/2, -1/4, 0) + x_3(0, 1, 1)$.

Exercise 2. $\mathbf{x}_1 = (2, 1, -3)$ og $\mathbf{x}_2 = (0, -3, 2)$.

Exercise 5.

$$\begin{pmatrix} 3 & 1 & 0 & 7 \\ 2 & -3 & 4 & -3 \\ 0 & 2 & -2 & 4 \end{pmatrix} \sim \begin{pmatrix} 1 & 1/3 & 0 & 7/3 \\ 0 & -11/3 & 4 & -23/3 \\ 0 & 1 & -1 & 2 \end{pmatrix} \sim \begin{pmatrix} 1 & 1/3 & 0 & 7/3 \\ 0 & 1 & 1 & 2 \\ 0 & 0 & 1/3 & -1/3 \end{pmatrix} \\ \sim \begin{pmatrix} 1 & 0 & 0 & 2 \\ 0 & 1 & 0 & 1 \\ 0 & 0 & 1 & -1 \end{pmatrix}$$

Exercise 6.

$$\begin{pmatrix} 2 & 3 & 4 & 5 \\ 2 & 5 & 4 & 7 \\ 4 & 7 & 8 & a \end{pmatrix} \sim \begin{pmatrix} 2 & 3 & 4 & 5 \\ 0 & 2 & 0 & 2 \\ 0 & 1 & 0 & a-10 \end{pmatrix} \sim \begin{pmatrix} 2 & 3 & 4 & 5 \\ 0 & 1 & 0 & 1 \\ 0 & 0 & 0 & a-11 \end{pmatrix}$$

Vi ser at det er bare for $a = 11$ at siste kolonne ikke blir en pivotkolonne.

Exercise 7. Vi får

$$\begin{pmatrix} 1 & 2 & -2 & 2 \\ 2 & 4 & -1 & 16 \\ 3 & 6 & -5 & 11 \end{pmatrix} \sim \begin{pmatrix} 1 & 2 & -2 & 2 \\ 0 & 0 & 3 & 12 \\ 0 & 0 & 1 & 5 \end{pmatrix} \sim \begin{pmatrix} 1 & 2 & -2 & 2 \\ 0 & 0 & 1 & 4 \\ 0 & 0 & 0 & 1 \end{pmatrix} \sim \begin{pmatrix} 1 & 2 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

Exercise 8.

$$\begin{pmatrix} 1 & 2 & 4 \\ 2 & 4 & 9 \\ -2 & -4 & 1 \\ -3 & -6 & 0 \end{pmatrix} \sim \begin{pmatrix} 1 & 2 & 4 \\ 0 & 0 & 1 \\ 0 & 0 & 9 \\ 0 & 0 & 12 \end{pmatrix} \sim \begin{pmatrix} 1 & 2 & 0 \\ 0 & 0 & 1 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{pmatrix}$$

Vi ser at kolonne 1 og 3 er pivotkolonner.

Exercise 9. Vi har at

$$\begin{pmatrix} 2 \\ 8 \\ 4 \end{pmatrix} = 2 \begin{pmatrix} 1 \\ 0 \\ 2 \end{pmatrix} + 2 \begin{pmatrix} 0 \\ 4 \\ 0 \end{pmatrix}.$$

Derfor er løsningen av det siste systemet

$$2 \begin{pmatrix} 5 \\ 6 \\ -3 \end{pmatrix} + 2 \begin{pmatrix} 3 \\ -1 \\ 2 \end{pmatrix} = \begin{pmatrix} 16 \\ 10 \\ -2 \end{pmatrix}.$$

Exercise 10. De siste radoperasjonene kan programmeres som følger.

```
A[2,:] = A[2,:] + 8*A[1,:]
A[2,:]=(-5)*A[2,:]
A[0,:]=A[0,:]-A[2,:]
A[1,:] = A[1,:] +(2/5)*A[2,:]
A[0,:] = A[0,:] + 2*A[1,:]
```

Exercise 11. La $\mathbf{x}$ være en løsning på systemet. Hvis det finnes en slik $\mathbf{c}$ så vil da $\mathbf{c}^T A \mathbf{x} = \mathbf{c}^T \mathbf{b}$ si at

$$0x_1 + 0x_2 + \cdots + 0x_n = 1,$$

som er umulig. Det kan derfor ikke finnes en slik $\mathbf{c}$, og dette viser den ene veien.

Anta nå at systemet ikke har en løsning. Etter Gausseliminasjon på den utvidede matrisen må vi da ende opp med en likning som over. Denne likningen er fått ved å legge sammen multiple av alle likningene. La $\mathbf{c}$ være vektoren med koeffisienter for hver likning. Da er $\mathbf{c}^T A = \mathbf{0}$ og $\mathbf{c}^T \mathbf{b} = 1$, slik at det finnes en slik $\mathbf{c}$.

Kapittel 4

Exercise 1(a). Vi radreduserer:

$$\begin{pmatrix} 2 & -1 & 1 & 1 & 0 & 0 \\ 0 & 1 & 4 & 0 & 1 & 0 \\ 1 & -2 & -6 & 0 & 0 & 1 \end{pmatrix} \xrightarrow{I \leftrightarrow III} \begin{pmatrix} 1 & -2 & -6 & 0 & 0 & 1 \\ 0 & 1 & 4 & 0 & 1 & 0 \\ 2 & -1 & 1 & 1 & 0 & 0 \end{pmatrix} \xrightarrow{III \sim 2I} \begin{pmatrix} 1 & -2 & -6 & 0 & 0 & 1 \\ 0 & 1 & 4 & 0 & 1 & 0 \\ 0 & 3 & 13 & 1 & 0 & -2 \end{pmatrix}$$

$$\xrightarrow{III \sim 3II} \begin{pmatrix} 1 & -2 & -6 & 0 & 0 & 1 \\ 0 & 1 & 4 & 0 & 1 & 0 \\ 0 & 0 & 1 & 1 & -3 & -2 \end{pmatrix} \xrightarrow{II \sim 4III} \begin{pmatrix} 1 & -2 & -6 & 0 & 0 & 1 \\ 0 & 1 & 0 & -4 & 13 & 8 \\ 0 & 0 & 1 & 1 & -3 & -2 \end{pmatrix}$$

$$\xrightarrow{I \sim 6III} \begin{pmatrix} 1 & -2 & 0 & 6 & -18 & -11 \\ 0 & 1 & 0 & -4 & 13 & 8 \\ 0 & 0 & 1 & 1 & -3 & -2 \end{pmatrix} \xrightarrow{I \sim 2II} \begin{pmatrix} 1 & 0 & 0 & -2 & 8 & 5 \\ 0 & 1 & 0 & -4 & 13 & 8 \\ 0 & 0 & 1 & 1 & -3 & -2 \end{pmatrix}$$

Det følger at $A^{-1} = \begin{pmatrix} -2 & 8 & 5 \\ -4 & 13 & 8 \\ 1 & -3 & -2 \end{pmatrix}$.

Exercise 1(b). Fra radoperasjonene over følger at vi kan skrive

$$A^{-1} = \begin{pmatrix} 1 & 2 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} 1 & 0 & 6 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & -4 \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & -3 & 1 \end{pmatrix} \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ -2 & 0 & 1 \end{pmatrix} \begin{pmatrix} 0 & 0 & 1 \\ 0 & 1 & 0 \\ 1 & 0 & 0 \end{pmatrix}$$

$$A = \begin{pmatrix} 0 & 0 & 1 \\ 0 & 1 & 0 \\ 1 & 0 & 0 \end{pmatrix} \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 2 & 0 & 1 \end{pmatrix} \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 3 & 1 \end{pmatrix} \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 4 \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} 1 & 0 & -6 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} 1 & -2 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix}$$

For A tok vi de inverse radoperasjonene i motsatt rekkefølge.

Exercise 2. Vi har at

$$\begin{pmatrix} a & b \\ c & d \end{pmatrix} \frac{1}{ad-bc} \begin{pmatrix} d & -b \\ -c & a \end{pmatrix}$$

$$= \frac{1}{ad-bc} \begin{pmatrix} ad-bc & -ab+ab \\ cd-cd & -bc+ad \end{pmatrix} = \frac{1}{ad-bc} \begin{pmatrix} ad-bc & 0 \\ 0 & ad-bc \end{pmatrix} = I_n.$$

Det følger at $\begin{pmatrix} a & b \\ c & d \end{pmatrix}^{-1} = \frac{1}{ad-bc} \begin{pmatrix} d & -b \\ -c & a \end{pmatrix}$ når $ad-bc \neq 0$. Når $ad-bc = 0$ har vi to muligheter:

1. $ad = bc = 0$: A må da ha en nullrad eller nullsøyle, og vi vet da at den ikke kan radreduseres til I_n , slik at den ikke er inverterbar.
2. $ad = bc \neq 0$: Da er alle fire tall $\neq 0$. Vi får derfor

$$\begin{pmatrix} a & b \\ c & d \end{pmatrix} \sim \begin{pmatrix} 1 & b/a \\ 0 & d-bc/a \end{pmatrix} = \begin{pmatrix} 1 & b/a \\ 0 & (ad-bc)/a \end{pmatrix} = \begin{pmatrix} 1 & b/a \\ 0 & 0 \end{pmatrix},$$

som heller ikke kan radreduseres til I_n .

Exercise 3. Anta at A er inverterbar. Da er

$$(A^{-1})^T A^T = (AA^{-1})^T = I^T = I,$$

slik at A^T er inverterbar med invers $(A^T)^{-1} = (A^{-1})^T$. Den andre veien følger ved å erstatte A med A^T .

Exercise 7. Løsningen av $A\mathbf{x} = \mathbf{b}$ er $\mathbf{x} = A^{-1}\mathbf{b}$, og løsningen av $\mathbf{y}A = \mathbf{b}$ er $\mathbf{y} = \mathbf{b}A^{-1}$. Følgende kode genererer tilfeldige A og $\mathbf{b}$, og sammenligner $A \setminus \mathbf{b}$ og $\mathbf{b}/A$ med disse løsningene.

```
A=rand(5,5);
b=rand(5,1);
A\b-inv(A)*b

b=rand(1,5);
b/A-b*inv(A)
```

Kapittel 5

Exercise 1. Siden rekkefølgen av vektorene $\mathbf{a}_i$ i $\text{Sp}(\mathbf{a}_1, \mathbf{a}_2, \dots, \mathbf{a}_n)$ ikke spiller noen rolle, så er det klart at $\text{Row}(A) = \text{Row}(B)$ hvis B fremkommer fra A ved et radbytte. Siden

$$\text{Sp}(s\mathbf{a}_1, \mathbf{a}_2, \dots, \mathbf{a}_n) = \text{Sp}(\mathbf{a}_1, \mathbf{a}_2, \dots, \mathbf{a}_n)$$

så er det også klart at $\text{Row}(A) = \text{Row}(B)$ hvis B fremkommer fra A ved å gange en rad med en skalar $\neq 0$.

Til slutt er det klart at

$$\text{Sp}(\mathbf{a}_1 + c\mathbf{a}_2, \mathbf{a}_2, \dots, \mathbf{a}_n) \subseteq \text{Sp}(\mathbf{a}_1, \mathbf{a}_2, \dots, \mathbf{a}_n)$$

Siden også $\mathbf{a}_1 = (\mathbf{a}_1 + c\mathbf{a}_2) - c\mathbf{a}_2$ så er det også klart at

$$\text{Sp}(\mathbf{a}_1, \mathbf{a}_2, \dots, \mathbf{a}_n) \subseteq \text{Sp}(\mathbf{a}_1 + c\mathbf{a}_2, \mathbf{a}_2, \dots, \mathbf{a}_n)$$

Det følger at

$$\text{Sp}(\mathbf{a}_1 + c\mathbf{a}_2, \mathbf{a}_2, \dots, \mathbf{a}_n) = \text{Sp}(\mathbf{a}_1, \mathbf{a}_2, \dots, \mathbf{a}_n).$$

Dermed er det klart at $\text{Row}(A) = \text{Row}(B)$ også hvis B fremkommer fra A ved å legge til et multiplum av en rad til en annen. Dermed vil ingen elementære radoperasjoner endre radrommet til matrisen, slik at radekvivalente matriser også har samme radrom.

Exercise 2(a). Radreduksjon av A gir $A \sim \begin{pmatrix} 1 & 0 & 3/2 & 11/6 \\ 0 & 1 & -1/2 & -7/6 \\ 0 & 0 & 0 & 0 \end{pmatrix}$. Vi ser at de to

første radene er pivotrader, slik at disse er en basis for radrommet til A , som derfor har dimensjon 2. Vi ser også at de to første søylene i A er pivotsøyer, slik at disse er en basis for søylerommet til A , som også har dimensjon 2. Rangen til A er dermed 2.

Exercise 2(b). En Vektor i nullrommet på oppfylle

$$x_1 + \frac{3}{2}x_3 + \frac{11}{6}x_4 = 0 \qquad x_2 - \frac{1}{2}x_3 - \frac{7}{6}x_4 = 0,$$

slik at

$$x_1 = -\frac{3}{2}x_3 - \frac{11}{6}x_4 \qquad x_2 = \frac{1}{2}x_3 + \frac{7}{6}x_4,$$

slik at

$$\begin{pmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{pmatrix} = \begin{pmatrix} -\frac{3}{2}x_3 - \frac{11}{6}x_4 \\ \frac{1}{2}x_3 + \frac{7}{6}x_4 \\ x_3 \\ x_4 \end{pmatrix} = x_3 \begin{pmatrix} -\frac{3}{2} \\ \frac{1}{2} \\ 1 \\ 0 \end{pmatrix} + x_4 \begin{pmatrix} -\frac{11}{6} \\ \frac{7}{6} \\ 0 \\ 1 \end{pmatrix},$$

slik at $(-3, 1, 2, 0)$ og $(-11, 7, 0, 6)$ er en basis for nullrommet til A , som dermed har dimensjon 2. Det siste kunne vi også ha sluttet fra det at rangen + dimensjonen til nullrommet er lik antall søyler i matrisen, som er 4.

Exercise 3(a). Vi radreduserer

$$\begin{aligned} \begin{pmatrix} 2 & 8 & 6 & 1 \\ -4 & -16 & -12 & 2 \\ -1 & -4 & -3 & 1 \end{pmatrix} &\xrightarrow{I \leftrightarrow III} \begin{pmatrix} -1 & -4 & -3 & 1 \\ -4 & -16 & -12 & 2 \\ 2 & 8 & 6 & 1 \end{pmatrix} \\ &\xrightarrow{I \cdot (-1)} \begin{pmatrix} 1 & 4 & 3 & -1 \\ -4 & -16 & -12 & 2 \\ 2 & 8 & 6 & 1 \end{pmatrix} \\ &\xrightarrow{II+4I, III-2I} \begin{pmatrix} 1 & 4 & 3 & -1 \\ 0 & 0 & 0 & -2 \\ 0 & 0 & 0 & 3 \end{pmatrix} \\ &\xrightarrow{III/(-2)} \begin{pmatrix} 1 & 4 & 3 & -1 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 3 \end{pmatrix} \\ &\xrightarrow{III-3II, I+II} \begin{pmatrix} 1 & 4 & 3 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 \end{pmatrix} \end{aligned}$$

Vi ser at søyle 1 og 4 i A er pivotsøyler, slik at disse utgjør en basis for søylerommet til A .

Exercise 3(b). Vi setter inn en siste søyle som gjør at alle rader har et pivotelement, og radreduserer baklengs:

$$\begin{pmatrix} 1 & 4 & 3 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 \end{pmatrix} \xrightarrow{III+3II, I-II} \begin{pmatrix} 1 & 4 & 3 & -1 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 3 & 1 \end{pmatrix}$$

$$\xrightarrow{II(\sim 2)} \begin{pmatrix} 1 & 4 & 3 & -1 & 0 \\ 0 & 0 & 0 & -2 & 0 \\ 0 & 0 & 0 & 3 & 1 \end{pmatrix}$$

$$\xrightarrow{II-4I, III+2I} \begin{pmatrix} 1 & 4 & 3 & -1 & 0 \\ -4 & -16 & -12 & 2 & 0 \\ 2 & 8 & 6 & 1 & 1 \end{pmatrix}$$

$$\xrightarrow{I \cdot (-1)} \begin{pmatrix} -1 & -4 & -3 & 1 & 0 \\ -4 & -16 & -12 & 2 & 0 \\ 2 & 8 & 6 & 1 & 1 \end{pmatrix}$$

$$\xrightarrow{I \leftrightarrow III} \begin{pmatrix} 2 & 8 & 6 & 1 & 1 \\ -4 & -16 & -12 & 2 & 0 \\ -1 & -4 & -3 & 1 & 0 \end{pmatrix}$$

Vi kan derfor utvide med vektoren $(1, 0, 0)$.

Exercise 4. Gjør følgende:

1. Radreduser de n første radene i A til $n \times n$ identitetsmatrisen.
2. Null ut de $m - n$ siste radene ved hjelp av elementære radoperasjoner.
3. Sett inn $\mathbf{e}_{n+1}, \dots, \mathbf{e}_{n+m}$, slik at vi får $m \times m$ identitetsmatrisen.
4. Radreduser baklengs

Etter dette vil vi fortsatt ha nuller over standardbasisvektorene vi satt inn, siden de $m - n$ siste radene aldri vil påvirke de n første radene i operasjonene vi gjorde. Vektorene $\mathbf{e}_{n+1}, \dots, \mathbf{e}_{n+m}$ som vi satt inn blir heller ikke endret ved å legge til radene over til disse. Etter baklengs radreduksjon får vi derfor

$$\{\mathbf{a}_1, \dots, \mathbf{a}_n, \mathbf{e}_{n+1}, \dots, \mathbf{e}_m\},$$

som derfor også må være en basis for $\mathbb{R}^m$.

Exercise 6. La $\mathbf{x}, \mathbf{y} \in A^\perp$, og la $\mathbf{a} \in A$. Da er

$$(\mathbf{x} + \mathbf{y}) \cdot \mathbf{a} = \mathbf{x} \cdot \mathbf{a} + \mathbf{y} \cdot \mathbf{a} = 0 + 0 = 0$$

$$(c\mathbf{x}) \cdot \mathbf{a} = c(\mathbf{x} \cdot \mathbf{a}) = c \cdot 0 = 0.$$

Dette viser at både $\mathbf{x} + \mathbf{y} \in A^\perp$, og at $c\mathbf{x} \in A^\perp$. Det følger at $A^\perp$ er et underrom.

Exercise 8. Anta underrommet er $\text{Sp}(\mathbf{a}_1, \mathbf{a}_2, \dots, \mathbf{a}_n)$, der vektorene er lineært uavhengige. Utvid disse til en basis for $\mathbb{R}^m$, og la A være matrisen med disse som søyler. Da vet vi at A er inverterbar, og at $A\mathbf{e}_i = \mathbf{a}_i$ for alle i . Da er også $A^{-1}\mathbf{a}_i = \mathbf{e}_i$. Sett nå

$$C = \left(\begin{array}{c|c} \mathbf{0} & \mathbf{0} \\ \hline \mathbf{0} & I \end{array} \right) A^{-1},$$

der øvre venstre blokk er $n \times n$. Da er $C\mathbf{a}_1 = C\mathbf{a}_2 = \dots = C\mathbf{a}_n = \mathbf{0}$, slik at

$$\text{Sp}(\mathbf{a}_1, \mathbf{a}_2, \dots, \mathbf{a}_n) \subseteq \text{Nul}(C).$$

Vi har også at

$$C\mathbf{a}_{n+1} = \mathbf{e}_{n+1}, C\mathbf{a}_{n+2} = \mathbf{e}_{n+2}, \dots, C\mathbf{a}_m = \mathbf{e}_m$$

Hvis $\mathbf{x} \notin \text{Sp}(\mathbf{a}_1, \mathbf{a}_2, \dots, \mathbf{a}_n)$ så kan vi skrive

$$\mathbf{x} = c_1\mathbf{a}_1 + \dots + c_n\mathbf{a}_n + c_{n+1}\mathbf{a}_{n+1} + \dots + c_m\mathbf{a}_m,$$

der minst en av $c_{n+1}, \dots, c_m \neq 0$. Vi får nå at

$$C\mathbf{x} = c_{n+1}\mathbf{e}_{n+1} + \dots + c_m\mathbf{e}_m,$$

som da er $\neq \mathbf{0}$. Det følger at $\mathbf{x} \notin \text{Nul}(C)$, slik at $\text{Nul}(C) \subseteq \text{Sp}(\mathbf{v}_1, \mathbf{v}_2, \dots, \mathbf{v}_n)$. Det følger at $\text{Nul}(C) = \text{Sp}(\mathbf{v}_1, \mathbf{v}_2, \dots, \mathbf{v}_n)$

Exercise 9(a). Start med å omskrive (5.3) til

$$\begin{aligned} \mathbf{a}_{r+1} &= \sum_{i=1}^r \frac{\mathbf{a}_{r+1} \cdot \mathbf{q}_i}{\mathbf{q}_i \cdot \mathbf{q}_i} \mathbf{q}_i + \mathbf{q}_{r+1} \\ &= (\mathbf{q}_1 \quad \mathbf{q}_2 \quad \dots \quad \mathbf{q}_r \quad \mathbf{q}_{r+1}) \begin{pmatrix} \frac{\mathbf{a}_{r+1} \cdot \mathbf{q}_1}{\mathbf{q}_1 \cdot \mathbf{q}_1} \\ \frac{\mathbf{a}_{r+1} \cdot \mathbf{q}_2}{\mathbf{q}_2 \cdot \mathbf{q}_2} \\ \vdots \\ \frac{\mathbf{a}_{r+1} \cdot \mathbf{q}_r}{\mathbf{q}_r \cdot \mathbf{q}_r} \\ 1 \end{pmatrix} \\ &= (\mathbf{q}_1 \quad \mathbf{q}_2 \quad \dots \quad \mathbf{q}_r \quad \mathbf{q}_{r+1} \quad \mathbf{q}_{r+2} \quad \dots \quad \mathbf{q}_n) \begin{pmatrix} \frac{\mathbf{a}_{r+1} \cdot \mathbf{q}_1}{\mathbf{q}_1 \cdot \mathbf{q}_1} \\ \frac{\mathbf{a}_{r+1} \cdot \mathbf{q}_2}{\mathbf{q}_2 \cdot \mathbf{q}_2} \\ \vdots \\ \frac{\mathbf{a}_{r+1} \cdot \mathbf{q}_r}{\mathbf{q}_r \cdot \mathbf{q}_r} \\ 1 \\ 0 \\ \vdots \\ 0 \end{pmatrix}. \end{aligned}$$

Dette viser at søyle $r + 1$ i venstre og høyre side faller overens. Alle søylene er derfor like, slik at de to matrisene er like.

Exercise 9(b). Før forløkka fyller vi inn første rad i de to matrisene. Inne i forløkka fylles først inn rad $j - 1$ i den øvretriangulære matrisen R . Etter dette har vi i søyle j i R alle koeffisientene i (5.3), slik at vi kan regne ut $\mathbf{q}_j$. Dette gjøres i andre linja i forløkka. Resultatet blir fylt inn i rad j i matrisen Q .

Kapittel 6

Exercise 3. Definer avbildningen T ved at $T(\mathbf{y}) = \mathbf{x} \times \mathbf{y}$. Vi får at

$$\bullet T(\mathbf{e}_1) = \begin{vmatrix} \mathbf{e}_1 & \mathbf{e}_2 & \mathbf{e}_3 \\ x_1 & x_2 & x_3 \\ 1 & 0 & 0 \end{vmatrix} = \begin{vmatrix} \mathbf{e}_2 & \mathbf{e}_3 \\ x_2 & x_3 \end{vmatrix} = x_3 \mathbf{e}_2 - x_2 \mathbf{e}_3 = (0, x_3, -x_2)$$

$$\bullet T(\mathbf{e}_2) = \begin{vmatrix} \mathbf{e}_1 & \mathbf{e}_2 & \mathbf{e}_3 \\ x_1 & x_2 & x_3 \\ 0 & 1 & 0 \end{vmatrix} = - \begin{vmatrix} \mathbf{e}_1 & \mathbf{e}_3 \\ x_1 & x_3 \end{vmatrix} = -(x_3 \mathbf{e}_1 - x_1 \mathbf{e}_3) = (-x_3, 0, x_1)$$

$$\bullet T(\mathbf{e}_3) = \begin{vmatrix} \mathbf{e}_1 & \mathbf{e}_2 & \mathbf{e}_3 \\ x_1 & x_2 & x_3 \\ 0 & 0 & 1 \end{vmatrix} = \begin{vmatrix} \mathbf{e}_1 & \mathbf{e}_2 \\ x_1 & x_2 \end{vmatrix} = x_2 \mathbf{e}_1 - x_1 \mathbf{e}_2 = (x_2, -x_1, 0)$$

Matrisen til T blir dermed $\begin{pmatrix} 0 & -x_3 & x_2 \\ x_3 & 0 & -x_1 \\ -x_2 & x_1 & 0 \end{pmatrix}$.

Exercise 4. I stegene i Gausseliminasjon er det kun ett radbytte som bidrar til å endre determinanten, som dermed blir -1 . Utvikler vi determinanten langs første søyle får vi

$$\begin{vmatrix} 2 & -1 & 1 \\ 0 & 1 & 4 \\ 1 & -2 & -6 \end{vmatrix} = 2(-6 + 8) - 0 + (-4 - 1) = 4 - 5 = -1.$$

Exercise 6. Gjør først $i - 1$ radbytter slik at rad i havner øverst. Bruk deretter (6.1). Fortegnet blir $(-1)^{i+j}$.

Exercise 7. Vi viser først del 2. Hvis det er den første raden som ganges med s så følger det fra (6.1)

$$\begin{aligned} \det(B) &= s a_{11} \det(A^{(11)}) + s a_{12} \det(A^{(12)}) + \cdots + s a_{1n} \det(A^{(1n)}) \\ &= s (a_{11} \det(A^{(11)}) + a_{12} \det(A^{(12)}) + \cdots + a_{1n} \det(A^{(1n)})) = s \det(A). \end{aligned}$$

Hvis en av de andre radene ganges med s så slår vi sammen tre operasjoner: Først et radbytte med rad 1, deretter ganger vi rad 1 med s , og til slutt gjør vi samme radbytte en gang til. Proposisjon 6.5 gir da

$$\det(B) = -(s(-\det(A))) = s\det(A).$$

La oss så vise del 3. La C være matrisen vi får ved å bytte om rad 1 og rad j i B . Første rad i C har da elementer $a_{jk} + sa_{ik}$, slik at (6.1) gir

$$\begin{aligned} -\det(B) &= \det(C) = \sum_{k=1}^n (-1)^{k+1} (a_{jk} + sa_{ik}) \det(C_{1k}) \\ &= \sum_{k=1}^n (-1)^{k+1} a_{jk} \det(C_{1k}) + s \sum_{k=1}^n (-1)^{k+1} a_{ik} \det(C_{1k}) \\ &= \sum_{k=1}^n (-1)^{k+1} a_{jk} \det(C_{1k}), \end{aligned}$$

der den andre summen ble null siden rad 1 og i er like (Korollar 6.6.). Det siste uttrykket er determinanten til matrisen som fremkommer ved å bytte rad 1 og j i A , og denne er $-\det(A)$. Det følger at $\det(A) = \det(B)$.

Exercise 8. Siden $1 = \det(I_n) = \det(A^{-1}A) = \det(A^{-1})\det(A)$, så følger at $\det(A^{-1}) = 1/\det(A)$.

Exercise 10. Vi har at

$$\det(A) = \det(A^T) = \sum_{j=1}^n (-1)^{j+i} a_{ji} \det(A^{(ji)}) = \sum_{i=1}^n (-1)^{i+j} a_{ij} \det(A^{(ij)}).$$

der vi utviklet determinanten til A^T langs rad i .

Exercise 12(c). Den inverse matrisen til A kan fås ved å løse $A\mathbf{x} = \mathbf{e}_1$, og $A\mathbf{x} = \mathbf{e}_2$. Cramer's regel gir at $A\mathbf{x} = \mathbf{e}_1$ har løsning

$$\left(\frac{\begin{vmatrix} 1 & b \\ 0 & d \end{vmatrix}}{\begin{vmatrix} a & b \\ c & d \end{vmatrix}}, \frac{\begin{vmatrix} a & 1 \\ c & 0 \end{vmatrix}}{\begin{vmatrix} a & b \\ c & d \end{vmatrix}} \right) = \frac{1}{ad-bc} \begin{pmatrix} d \\ -c \end{pmatrix},$$

og at $A\mathbf{x} = \mathbf{e}_2$ har løsning

$$\left(\frac{\begin{vmatrix} 0 & b \\ 1 & d \end{vmatrix}}{\begin{vmatrix} a & b \\ c & d \end{vmatrix}}, \frac{\begin{vmatrix} a & 0 \\ c & 1 \end{vmatrix}}{\begin{vmatrix} a & b \\ c & d \end{vmatrix}} \right) = \frac{1}{ad-bc} \begin{pmatrix} -b \\ a \end{pmatrix}.$$

Den inverse matrisen blir dermed $\frac{1}{ad-bc} \begin{pmatrix} d & -b \\ -c & a \end{pmatrix}$.

Kapittel 7

Exercise 1 (a). Det karakteristiske polynomet er

$$(0.7 - \lambda)(0.4 - \lambda) - 0.3 \cdot 0.6 = \lambda^2 - 1.1\lambda + 0.1,$$

som har røtter $\lambda = \frac{1.1 \pm \sqrt{1.21 - 0.4}}{2} = \frac{1.1 \pm 0.9}{2}$, slik at $\lambda_1 = 0.1$, $\lambda_2 = 1$.

Eigenverdi for $\lambda_1 = 0.1$:

$$A - 0.1I_2 = \begin{pmatrix} 0.6 & 0.6 \\ 0.3 & 0.3 \end{pmatrix} \sim \begin{pmatrix} 1 & 1 \\ 0 & 0 \end{pmatrix},$$

slik at $(-x_2, x_2)$ er en generell egenvektor. En spesiell egenvektor er $\mathbf{v}_1 = (-1, 1)$.

Eigenvektor for $\lambda_2 = 1$:

$$A - I_2 = \begin{pmatrix} -0.3 & 0.6 \\ 0.3 & -0.6 \end{pmatrix} \sim \begin{pmatrix} 1 & -1 \\ 0 & 0 \end{pmatrix},$$

slik at (x_2, x_2) er en generell egenvektor. En spesiell egenvektor er $\mathbf{v}_2 = (1, 1)$.

Exercise 1 (b). Vi radreduserer

$$\begin{pmatrix} -1 & 1 & 1 \\ 1 & 1 & 5 \end{pmatrix} \sim \begin{pmatrix} 1 & 1 & 5 \\ 0 & 2 & 6 \end{pmatrix} \sim \begin{pmatrix} 1 & 0 & 2 \\ 0 & 1 & 3 \end{pmatrix},$$

slik at $\mathbf{x}_0 = 2\mathbf{v}_1 + 3\mathbf{v}_2$. Dette gir

$$\lim_{n \rightarrow \infty} A^n \mathbf{x}_0 = \lim_{n \rightarrow \infty} (2A^n \mathbf{v}_1 + 3A^n \mathbf{v}_2) = \lim_{n \rightarrow \infty} (2 \cdot 0.1^n \mathbf{v}_1 + 3\mathbf{v}_2) = 3\mathbf{v}_2 = (3, 3).$$

Exercise 2. Vi har

$$\det(A - \lambda I) = (1 - \lambda) \begin{vmatrix} 3 - \lambda & 1 \\ 1 & 3 - \lambda \end{vmatrix} = (1 - \lambda)((3 - \lambda)^2 - 1)$$

der vi utviklet determinanten langs første rad. Siden $(1 - \lambda)$ ble en faktor her så trenger vi ikke bruke polynomdivisjon. Vi trenger heller ikke gange ut $(3 - \lambda)^2 - 1$, siden vi ser at denne er null kun når $\lambda = 3 \pm 1$. Eigenverdiene er derfor $\lambda_1 = 1$, $\lambda_2 = 2$, $\lambda_3 = 4$.

Eigenvektor for $\lambda_1 = 1$:

$$A - I_3 = \begin{pmatrix} 0 & 0 & 0 \\ 0 & 2 & 1 \\ 0 & 1 & 2 \end{pmatrix} \sim \begin{pmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ 0 & 0 & 0 \end{pmatrix},$$

slik at en egenvektor blir $(1, 0, 0)$.

Egenvektor for $\lambda_2 = 2$:

$$A - 2I_3 = \begin{pmatrix} -1 & 0 & 0 \\ 0 & 1 & 1 \\ 0 & 1 & 1 \end{pmatrix} \sim \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 1 \\ 0 & 0 & 0 \end{pmatrix},$$

slik at en egenvektor blir $(0, -1, 1)$.

Egenvektor for $\lambda_3 = 4$:

$$A - 4I_3 = \begin{pmatrix} -3 & 0 & 0 \\ 0 & -1 & 1 \\ 0 & 1 & -1 \end{pmatrix} \sim \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & -1 \\ 0 & 0 & 0 \end{pmatrix},$$

slik at en egenvektor blir $(0, 1, 1)$.

Exercise 3. Vi har $\det(A - \lambda I) = \lambda^2 + 4$, og denne er 0 for $\lambda = \pm 2i$.

Egenvektor for $\lambda_1 = 2i$:

$$A - 2iI_2 = \begin{pmatrix} -2i & -4 \\ 1 & -2i \end{pmatrix} \sim \begin{pmatrix} 1 & -2i \\ 0 & 0 \end{pmatrix},$$

slik at $(2i, 1)$ er en spesiell egenvektor. Men da er $(-2i, 1)$ en spesiell egenvektor for $\lambda_2 = -2i$.

Exercise 4. Vi har at $\det(A - \lambda I_2) = (6 - \lambda)^2$, slik at $\lambda = 6$ er eneste egenverdi.

Vi finner egenvektorene:

$$A - 6I_2 = \begin{pmatrix} 0 & 2 \\ 0 & 0 \end{pmatrix} \sim \begin{pmatrix} 0 & 1 \\ 0 & 0 \end{pmatrix},$$

slik at $(1, 0)$ er en spesiell egenvektor. Denne vektoren utspenner et rom med dimensjon 1, slik at vi ikke kan ha en basis for $\mathbb{R}^2$ som består av egenvektorer for A .

Exercise 8. Anta $A\mathbf{v} = \lambda_1\mathbf{v}$ og $B\mathbf{v} = \lambda_2\mathbf{v}$. Da er

$$(A + B)\mathbf{v} = A\mathbf{v} + B\mathbf{v} = \lambda_1\mathbf{v} + \lambda_2\mathbf{v} = (\lambda_1 + \lambda_2)\mathbf{v}.$$

Derfor er $\mathbf{v}$ også egenvektor for $A + B$, med $\lambda_1 + \lambda_2$ som tilhørende egenverdi.

Kapittel 8

Exercise 1(a). Vi har at

$$\det(\lambda I - A) = (\lambda - 0.7)(\lambda - 0.2) - 0.24 = \lambda^2 - 0.9\lambda - 0.1.$$

Denne er 0 når $\lambda = \frac{0.9 \pm 1.1}{2}$. Derfor blir egenverdiene $\lambda_1 = -0.1$ og $\lambda_2 = 1$.

Egenvektor for $\lambda_1 = -0.1$:

$$-0.1I - A = \begin{pmatrix} -0.8 & -0.8 \\ -0.3 & -0.3 \end{pmatrix} \sim \begin{pmatrix} 1 & 1 \\ 0 & 0 \end{pmatrix}.$$

En spesiell egenvektor blir $\mathbf{v}_1 = \begin{pmatrix} -1 \\ 1 \end{pmatrix}$

Egenvektor for $\lambda_2 = 1$:

$$I - A = \begin{pmatrix} 0.3 & -0.8 \\ -0.3 & 0.8 \end{pmatrix} \sim \begin{pmatrix} 1 & -8/3 \\ 0 & 0 \end{pmatrix}$$

En spesiell egenvektor blir $\mathbf{v}_2 = \begin{pmatrix} 8 \\ 3 \end{pmatrix}$

Exercise 1(b). En matrise A er diagonaliserbar hvis den kan skrives på formen $A = PDP^{-1}$, der D er diagonal og P er inverterbar. P inneholder da egenvektorene som søyler, og D har egenverdiene på diagonalen.

Vi vet fra læreboka at egenvektorer tilhørende forskjellige egenverdier er lineært uavhengige. Siden vi her har to forskjellige egenverdier så har vi også to lineært uavhengige egenvektorer. Vi kan da få P ved å sette inn de to egenvektorene som søyler, med de tilhørende egenverdiene på diagonalen i D . Vi har derfor $A = PDP^{-1}$ med $P = \begin{pmatrix} -1 & 8 \\ 1 & 3 \end{pmatrix}$ og $D = \begin{pmatrix} -0.1 & 0 \\ 0 & 1 \end{pmatrix}$, og

$$P^{-1} = \frac{1}{\det(P)} \begin{pmatrix} 3 & -8 \\ -1 & -1 \end{pmatrix} = \frac{1}{11} \begin{pmatrix} -3 & 8 \\ 1 & 1 \end{pmatrix}.$$

Det er raskt å sjekke at egenvektorene $(-1, 1)$ og $(8, 3)$ ikke er ortogonale, slik at vi ikke har en ortonormal basis av egenvektorer. Her kan man også argumentere som følger: Hvis A har en ortonormal basis av egenvektorer, så kan den skrives på formen $A = PDP^T$, der P har egenvektorene som søyler. Men da er A symmetrisk siden

$$A^T = (PDP^T)^T = PD^T P^T = PDP^T = A.$$

Matrisen her var ikke symmetrisk, og kan da heller ikke ha en ortonormal basis av egenvektorer.

Exercise 2. Se på vektoren $\left(\frac{c}{\sqrt{c^2+d^2}}, \frac{d}{\sqrt{c^2+d^2}}\right)$. Det er lett å verifisere at denne har lengde 1. Det finnes da en vinkel θ slik at

$$\left(\frac{c}{\sqrt{c^2+d^2}}, \frac{d}{\sqrt{c^2+d^2}}\right) = (\cos\theta, \sin\theta).$$

Men da er $c = \sqrt{c^2+d^2} \cos\theta$ og $d = \sqrt{c^2+d^2} \sin\theta$, slik at

$$\begin{aligned} |c \cos(bt) - d \sin(bt)| &= |\sqrt{c^2+d^2} \cos\theta \cos(bt) - \sqrt{c^2+d^2} \sin\theta \sin(bt)| \\ &= \sqrt{c^2+d^2} |\cos\theta \cos(bt) - \sin\theta \sin(bt)| \\ &= \sqrt{c^2+d^2} |\cos\theta \cos(bt+\theta)| \leq \sqrt{c^2+d^2}, \end{aligned}$$

siden $0 \leq |\cos\theta| \leq 1$ for alle θ . Her brukte vi også formelen for cosinus til en sum av to vinkler.

Exercise 3(a). Vi har her $A = \begin{pmatrix} 1 & 1 & 0 \\ -1 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix}$. Vi får at

$$A - \lambda I = \begin{pmatrix} 1-\lambda & 1 & 0 \\ -1 & 1-\lambda & 0 \\ 0 & 0 & 1-\lambda \end{pmatrix}.$$

Determinanten til denne blir $(1-\lambda)((1-\lambda)^2 + 1)$, slik at egenverdiene blir $1 \pm i$, og 1. Egenvektorer blir

$$\mathbf{p}_1 = \begin{pmatrix} 1 \\ i \\ 0 \end{pmatrix} \quad \mathbf{p}_2 = \begin{pmatrix} 1 \\ -i \\ 0 \end{pmatrix} \quad \mathbf{p}_3 = \begin{pmatrix} 0 \\ 0 \\ 1 \end{pmatrix},$$

slik at den generelle løsningen blir

$$C e^{(1+i)t} \begin{pmatrix} 1 \\ i \\ 0 \end{pmatrix} + D e^{(1-i)t} \begin{pmatrix} 1 \\ -i \\ 0 \end{pmatrix} + E e^t \begin{pmatrix} 0 \\ 0 \\ 1 \end{pmatrix}.$$

Exercise 3(b). Initialbetingelsen gir likningen

$$\begin{pmatrix} 1 \\ 1 \\ 1 \end{pmatrix} = \begin{pmatrix} 1 & 1 & 0 \\ i & -i & 0 \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} C \\ D \\ E \end{pmatrix},$$

som har løsningen $C = (1 - i)/2$, $D = (1 + i)/2 = \overline{C}$, $E = 1$. Dette gir

$$\begin{aligned}
 & Ce^{(1+i)t} \begin{pmatrix} 1 \\ i \\ 0 \end{pmatrix} + De^{(1-i)t} \begin{pmatrix} 1 \\ -i \\ 0 \end{pmatrix} + Ee^t \begin{pmatrix} 0 \\ 0 \\ 1 \end{pmatrix} \\
 &= \frac{1-i}{2} e^{(1+i)t} \begin{pmatrix} 1 \\ i \\ 0 \end{pmatrix} + \frac{\overline{1-i}}{2} e^{(1-i)t} \overline{\begin{pmatrix} 1 \\ i \\ 0 \end{pmatrix}} + e^t \begin{pmatrix} 0 \\ 0 \\ 1 \end{pmatrix} \\
 &= 2\Re \left(\frac{1-i}{2} e^{(1+i)t} \begin{pmatrix} 1 \\ i \\ 0 \end{pmatrix} \right) + e^t \begin{pmatrix} 0 \\ 0 \\ 1 \end{pmatrix} \\
 &= e^t \Re \left((\cos t + i \sin t) \left(\begin{pmatrix} 1 \\ 1 \\ 0 \end{pmatrix} + i \begin{pmatrix} -1 \\ 1 \\ 0 \end{pmatrix} \right) \right) + e^t \begin{pmatrix} 0 \\ 0 \\ 1 \end{pmatrix} \\
 &= e^t \left(\begin{pmatrix} \cos t \\ \cos t \end{pmatrix} + \begin{pmatrix} \sin t \\ -\sin t \end{pmatrix} + \begin{pmatrix} 0 \\ 0 \\ 1 \end{pmatrix} \right) \\
 &= e^t \begin{pmatrix} \cos t + \sin t \\ \cos t - \sin t \\ 1 \end{pmatrix},
 \end{aligned}$$

der vi regnet ut

$$\frac{1-i}{2} \begin{pmatrix} 1 \\ i \\ 0 \end{pmatrix} = \begin{pmatrix} 1/2 \\ 1/2 \\ 0 \end{pmatrix} + i \begin{pmatrix} -1/2 \\ 1/2 \\ 0 \end{pmatrix}$$

Exercise 4(a). $y(t) = e^t + e^{4t}$.

Exercise 4(b). $e^{-t}(c \cos t + d \sin t)$.

Exercise 4(c). $y(t) = 3e^{2t} + 2te^{2t}$.

Exercise 5(a). Vi har at

$$y'' + py' + qy = (\lambda^2 + p\lambda + q)e^{\lambda t} = 0,$$

siden λ tilfredsstiller den karakteristiske likningen.

Exercise 5(b). Vi får først at

$$y' = (1 + \lambda t)e^{\lambda t} \qquad y'' = (2\lambda + \lambda^2 t)e^{\lambda t},$$

og deretter

$$\begin{aligned} y'' + py' + qy &= (2\lambda + \lambda^2 t + p(1 + \lambda t) + qt) e^{\lambda t} \\ &= \left(-p + \frac{p^2}{4}t + p\left(1 - \frac{p}{2}t\right) + \frac{p^2}{4}t\right) e^{\lambda t} \\ &= \left(-p + \frac{p^2}{4}t + p - \frac{p^2}{2}t + \frac{p^2}{4}t\right) e^{\lambda t} = 0. \end{aligned}$$

Exercise 5(c). Vi har at

$$y(t_0) = Ce^{\lambda t_0} + Dt_0 e^{\lambda t_0} \quad y'(t_0) = C\lambda e^{\lambda t_0} + D(\lambda t_0 + 1)e^{\lambda t_0},$$

som kan skrives som

$$e^{\lambda t_0} \begin{pmatrix} 1 & t_0 \\ \lambda & \lambda t_0 + 1 \end{pmatrix} \begin{pmatrix} C \\ D \end{pmatrix} = \begin{pmatrix} y(t_0) \\ y'(t_0) \end{pmatrix}.$$

Det er lett å se at matrisen her er inverterbar, slik at vi har et unikt valg for C og D .

Exercise 6(a). Vi har at

$$Q_k^2 = P_{:k}(P^{-1})_{k:} P_{:k}(P^{-1})_{k:} = P_{:k}(P^{-1})_{k:} = Q_k$$

der vi brukte at $P^{-1}P = I$.

Exercise 6(b). Vi har at $Q_r Q_s = P_{:r}(P^{-1})_{r:} P_{:s}(P^{-1})_{s:} = 0$ når $r \neq s$, siden $P^{-1}P = 0$ utenfor diagonalen.

Exercise 6(d). Det er klart at

$$p(A) = P \begin{pmatrix} p(\lambda_1) & 0 & \cdots & 0 \\ 0 & p(\lambda_2) & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & p(\lambda_n) \end{pmatrix} P^{-1}$$

for alle polynomer p . Velg nå et polynom p som er 1 i λ_i , og 0 i de andre egenverdiene. Da får vi at $p(A) = \lambda_i \sum Q_k$.

Kapittel 9

Exercise 1. Den karakteristiske likningen blir $(14 - \lambda)(11 - \lambda) - 4 = 0$, som gir $\lambda^2 - 25\lambda + 150 = 0$. Egenverdiene blir $\frac{25 \pm \sqrt{625 - 600}}{2} = \frac{25 \pm 5}{2}$, som gir $\lambda_1 = 10$ og $\lambda_2 = 15$.

Egenvektor for $\lambda_1 = 10$:

$$\begin{pmatrix} 14 & -2 \\ -2 & 11 \end{pmatrix} - 10I = \begin{pmatrix} 4 & -2 \\ -2 & 1 \end{pmatrix} \sim \begin{pmatrix} 1 & -1/2 \\ 0 & 0 \end{pmatrix}$$

En spesiell egenvektor blir $(1, 2)$. En egenvektor med lengde 1 blir $(1, 2)/\sqrt{5}$.

Egenvektor for $\lambda_2 = 15$:

$$\begin{pmatrix} 14 & -2 \\ -2 & 11 \end{pmatrix} - 15I = \begin{pmatrix} -1 & -2 \\ -2 & -4 \end{pmatrix} \sim \begin{pmatrix} 1 & 2 \\ 0 & 0 \end{pmatrix}$$

En spesiell egenvektor blir $(-2, 1)$. En egenvektor med lengde 1 blir $(-2, 1)/\sqrt{5}$.

En ortogonal diagonalisering blir dermed

$$A = \frac{1}{\sqrt{5}} \begin{pmatrix} 1 & -2 \\ 2 & 1 \end{pmatrix} \begin{pmatrix} 10 & 0 \\ 0 & 15 \end{pmatrix} \left(\frac{1}{\sqrt{5}} \begin{pmatrix} 1 & -2 \\ 2 & 1 \end{pmatrix} \right)^T.$$

Exercise 4(a). Vi har at

$$\det(\lambda I - A) = (\lambda - (1 + \epsilon)/2)^2 - (1 - \epsilon)^2/4 = \lambda^2 - (1 + \epsilon)\lambda + \epsilon.$$

Denne har røtter

$$\frac{1 + \epsilon \pm \sqrt{(1 + \epsilon)^2 - 4\epsilon}}{2} = \frac{1 + \epsilon \pm (1 - \epsilon)}{2}.$$

Den største egenverdien er $\lambda_1 = 1$, den minste er $\lambda_2 = \epsilon$. La oss finne egenvektorene.

- En egenvektor for $\lambda_1 = 1$:

$$I - A = \begin{pmatrix} 1 - (1 + \epsilon)/2 & -(1 - \epsilon)/2 \\ -(1 - \epsilon)/2 & 1 - (1 + \epsilon)/2 \end{pmatrix} = \begin{pmatrix} (1 - \epsilon)/2 & -(1 - \epsilon)/2 \\ -(1 - \epsilon)/2 & (1 - \epsilon)/2 \end{pmatrix} \sim \begin{pmatrix} 1 & -1 \\ 0 & 0 \end{pmatrix},$$

slik at vi kan sette $\mathbf{v}_1 = (1, 1)$.

- En egenvektor for $\lambda_2 = \epsilon$:

$$\epsilon I - A = \begin{pmatrix} \epsilon - (1 + \epsilon)/2 & -(1 - \epsilon)/2 \\ -(1 - \epsilon)/2 & \epsilon - (1 + \epsilon)/2 \end{pmatrix} = \begin{pmatrix} -(1 - \epsilon)/2 & -(1 - \epsilon)/2 \\ -(1 - \epsilon)/2 & -(1 - \epsilon)/2 \end{pmatrix} \sim \begin{pmatrix} 1 & 1 \\ 0 & 0 \end{pmatrix},$$

slik at vi kan sette $\mathbf{v}_2 = (1, -1)$.

Exercise 4(b). Vi har at $\mathbf{y} = \mathbf{v}_1 + 10^{-3}\epsilon^{-1}\mathbf{v}_2$, slik at

$$A\mathbf{y} = A\mathbf{v}_1 + 10^{-3}\epsilon^{-1}A\mathbf{v}_2 = \mathbf{v}_1 + 10^{-3}\mathbf{v}_2 = \mathbf{b} + \mathbf{e}.$$

Exercise 5(a). Vi har at

$$\det(\lambda I - A) = (\lambda - 0.2)^2 - 0.64,$$

og dette er 0 kun $\lambda - 0.2 = \pm 0.8$. Derfor blir egenverdiene $\lambda_1 = -0.6$ og $\lambda_2 = 1$.

Egenvektor for $\lambda_1 = -0.6$:

$$-0.6I - A = \begin{pmatrix} -0.8 & -0.8 \\ -0.8 & -0.8 \end{pmatrix} \sim \begin{pmatrix} 1 & 1 \\ 0 & 0 \end{pmatrix}.$$

En spesiell egenvektor blir $\mathbf{v}_1 = \begin{pmatrix} -1 \\ 1 \end{pmatrix}$

Egenvektor for $\lambda_2 = 1$:

$$I - A = \begin{pmatrix} 0.8 & -0.8 \\ -0.8 & 0.8 \end{pmatrix} \sim \begin{pmatrix} 1 & -1 \\ 0 & 0 \end{pmatrix}$$

En spesiell egenvektor blir $\mathbf{v}_2 = \begin{pmatrix} 1 \\ 1 \end{pmatrix}$

Exercise 5(b). Matrisen er symmetrisk. Den er også inverterbar, siden begge egenverdiene er $\neq 0$. Kondisjonstallet er derfor forholdet mellom største og minste egenverdi i absoluttverdi, som gir $K(A) = 1/0.6 = 5/3$.

Exercise 6. Siden A er inverterbar er alle $\lambda_i \neq 0$. Hvis vi transponerer begge sider i $A^{-1}A = I$ får vi at $A^T(A^{-1})^T = I$. Siden A er symmetrisk betyr dette at $A(A^{-1})^T = I$, slik at $A^{-1} = (A^{-1})^T$. Det følger at A^{-1} også er symmetrisk.

Anta $A\mathbf{v} = \lambda\mathbf{v}$. Da må $\mathbf{v} = A^{-1}(\lambda\mathbf{v})$, slik at $A^{-1}\mathbf{v} = \frac{1}{\lambda}\mathbf{v}$. Det følger at $1/\lambda$ er en egenverdi for A^{-1} når λ er en egenverdi for A . A og A^{-1} har også de samme egenvektorene.

Dette følger også lett fra at A er ortogonalt diagonaliserbar, slik at vi kan skrive $A^P D P^{-1}$, der D er diagonalmatrisen med egenverdiene λ_i på diagonalen. Fra dette følger at $A^{-1} = P D^{-1} P^T$, der D^{-1} er diagonalmatrisen med $1/\lambda_i$ på diagonalen.

Exercise 7. Det er klart at

$$|U\mathbf{x}|^2 = \mathbf{x}^T U^T U \mathbf{x} = |\mathbf{x}|^2.$$

Det følger at, hvis $\mathbf{x}$ er en egenvektor, så er $|U\mathbf{x}| = |\lambda||\mathbf{x}| = |\mathbf{x}|$, slik at $|\lambda| = 1$. Siden alle egenverdiene er like i absoluttverdi følger det at $K(A) = 1$. Merk at vi ikke trenger å gå via ortogonal diagonaliserbarhet her.

Exercise 8(a). En linje i et slikt plott betyr at $y = a \log x + b$, der a og b er konstanter vi kan lese ut fra linja (a er stigningstallet, b er skjæringen med y -aksen). $y = a \log x + b$ gir at $x = e^{(y-b)/a} = e^{-b/a} e^{y/a}$, slik at x er eksponensiell i y med konstant $K = e^{-b/a}$ og eksponent $1/a$.

Exercise 8(b). En linje i et slikt plott betyr at $\log y = ax + b$. Dette gir at $y = e^b e^{ax}$, slik at y er eksponensiell i x med konstant $K = e^b$ og eksponent a .

Kapittel 10

Exercise 1(a). Sant

Exercise 1(b). Usant

Exercise 1(c). Usant

Exercise 2. Hvis vi antar at datamaskinen regner ut x^n ved å gange inn x $n-1$ ganger, så vil det første kodesegmentet bruke $n(1 + \dots + n-2) = n(n-1)(n-2)/2$ operasjoner, mens det andre kodesegmentet bruker $n^2 - n$ operasjoner, og derfor er å foretrekke.

Det er forøvrig ikke sikkert at datamaskinen regner ut x^n ved hjelp av $n-1$ multiplikasjoner. Sannsynligvis så bruker maskinen en mer effektiv algoritme. Legg spesielt merke til at $x^n = e^{n \ln x}$, som maskinen kan utnytte til å regne ut potenser såfremt den har effektive algoritmer for å regne ut logaritmer og eksponensialfunksjonen.

Exercise 3(a). Siden $p_1(x_0) = p_2(x_0) = f(x_0)$, $p_1(x_1) = p_2(x_1) = f(x_1)$, $p_1(x_2) = p_2(x_2) = f(x_2)$, så får vi at

$$p(x_0) = p_2(x_0) - p_1(x_0) = f(x_0) - f(x_0) = 0$$

$$p(x_1) = p_2(x_1) - p_1(x_1) = f(x_1) - f(x_1) = 0$$

$$p(x_2) = p_2(x_2) - p_1(x_2) = f(x_2) - f(x_2) = 0.$$

Derfor er verdien i interpolasjonspunktene 0.

Exercise 3(b). Andregradspolynomet p har tre forskjellige nuller, men et andregradspolynom forskjellig fra 0 kan ikke ha mer enn to nuller. Vi må derfor ha at $p = 0$, slik at $p_1 = p_2$, slik at det interpolerende polynomet er unikt.

Exercise 3(c). Hvis vi definerer $p = p_1 - p_2$ som over, så ser vi som før at $p(x_0) = p(x_1) = \dots = p(x_n) = 0$, slik at p er et polynom av grad n med $n+1$ forskjellige nuller. Fra algebraens fundamentalteorem (d.v.s. at et polynom av grad n ikke kan ha mer enn n røtter) følger det at $p = 0$, slik at $p_1 = p_2$, slik at det interpolerende polynomet er unikt også i dette tilfellet.

Exercise 4(a). Hvis vi kaller beregningene av høyresidene for s , og starter med siste likning, så blir stegene

$$\begin{aligned} s &= c_n \\ s &= c_{n-1} + xs \\ \vdots &\quad \quad \quad \vdots \\ s &= c_2 + xs \\ s &= c_1 + xs \\ s &= c_0 + xs. \end{aligned}$$

Etter dette får vi at $s = p_n(x)$. Koden kan se slik ut

```
s = c[n]
for k in range(n-1,-1,-1):
    s = c[k] + x*s
```

Exercise 4(b). I hver iterasjon av løkka gjøres en addisjon og en multiplikasjon. Det totale antallet er derfor $2n$ operasjoner, som er en forbedring fra $3n - 1$. Horner's regel er faktisk en optimal metode når det kommer til å regne ut verdier av polynomet. Den er også optimal når det kommer til minnebruk — man trenger bare variabelen s i tillegg til x og c . Skal du integrere den i koden i Program 10.1 så skal du bytte ut `x_plot_powers*c` med det du får fra koden over.

Exercise 5. 16

Exercise 6(a). Setter vi $x = 0$ får vi at $1 = c_0(-1)(-3)(-4)$, slik at $c_0 = -1/12$.
 Setter vi $x = 1$ får vi at $0 = c_1(1)(1-3)(1-4)$, slik at $c_1 = 0$.
 Setter vi $x = 3$ får vi at $2 = c_2(3)(3-1)(3-4)$, slik at $c_2 = -1/3$.
 Setter vi $x = 4$ får vi at $1 = c_3(4)(4-1)(4-3)$, slik at $c_3 = 1/12$.
 Vi får derfor

$$p_3(x) = -\frac{(x-1)(x-3)(x-4)}{12} - \frac{x(x-1)(x-4)}{3} + \frac{x(x-1)(x-3)}{12}.$$

Exercise 6(b). Setter vi $p_3(x) = c_0 + c_1x + c_2x(x-1) + c_3x(x-1)(x-3)$, så får vi likningene

$$\begin{aligned} 1 &= c_0 \\ 0 &= c_0 + c_1 \\ 2 &= c_0 + 3c_1 + 6c_2 \\ 1 &= c_0 + 4c_1 + 12c_2 + 12c_3. \end{aligned}$$

De første to likningene gir at $c_0 = 1$ og $c_1 = -1$. Setter vi dette inn i den tredje likningen får vi at $c_2 = (2 - 1 + 3)/6 = 2/3$. Setter vi dette inn i den fjerde likningen får vi at $c_3 = (1 - 1 + 4 - 8)/12 = -1/3$. Dette gir

$$p_3(x) = 1 - x + \frac{2}{3}x(x-1) - \frac{1}{3}x(x-1)(x-3).$$

Exercise 7(a). Newtonformen til det interpolerende andregangspolynomet er

$$p_2(x) = f(x_0) + f[x_0, x_1](x - x_0) + f[x_0, x_1, x_2](x - x_0)(x - x_1).$$

Vi har at

$$\begin{aligned} f[x_0, x_1] &= \frac{f(x_1) - f(x_0)}{x_1 - x_0} = \frac{1 - 2}{1 - 0} = -1 \\ f[x_1, x_2] &= \frac{f(x_2) - f(x_1)}{x_2 - x_1} = \frac{0 - 1}{2 - 1} = -1 \\ f[x_0, x_1, x_2] &= \frac{f[x_1, x_2] - f[x_0, x_1]}{x_2 - x_0} = \frac{-1 - (-1)}{2 - 0} = 0. \end{aligned}$$

Setter vi dette inn i $p_2(x)$ får vi at $p_2(x) = 2 - (x - x_0) = 2 - x$.

Exercise 7(b). Både f og p_n er interpolerende polynomer av grad n , og da vet vi fra 3 at de må være like. I (a) var funksjonen allerede et andregradspolynom, slik at den må være lik dens interpolant også. Derfor fikk vi at p_n var lik funksjonen selv.

Exercise 8(a). Lineær interpolant p_1 :

$$p_1(x) = y_1 + (y_2 - y_1)(x - 1).$$

Dette gir $p_1(3/2) = y_1 + \frac{1}{2}(y_2 - y_1) = \frac{1}{2}(y_1 + y_2)$.

Exercise 8(b). Newtonformen for polynomet som interpolerer i de 4 punktene kan vi finne ved å løse

$$\begin{aligned} y_0 &= c_0 \\ y_1 &= c_0 + c_1 \\ y_2 &= c_0 + 2c_1 + 2c_2 \\ y_3 &= c_0 + 3c_1 + 6c_2 + 6c_3. \end{aligned}$$

Dette gir først $c_0 = y_0$, så $c_1 = y_1 - y_0$, så

$$c_2 = \frac{y_2 - y_0 - 2(y_1 - y_0)}{2} = \frac{y_2 - 2y_1 + y_0}{2}$$

$$c_3 = \frac{y_3 - y_0 - 3(y_1 - y_0) - 6(y_2 - 2y_1 + y_0)/2}{6} = \frac{y_3 - 3y_2 + 3y_1 - y_0}{6},$$

slik at det interpolerende kubiske polynomet blir

$$p_3(x) = y_0 + (y_1 - y_0)x + \frac{y_2 - 2y_1 + y_0}{2}x(x-1) + \frac{y_3 - 3y_2 + 3y_1 - y_0}{6}x(x-1)(x-2).$$

Dette gir til slutt

$$p_3(3/2) = y_0 + (y_1 - y_0)\frac{3}{2} + \frac{y_2 - 2y_1 + y_0}{2}\frac{3}{4} - \frac{y_3 - 3y_2 + 3y_1 - y_0}{6}\frac{3}{8}$$

$$= \frac{16y_0 + 24y_1 - 24y_0 + 6y_2 - 12y_1 + 6y_0 - y_3 + 3y_2 - 3y_1 + y_0}{16}$$

$$= \frac{-y_0 + 9y_1 + 9y_2 - y_3}{16}.$$

Exercise 9. I den første løkka regner vi ut koeffisientmatrisen A , som vi trenger for å finne newtonformen. i den neste løkka gjør vi fremoversubstitusjon for å finne newtonformen. Hver av disse to løkkene trenger omtrent n^2 operasjoner, slik at vi totalt får $2n^2$ operasjoner.

Exercise 10(a). La oss prøve med et kubisk polynom på formen

$$p_3(x) = c_0 + c_1(x - x_0) + c_2(x - x_0)(x - x_1) + c_3(x - x_0)(x - x_1)^2.$$

Vi har at $p'_3(x) = c_1 + c_2(2x - x_0 - x_1) + c_2(x - x_1)(3x - 2x_0 - x_1)$, og interpolasjonsbetingelsene gir

$$\begin{aligned} f(x_0) &= c_0 \\ f(x_1) &= c_0 + c_1(x_1 - x_0) \\ f'(x_1) &= c_1 + c_2(x_1 - x_0) \\ f(x_2) &= c_0 + c_1(x_2 - x_0) + c_2(x_2 - x_0)(x_2 - x_1) + c_3(x_2 - x_0)(x_2 - x_1)^2 \end{aligned}$$

Dette gir matriselikningen

$$\begin{pmatrix} f(x_0) \\ f(x_1) \\ f'(x_1) \\ f(x_2) \end{pmatrix} = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 1 & x_1 - x_0 & 0 & 0 \\ 0 & 1 & x_1 - x_0 & 0 \\ 1 & x_2 - x_0 & (x_2 - x_0)(x_2 - x_1) & (x_2 - x_0)(x_2 - x_1)^2 \end{pmatrix} \begin{pmatrix} c_0 \\ c_1 \\ c_2 \\ c_3 \end{pmatrix}$$

Igjen er koeffisientmatrisen her nedretriangulær med ikkenuller på diagonalen, slik at den er inverterbar. Det følger at koeffisientene er unike, slik at det finnes et unikt polynom.

Exercise 10(b). Ja, det er mulig å generalisere dette. Anta at $f(x_i), f'(x_i), \dots, f^{(k-1)}$ er kjente. Vi har bidrag fra $r_n(x)(x-x_i), r_n(x)(x-x_i)^2, \dots, r_n(x)(x-x_i)^k, r_n(x)(x-x_i)^k s_n(x)$, der r_n representerer bidraget fra $x_0, \dots, x_{i-1}$, s_n bidraget fra $x_{i+1}, \dots, x_n$.

Verdien til den deriverte av $r_n(x)(x-x_i)$ i x_i er $r_n(x_i)$. Den deriverte til $r_n(x)(x-x_i)^2$ o.s.v. i x_i er 0.

Verdien til derivert nr. $k-1$ til $r_n(x)(x-x_i)^{k-1}$ er $(k-1)!r_n(x)$, som er $\neq 0$.

Igen får vi et triangulært system med ikkenummer på diagonalen.

Exercise 11. Vi har at $f^{(n+1)}(x) = e^x$, slik at $M_{n+1} = e$. Teorem 10.15 gir da at, for alle $x \in [0, 1]$, så er

$$|f(x) - p_n(x)| \leq \frac{e \cdot (1-0)^{n+1}}{(n+1)!} = \frac{e}{(n+1)!}$$

Vi krever da at $\frac{e}{(n+1)!} \leq 10^{-3}$, det vil si at $(n+1)! \geq 1000e$. Dette kan vi få til ved å velge $n \geq 6$.

Exercise 13(b). Vi får

$$\begin{aligned} \int_{x_0}^{x_2} (x-x_0)(x-x_1) dx &= \left[\frac{1}{2}(x-x_0)^2(x-x_1) \right]_{x_0}^{x_2} - \frac{1}{2} \int_{x_0}^{x_2} (x-x_0)^2 dx \\ &= 2h^3 - \frac{8}{6}h^3 = \frac{2}{3}h^3. \end{aligned}$$

og

$$\begin{aligned} &\int_{x_0}^{x_3} (x-x_0)(x-x_1)(x-x_2) dx \\ &= \left[\frac{1}{2}(x-x_0)^2(x-x_1)(x-x_2) \right]_{x_0}^{x_3} - \frac{1}{2} \int_{x_0}^{x_3} (x-x_0)^2((x-x_1) + (x-x_2)) dx \\ &= 9h^4 - \frac{1}{6}[(x-x_0)^3((x-x_1) + (x-x_2))]_{x_0}^{x_3} + \frac{1}{6} \int_{x_0}^{x_3} 2(x-x_0)^3 dx \\ &= 9h^4 - \frac{27}{2}h^4 + \frac{1}{24}[(x-x_0)^4]_{x_0}^{x_3} \\ &= -\frac{9}{2}h^4 + \frac{27}{8}h^4 = -\frac{9}{8}h^4 \end{aligned}$$

Exercise 14. Følgende kode genererer plottet vist i Figur10.1.

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

xp = np.linspace(0,1,100).reshape((-1,1))
```

```

for n in range(2,5):
    x = np.linspace(0,1,n+1).reshape((-1,1))
    p = np.arange(0,n+1).reshape((1,-1))
    V = x**p
    c = np.linalg.solve(V, np.cos(2*np.pi*x))
    x_powers = x**p
    y = x_powers@c
    plt.plot(xp,y)
plt.plot(xp,np.cos(2*np.pi*xp))
plt.legend(['n=2', 'n=3', 'n=4', 'cos(2$pi$ x)'])
plt.show()

```

Kapittel 11

Exercise 1. Polynomet som interpolerer f i a og $a-h$ er linja $p_1(x) = f(a) + \frac{f(a)-f(a-h)}{h}(x-a)$. Denne har derivert $(f(a)-f(a-h))/h$, som er den venstresidige varianten til newtonkvotienten.

Exercise 3(a). Vi skriver nå

$$f(a+h) = f(a) + hf'(a) + \frac{h^2}{2}f''(a) + \frac{h^3}{6}f'''(\xi_h)$$

for en ξ_h i intervallet fra a til $a+h$. Vi kan nå skrive

$$f'(a) - \frac{f(a+h) - f(a)}{h} = -\frac{h}{2}f''(a) - \frac{h^2}{6}f'''(\xi_h).$$

Etter at vi tar absoluttverdier blir feilestimatet

$$\frac{h}{2}|f''(a)| + \frac{h^2}{6} \max_{x \in [a, a+h]} |f'''(x)|.$$

Dette feilestimatet ligner på det vi fikk da vi brukte et Taylorutvikling av orden én, men vi trenger også en generelt estimat på den tredjederiverte. Vi trenger fremdeles et estimat på den andrederiverte, men bare i punktet a .

Exercise 3(b). Vi skriver nå $f(a+h) = f(a) + f'(\xi_h)h$ for en ξ_h i intervallet mellom a and $a+h$. I dette uttrykket er ikke $f'(a)$ til stede, så det er ikke mulig å få et estimat på trunkeringsfeilen med en slik Taylorutvikling.

Exercise 3(c). Det lineære Taylorpolynomet er best siden det er den korteste mulige Taylorutviklingen som kan gi et estimat for $f'(a)$ (som vi viste i (b)), og også det som gir det enkleste uttrykket for trunkeringsfeilen, på den måten at den ikke er avhengig av deriverte av høyere orden enn to (som vi viste i (a)).

Exercise 4(a). Tilnærmingen er eksakt for alle polynomer av grad 3 siden $M_4 = 0$ for disse (den fjerdederiverte av et tredjegradspolynom er 0). Hvis vi prøver med $f(x) = x^4$ blir tilnærmingen $\frac{h^4 + h^4}{h^2} = h^2$, som er forskjellig fra $f''(0) = 0$.

Exercise 4(b). $\frac{h^2}{12} \max_{x \in [-h, h]} |f^{(4)}(x)|$ er riktig svar

Exercise 5(a). Sant. Siden $M_3 = 0$ for alle andregradspolynomer, og siden den symmetriske newtonkvotienten har trunkeringsfeil $\leq M_3 \frac{h^2}{6}$, så er den symmetriske newtonkvotienten eksakt for alle andregradspolynomer.

Exercise 5(d). Sant. Siden $M_5 = 0$ for alle fjerdegradspolynomer, og siden firepunktsmetoden har trunkeringsfeil $\leq \frac{M_5}{18} h^4$, så er firepunktsmetoden eksakt for alle fjerdegradspolynomer. Hvis vi prøver med $f(x) = x^5$ så blir tilnærmingen til den deriverte i 0

$$\frac{(-2h)^5 - 8(-h)^5 + 8h^5 - (2h)^5}{12h} = \frac{h^5(-32 + 8 + 8 - 32)}{12h} = -4h^4,$$

mens $f'(0) = 0$, slik at metoden ikke er eksakt for $f(x) = x^5$.

Exercise 7(a). Det er kanskje enklest å vise først at dette holder for de tre funksjonene $f(x) = 1$, $f(x) = x$, og $f(x) = x^2$:

- For $f(x) = 1$ er $f'(a) = 0$, og den symmetriske Newtonkvotienten er

$$\frac{1 - 1}{2h} = 0.$$

- For $f(x) = x$ er $f'(a) = 1$, og den symmetriske Newtonkvotienten er

$$\frac{a + h - (a - h)}{2h} = 1.$$

- For $f(x) = x^2$ er $f'(a) = 2a$, og den symmetriske Newtonkvotienten er

$$\frac{(a + h)^2 - (a - h)^2}{2h} = \frac{4ah}{2h} = 2a.$$

Den symmetriske Newtonkvotienten blir da også riktig for alle andregradspolynomer, siden alle slike kan skrives som lineære kombinasjoner av 1, x , og x^2 , og siden derivasjon og Symmetrisk Newtonkvotient er lineære operasjoner.

Exercise 7(b). Vi har at

$$\begin{aligned}\frac{f_r(a) + f_l(a)}{2} &= \frac{\frac{f(a+h)-f(a)}{h} + \frac{f(a)-f(a-h)}{h}}{2} \\ &= \frac{f(a+h) - f(a) + f(a) - f(a-h)}{2h} = \frac{f(a+h) - f(a-h)}{2h},\end{aligned}$$

som er den symmetriske Newtonkvotienten.

Exercise 8. Vi skal løse disse to deloppgavene på tre forskjellige måter. Først skal vi finne tilnærmingene ved å derivere det interpolerende polynomet. Deretter skal vi bruke Teorem 11.4. Til slutt utleder vi de samme tilnærmingene ved å kombinere Taylorrekker av passende orden i de gitte punktene

Exercise 8(a). Newtonformen til polynomet som interpolerer i $a-h$, a , og $a+2h$ er

$$p_2(x) = c_0 + c_1(x - (a - h)) + c_2(x - a)(x - (a - h)).$$

Vi evaluerer dette i de tre punktene, og løser for c_0, c_1, c_2 :

$$f(a - h) = c_0$$

$$f(a) = c_0 + c_1 h f(a + 2h) = c_0 + 3c_1 h + 6c_2 h^2.$$

Løser vi dette får vi først $c_0 = f(a - h)$, $c_1 = \frac{f(a) - f(a - h)}{h}$, og så

$$c_2 = \frac{f(a + 2h) - f(a - h) - 3(f(a) - f(a - h))}{6h^2} = \frac{f(a + 2h) - 3f(a) + 2f(a - h)}{6h^2}.$$

Dermed får vi

$$p_2(x) = f(a - h) + \frac{f(a) - f(a - h)}{h}(x - (a - h)) + \frac{f(a + 2h) - 3f(a) + 2f(a - h)}{6h^2}(x - a)(x - (a - h)),$$

og så

$$p'_2(x) = \frac{f(a) - f(a - h)}{h} + \frac{f(a + 2h) - 3f(a) + 2f(a - h)}{6h^2}((x - a) + (x - (a - h))),$$

og til slutt

$$p'_2(a) = \frac{f(a) - f(a - h)}{h} + \frac{f(a + 2h) - 3f(a) + 2f(a - h)}{6h} = \frac{f(a + 2h) + 3f(a) - 4f(a - h)}{6h}.$$

Teorem 11.4 gir med $m_1 = -1$, $m_2 = 2$.

$$\begin{pmatrix} h p_2^{(1)}/1! \\ h^2 p_2^{(2)}/2! \end{pmatrix} = \begin{pmatrix} -1 & (-1)^2 \\ 2 & 2^2 \end{pmatrix}^{-1} \begin{pmatrix} f(a - h) - f(a) \\ f(a + 2h) - f(a) \end{pmatrix} = -\frac{1}{6} \begin{pmatrix} 4 & -1 \\ -2 & -1 \end{pmatrix} \begin{pmatrix} f(a - h) - f(a) \\ f(a + 2h) - f(a) \end{pmatrix}$$

Første rad her sier at

$$\begin{aligned} f'(a) \approx p_2^{(1)} &= -\frac{1}{6h} (4(f(a-h) - f(a)) - (f(a+2h) - f(a))) \\ &= \frac{-4f(a-h) + 3f(a) + f(a+2h)}{6h}. \end{aligned}$$

Alternativt kan vi skrive opp

$$\begin{aligned} f(a-h) &= f(a) - f'(a)h + \frac{f''(a)}{2}h^2 + \frac{f'''(c_1)}{6}h^3 \\ f(a+2h) &= f(a) + 2f'(a)h + 2f''(a)h^2 + 4\frac{f'''(c_2)}{3}h^3 \end{aligned}$$

Ganger vi den første likningen med 4 og trekker de fra hverandre (dette eliminerer $f''(a)$) får vi at

$$4f(a-h) - f(a+2h) = 3f(a) - 6f'(a)h + CM_3h^3,$$

der C er et tall. Dette gir tilnærmingen $f'(a) \approx \frac{3f(a) - 4f(a-h) + f(a+2h)}{6h}$, der feilen er $\approx CM_3h^2$.

Exercise 8(b). Newtonformen til polynomet som interpolerer i a , $a+h$, og $a+2h$ er

$$p_2(x) = c_0 + c_1(x-a) + c_2(x-a)(x-(a+h)).$$

Vi evaluerer dette i de tre punktene, og løser for c_0, c_2, c_2 :

$$\begin{aligned} f(a) &= c_0 \\ f(a+h) &= c_0 + c_1h + c_2h^2 \end{aligned}$$

Løser vi dette får vi først $c_0 = f(a)$, $c_1 = \frac{f(a+h)-f(a)}{h}$, og så

$$c_2 = \frac{f(a+2h) - f(a) - 2f(a+h) + 2f(a)}{2h^2} = \frac{f(a+2h) - 2f(a+h) + f(a)}{2h^2}.$$

Dermed får vi

$$p_2(x) = f(a) + \frac{f(a+h)-f(a)}{h}(x-a) + \frac{f(a+2h)-2f(a+h)+f(a)}{2h^2}(x-a)(x-(a+h)),$$

og så

$$p_2'(x) = \frac{f(a+h)-f(a)}{h} + \frac{f(a+2h)-2f(a+h)+f(a)}{2h^2}((x-a) + (x-(a+h))),$$

og til slutt

$$p'_2(a) = \frac{f(a+h) - f(a)}{h} - \frac{f(a+2h) - 2f(a+h) + f(a)}{2h} = \frac{-f(a+2h) + 4f(a+h) - 3f(a)}{2h}$$

Teorem 11.4 gir med $m_1 = 1$, $m_2 = 2$.

$$\begin{pmatrix} hp_2^{(1)}/1! \\ h^2 p_2^{(2)}/2! \end{pmatrix} = \begin{pmatrix} 1 & 1 \\ 2 & 4 \end{pmatrix}^{-1} \begin{pmatrix} f(a+h) - f(a) \\ f(a+2h) - f(a) \end{pmatrix} = \frac{1}{2} \begin{pmatrix} 4 & -1 \\ -2 & 1 \end{pmatrix} \begin{pmatrix} f(a+h) - f(a) \\ f(a+2h) - f(a) \end{pmatrix}$$

Første rad her sier at

$$\begin{aligned} f'(a) &\approx p_2^{(1)} = \frac{1}{2h} (4(f(a+h) - f(a)) - (f(a+2h) - f(a))) \\ &= \frac{-3f(a) + 4f(a+h) - f(a+2h)}{2h}. \end{aligned}$$

Alternativt kan vi skrive opp

$$\begin{aligned} f(a+h) &\approx f(a) + f'(a)h + \frac{f''(a)}{2}h^2 + \frac{f'''(c_1)}{6}h^3 \\ f(a+2h) &\approx f(a) + 2f'(a)h + 2f''(a)h^2 + 4\frac{f'''(c_2)}{3}h^3 \end{aligned}$$

Ganger vi den første likningen med 4 og trekker de fra hverandre (dette eliminerer $f''(a)$) får vi at

$$4f(a+h) - f(a+2h) = 3f(a) + 2f'(a)h + CM_3h^3,$$

der C er et tall. Dette gir tilnærmingen $f'(a) \approx \frac{-3f(a) + 4f(a+h) - f(a+2h)}{2h}$, der feilen er $\approx CM_3h^2$ også her.

Exercise 9(a). Hvis tilnærmingen $f'(a) \approx c_1 f(a-h) + c_2 f(a+h)$ skal være eksakt for $f(x) = 1$, så må vi ha at $0 = c_1 + c_2$, siden $f(a-h) = f(a+h) = 1$, og siden $f'(x) = 0$. Derfor må vi ha at $c_2 = -c_1$.

Hvis metoden skal være eksakt for $f(x) = x$ så må vi på samme måte ha at

$$1 = c_1(a-h) + c_2(a+h) = c_1(a-h) - c_1(a+h) = -2c_1h,$$

slik at $c_1 = -\frac{1}{2h}$, slik at også $c_2 = \frac{1}{2h}$. Metoden blir derfor $-\frac{1}{2h}f(a-h) + \frac{1}{2h}f(a+h) = \frac{f(a+h) - f(a-h)}{2h}$

Exercise 9(b). Hvis $f(x) = cx + d$ så har vi at $f'(x) = c$, og metoden blir

$$\frac{f(a+h) - f(a-h)}{2h} = \frac{c(a+h) + d - (c(a-h) + d)}{2h} = \frac{2ch}{2h} = c,$$

slik at metoden er eksakt for alle polynomer av grad ≤ 1 . Vi ser at metoden faller sammen med den symmetriske Newtonkvotienten, og den har derfor en feil av orden h^2 , som er bedre enn Newtonkvotienten (som har en feil av orden h). Den er verre enn firepunktsmetoden, som har orden h^4 .

Metoden også er eksakt for polynomer av grad ≤ 2 . Det er flere måter å se dette på. For det første, feilestimatet fra den seksjonen bruker $f^{(3)}(x)$, og siden alle andregradspolynomer har en tredjederivert lik 0, så må feilen også være 0. Man kan også som over substituere $f(x) = x^2$ i formelen:

$$\frac{f(a+h) - f(a-h)}{2h} = \frac{(a+h)^2 - (a-h)^2}{2h} = \frac{4ah}{2h} = 2a,$$

som også er $f'(a)$. Til slutt, den symmetriske Newtonkvotienten var definert som den deriverte i a til den unike parabolen som interpolerer f i $a-h$, a , og $a+h$. Hvis f selv er en parabel så er den lik interpolanten siden denne er unik, slik at den symmetriske Newtonkvotienten må returnere den deriverte.

Exercise 9(c). Hvis approksimasjonsmetoden $f''(a) \approx c_1 f(a-h) + c_2 f(a) + c_3 f(a+h)$ er eksakt for $f(x) = 1$, så må vi ha at $0 = c_1 + c_2 + c_3$. Hvis den er eksakt for $f(x) = x$ så må vi ha at

$$0 = c_1(a-h) + c_2 a + c_3(a+h) = a(c_1 + c_2 + c_3) + h(-c_1 + c_3) = h(-c_1 + c_3),$$

som gir at $c_1 = c_3$. Hvis den er eksakt for $f(x) = x^2$ så må vi ha at

$$\begin{aligned} 2 &= c_1(a-h)^2 + c_2 a^2 + c_3(a+h)^2 \\ &= a^2(c_1 + c_2 + c_3) - 2ahc_1 + 2ahc_3 + h^2(c_1 + c_3) = 2c_1 h^2, \end{aligned}$$

som gir at $c_1 = \frac{1}{h^2}$. Vi får derfor også at $c_3 = \frac{1}{h^2}$, og at $c_2 = -c_1 - c_3 = -\frac{2}{h^2}$, slik at metoden blir

$$\frac{1}{2h} f(a-h) - \frac{1}{h} f(a) + \frac{1}{2h} f(a+h) = \frac{f(a-h) - 2f(a) + f(a+h)}{h^2}.$$

Vi ser at dette faller sammen med trepunktsmetoden for den andraderiverte som vi allerede har sett på.

Exercise 9(d). Alle tredjegradspolynomer har fjerdederivert lik 0, og derfor blir

trunkeringsfeilen 0 ($M_4 = 0$). Alternativt kan vi substituere $f(x) = x^3$ i formelen:

$$\begin{aligned} & \frac{f(a-h) - 2f(a) + f(a+h)}{2h} \\ &= \frac{(a-h)^3 - 2a^3 + (a+h)^3}{h^2} \\ &= \frac{a^3 - 3a^2h + 3ah^2 - h^3 - 2a^3 + a^3 + 3a^2h + 3ah^2 + h^3}{h^2} \\ &= \frac{6ah^2}{h^2} = 6a, \end{aligned}$$

som faller sammen med den andrederiverte til f i a .

Exercise 10(a). Vi kan tegne grafen sammen med sekantene som følger:

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

x = np.arange(0,6,0.05)
f = lambda x: (-x**2+10*x-5)/4
plt.plot(x,f(x))
plt.plot([1,3],[f(1),f(3)])
plt.plot([1,5],[f(1),f(5)])
plt.plot([3,5],[f(3),f(5)])
plt.show()
```

Exercise 10(b). Med 6 sifre:

$(f(a+h) - f(a))/h = 0.975$, relativ feil: 0.025.

$(f(a) - f(a-h))/h = 1.025$, relativ feil: 0.025.

$(f(a+h) - f(a-h))/(2h) = 1$, relativ feil: 8.88178×10^{-16} .

Kapittel 12

Exercise 1(a). På hvert delintervall antar f minste verdi i venstre endepunkt, største verdi i høyre endepunkt. Her kan vi bruke sum-kommandoen på følgende måte til å finne $\underline{I}$ og $\bar{I}$.

```
import numpy as np

def int_vals(n):
    l = (1/n)*np.sum(np.exp( np.arange(0,n)/n ))
```

```

u = (1/n)*np.sum(np.exp( np.arange(1,n+1)/n ))
l_err = np.abs(1-(np.exp(1)-1))
u_err = np.abs(u-(np.exp(1)-1))
return l,u,l_err,u_err

```

Listing 1. Utregning av øvre/nedre-integralet

kjører vi koden får vi

```

>> l,u,l_err,u_err=int_vals(10)
1.6337993999663625,
1.805627582812267,
0.08448242849268262,
0.0873457543532219

```

Exercise 1. Kjører vi koden

```

n = 1
l,u,l_err,u_err=int_vals(n)
while l_err > 3*10**(-3):
    n += 1
    l,u,l_err,u_err = int_vals(n)
print(n)

```

får vi at $n = 287$.

Exercise 2(a). Sant

Exercise 2(b). Galt

Exercise 2(c). Sant

Exercise 2(d). True

Exercise 3. Vi får at

$$I_{mid} = \frac{1}{2}f(1/4) + \frac{1}{2}f(3/4) = \frac{1}{2}\frac{1}{16} + \frac{1}{2}\frac{9}{16} = \frac{1}{2}\frac{10}{16} = \frac{5}{16},$$

slik at det første alternativet er riktig.

Exercise 4. Intervallet $[0, \pi/2]$ splittes her opp i intervallene

$$[0, \pi/12], [\pi/12, 2\pi/12], [2\pi/12, 3\pi/12], [3\pi/12, 4\pi/12], \\ [4\pi/12, 5\pi/12], [5\pi/12, 6\pi/12].$$

Midtpunktene på disse intervallene er $\pi/24, 3\pi/24, 5\pi/24, 7\pi/24, 9\pi/24, 11\pi/24$. Med $h = \pi/12$ gir midtpunktsmetoden

$$\begin{aligned} I_{mid} &= \frac{\pi}{12} (f(\pi/24) + f(3\pi/24) + f(5\pi/24) + f(7\pi/24) + f(9\pi/24) + f(11\pi/24)) \\ &= \frac{\pi}{12} \left(\frac{\sin(\pi/24)}{1 + (\pi/24)^2} + \frac{\sin(3\pi/24)}{1 + (3\pi/24)^2} + \frac{\sin(5\pi/24)}{1 + (5\pi/24)^2} + \frac{\sin(7\pi/24)}{1 + (7\pi/24)^2} \right. \\ &\quad \left. + \frac{\sin(9\pi/24)}{1 + (9\pi/24)^2} + \frac{\sin(11\pi/24)}{1 + (11\pi/24)^2} \right) \\ &\approx 0.530624. \end{aligned}$$

Exercise 5. Vi kan godt lage en funksjon her, som vi kan bruke på forskjellige input:

```
def midpointn(f, a, b, n):  
    h = (b - a)/n  
    return h*sum( f(a + h/2 + np.arange(0,n)*h))
```

Kode som løser oppgaven blir da som følger

```
midpointn(np.exp, 0, 1, 10)
```

Exercise 5(a). Tilnærmingen med 10 delintervaller: 1.71757 (med 6 sifre)

Exercise 5(b). Fra Teorem 12.7 vet vi at feilen er begrenset av $(b-a) \frac{h^2}{24} \max_{x \in [a,b]} |f''(x)|$. Siden $f''(x) = e^x$, $\max_{x \in [0,1]} |f''(x)| = e$, slik at dette er $\frac{eh^2}{24}$. Vi må derfor sørge for at $\frac{eh^2}{24} < 10^{-10}$, slik at $h < \sqrt{\frac{24}{e}} 10^{-5} \approx 2.97 \times 10^{-5}$.

Exercise 6. Tilnærming med 10 delintervaller: 5.36648 (med 6 sifre). $h \leq 4.89 \times 10^{-5}$.

Exercise 7(a). Feil. Midtpunktsmetoden har feil begrenset av $(b-a)h^2 M_2/24$, mens det beste estimatet vi fant for trapesmetoden var $(b-a)h^2 M_2/12$.

Exercise 7(b). Feil. I stedet for å regne ut samme funksjonsverdi $f(x_i)$ to ganger så kan vi skrive $2f(x_i)$, siden alt skal summeres.

Exercise 7(c). Sant. For ett intervall trengs 3 delepunkter. For hvert intervall vi legger til ytterligere to delepunkter, slik at det totale antallet alltid blir odde.

Exercise 7(d). Sant, men dette bygger på Oppgave 15. Hvis vi bruker bare estimatet fra (12.12) så kan vi ikke konkludere med dette.

Exercise 8(a). Vi har at

$$I_{\text{trap}} = \frac{1}{2} \left(\frac{f(0) + f(1)}{2} + f\left(\frac{1}{2}\right) \right) = \frac{1}{2} \left(\frac{0+1}{2} + \frac{1}{4} \right) = \frac{3}{8},$$

slik at det andre alternativet er riktig.

Exercise 8(b). Alle er eksakte.

Exercise 8(c). Det følger fra (12.12) at Simpsons metod er eksakt for alle polynomer av grad to.

Exercise 8(d). Det siste alternativet er riktig.

Exercise 9. Oppgave 4: Tilnærming: 0.519725 (med 6 sifre).

Oppgave 5: Tilnærming med 10 delintervaller: 1.71971 (med 6 sifre). $h \leq 2.1 \times 10^{-5}$.

Vi kan godt lage en funksjon her, som vi kan bruke på forskjellige input:

```
def trapezoidaln(f, a, b, n):
    h = (b - a)/n
    return h*( (f(a) + f(b))/2 + sum( f(a + np.arange(1,n)*h)))
```

Kode som løser Oppgave 5 og 4 blir da som følger

```
trapezoidaln(np.exp,0,1,10)
trapezoidaln(lambda x : np.sin(x)/(1+x**2), 0, pi/2, 6)
```

Feilen er begrenset av

$$(b-a) \frac{h^2}{12} \max_{x \in [0,1]} |f''(x)| = \frac{h^2}{12} e.$$

Vi har at $\frac{h^2}{12} e \leq 10^{-10}$ når $h \leq \sqrt{12/e} \times 10^{-5} \approx 2.1 \times 10^{-5}$.

Exercise 10(a). Siden feilen i trapesmetoden er begrenset av

$$(b-a) \frac{h^2}{12} \max_{x \in [a,b]} |f''(x)|,$$

og siden $f''(x) = \frac{8}{(1+2x)^3}$ (som får maksimum absolutt verdi for $x = 0$), så trenger vi å velge h slik at

$$(b-a) \frac{h^2}{12} \max_{x \in [a,b]} |f''(x)| = 8 \frac{h^2}{12} \leq 10^{-10},$$

som gir at $h \leq 10^{-5} \sqrt{3/2}$. Antall funksjonsevalueringer er da $\approx 1/h = 10^5 / \sqrt{2/3} \approx 81649.66$, som betyr at minst 81650 evalueringer trengs.

Exercise 10(b). Siden feilen i midtpunktsmetoden er begrenset av

$$(b-a) \frac{h^2}{24} \max_{x \in [a,b]} |f''(x)|,$$

så får vi som i (a) at

$$(b-a) \frac{h^2}{24} \max_{x \in [a,b]} |f''(x)| = 2 \frac{h^2}{6} \leq 10^{-10},$$

som gir at $h \leq 10^{-5} \sqrt{3}$. Antall funksjonsevalueringer er derfor omtrent $1/h = 10^5 / \sqrt{3} \approx 57735.03$. Med andre ord, minst 57736 evalueringer trengs.

Exercise 10(c). Siden feilen i Simpsons metode er begrenset av

$$(b-a) \frac{h^4}{180} \max_{x \in [a,b]} |f^{(iv)}(x)|,$$

og siden $f^{(iv)}(x) = \frac{384}{(1+2x)^5}$ (som får maksimum absolutt verdi for $x = 0$), så trenger vi velger h slik at

$$(b-a) \frac{h^4}{180} \max_{x \in [a,b]} |f^{(iv)}(x)| = 384 \frac{h^4}{180} \leq 10^{-10},$$

som betyr at h er begrenset av $h \leq (180 \cdot 10^{-10} / 384)^{1/4}$. Antall funksjonsevalueringer er da $\approx 1/h = (180 \cdot 10^{-10} / 384)^{-1/4} \approx 382.17$, som betyr at minst 383 evalueringer trengs.

Exercise 11. Oppgave 4: Tilnærming: 0.527217 (med 6 sifre).

Oppgave 5: Tilnærming med 10 delintervaller: 1.718282782 (med 10 sifre). $h \leq 1.8 \times 10^{-2}$

Vi kan godt lage en funksjon her, som vi kan bruke på forskjellige input:

```
def simpsonn(f, a, b, n):
    h = (b - a) / (2*n)
    return h * ( f(a) + f(b) + 2*sum( f(a+2*h*np.arange(1,n) )) \
                + 4*sum( f(a+h+2*h*np.arange(0,n))) ) / 3
```

Kode som løser Oppgave 5 og 4 blir da som følger

```
simpsonn(np.exp,0,1,10)
simpsonn(lambda x : np.sin(x)/(1+x**2), 0, pi/2, 6)
```

Exercise 12. Siden $\int_a^b f(x) dx = w_1 f(a) + w_2 f((a+b)/2) + w_3 f(b)$ for $f(x) = x^i$ for $i = 0, 1, 2$, så har vi også at dette holder for polynomet p_2 som interpolerer f i a , $(a+b)/2$, og b . Derfor får vi at

$$\int_a^b p_2(x) dx = w_1 p_2(a) + w_2 p_2((a+b)/2) + w_3 p_2(b) = w_1 f(a) + w_2 f((a+b)/2) + w_3 f(b).$$

Vi har også vist at det første er lik $\frac{b-a}{6}(f(a) + 4f((a+b)/2) + f(b))$, slik at

$$w_1 f(a) + w_2 f((a+b)/2) + w_3 f(b) = \frac{b-a}{6}(f(a) + 4f((a+b)/2) + f(b)).$$

Ved å sette $f(a) = 1$, $f((a+b)/2) = f(b) = 0$ ser vi at $w_1 = \frac{b-a}{6}$, og tilsvarende for w_2 , w_3 . Det følger at metoden er lik Simpsons metode.

Exercise 13(a). Delvis integrasjon på (12.13) med constant C i den antideriverte av $x - m$ gir

$$\begin{aligned} L &= - \left[\left(C + \frac{1}{2}(x-m)^2 \right) f'(x) \right]_a^b + \int_a^b \left(C + \frac{1}{2}(x-m)^2 \right) f''(x) dx \\ &= \int_a^b \left(C + \frac{1}{2}(x-m)^2 \right) f''(x) dx. \end{aligned}$$

som viser (12.14).

Exercise 13(b). For $x \in [a, b]$ har vi at $\frac{1}{2}(x-m)^2 \leq \frac{1}{8}(b-a)^2 = -C$, slik at $-C - \frac{1}{2}(x-m)^2 \geq 0$ for slike x . Tar vi absoluttverdier i (12.14) med dette valget av C får

vi

$$\begin{aligned}
|L| &= \left| \int_a^b \left(\frac{1}{2}(x-m)^2 - (b-a)^2/8 \right) f''(x) dx \right| \\
&\leq \int_a^b \left| \frac{1}{2}(x-m)^2 - (b-a)^2/8 \right| |f''(x)| dx \\
&\leq \int_a^b \left((b-a)^2/8 - \frac{1}{2}(x-m)^2 \right) M_2 dx \\
&= M_2 \left[x(b-a)^2/8 - \frac{1}{6}(x-m)^3 \right]_a^b \\
&= M_2 \left((b-a)^3/8 - \frac{1}{48}(b-a)^3 - \frac{1}{48}(b-a)^3 \right) \\
&= M_2 \left(\frac{1}{8} - \frac{1}{24} \right) (b-a)^3 = \frac{M_2(b-a)^3}{12}.
\end{aligned}$$

Exercise 13(c). Som for midtpunktsmetoden så får man den globale feilen ved å summere de lokale feilene (erstatt $b-a$ med h , og nh med $b-a$):

$$\sum_{k=1}^n \frac{M_2 h^3}{12} = n \frac{M_2 h^3}{12} = (b-a) \frac{h^2}{12} M_2.$$

Dette var det vi skulle vise.

Exercise 14(a). Det er nok å verifisere Simpsons metode på hvert delintervall. Integralene blir

$$\begin{aligned}
\int_{a-h}^{a+h} x^3 dx &= \frac{1}{4}((a+h)^4 - (a-h)^4) = 2a^3h + 2ah^3 \\
\int_{a-h}^{a+h} x^2 dx &= \frac{1}{3}((a+h)^3 - (a-h)^3) = \frac{1}{3}(6a^2h + 2h^3) = \frac{h}{3}(6a^2 + 2h^2) \\
\int_{a-h}^{a+h} x dx &= \frac{1}{2}((a+h)^2 - (a-h)^2) = \frac{1}{2}4ah = 2ah \\
\int_{a-h}^{a+h} dx &= 2h.
\end{aligned}$$

Vi får også at metoden blir

$$\begin{aligned}\frac{h}{3}((a-h)^3 + 4a^3 + (a+h)^3) &= \frac{h}{3}(6a^3 + 6ah^2) = 2a^3h + 2ah^3 \\ \frac{h}{3}((a-h)^2 + 4a^2 + (a+h)^2) &= \frac{h}{3}(6a^2 + 2h^2) \\ \frac{h}{3}((a-h) + 4a + (a+h)) &= 2ah \\ \frac{h}{3}(1 + 4 + 1) &= 2h.\end{aligned}$$

Dette viser at Simpsons metode er eksakt for de gitte polynomene.

Exercise 14(b). For $f(x) = bx^3 + cx^2 + dx + e$ blir integralet

$$\int_{a-h}^{a+h} f(x) dx = b \int_{a-h}^{a+h} x^3 dx + c \int_{a-h}^{a+h} x^2 dx + d \int_{a-h}^{a+h} x dx + e \int_{a-h}^{a+h} 1 dx.$$

Metoden blir nå

$$\begin{aligned}&\frac{h}{3}(f(a-h) + 4f(a) + f(a+h)) \\ &= b \frac{h}{3}((a-h)^3 + 4a^3 + (a+h)^3) + c \frac{h}{3}((a-h)^2 + 4a^2 + (a+h)^2) \\ &\quad + d \frac{h}{3}((a-h) + 4a + (a+h)) + e \frac{h}{3}(1 + 4 + 1)\end{aligned}$$

Likhet følger nå fra at vi har likhet for x^3 , x^2 , x , og 1 .

Exercise 14(c). Siden $f^{(4)}(x) = 0$ for alle x for ethvert kubisk polynom, så følger det direkte fra feilestimatet at metoden er eksakt for slike funksjoner.

Exercise 14(d). Vi har at

$$\begin{aligned}q(m-y) &= (h-y)(-y)(-h-y) \\ q(m+y) &= (h+y)y(-h+y) = -(h-y)(-y)(-h-y),\end{aligned}$$

slik at $q(m-y) = -q(m+y)$. Det følger at q er anti-symmetrisk om m , slik at $\int_a^b q(x) dx = 0$. Anta nå at f er et polynom av grad 3. Newtonformene er

$$\begin{aligned}p_2(x) &= c_0 + c_1(x-a) + c_2(x-a)(x-m) \\ f(x) &= c_0 + c_1(x-a) + c_2(x-a)(x-m) + c_3(x-a)(x-m)(x-b)\end{aligned}$$

Siden $\int_a^b (x-a)(x-m)(x-b) dx = \int_a^b q(x) dx = 0$ så følger det at $\int_a^b f(x) dx = \int_a^b p_2(x)$, slik at Simpsons metode for polynomer av grad 3.

Exercise 15(a). Delvis integrasjon tre ganger gir

$$\begin{aligned}\int_a^b xq(x) &= \int_a^b x(x-a)(x-m)(x-b) dx = -\frac{1}{2} \int_a^b (x-a)^2(x-m)(x-b) dx \\ &= \frac{1}{6} \int_a^b (x-a)^3(x-b) dx \\ &= -\frac{1}{24} \int_a^b (x-a)^4 dx \\ &= -\frac{1}{120} (b-a)^5,\end{aligned}$$

som viser første del. At $(xq(x))^{(4)} = 4!$ følger ved å bruke regelen for å derivere et produkt fire ganger.

Exercise 15(b). Vi har at $p_2^{(4)}(x) = q^{(4)}(x) = 0$. Innsetting av $(xq(x))^{(4)} = 4!$, og $\int_a^b xq(x) dx = -\frac{1}{120}(b-a)^5$ fra (a), viser første del. For den andre delen, vi vet fra før at $\int_a^b q(x) dx = 0$. Integrasjon av ϕ fra a til b gir derfor

$$\int_a^b f(x) dx - \int_a^b p_2(x) dx - E = E - E = 0.$$

Den siste delen følger fra at f og p_2 stemmer overens på a , m , og b , og fra at $q(a) = q(m) = q(b) = 0$.

Exercise 15(c). $\int_a^m \phi(x) dx = 0$ er det samme som

$$0 = \int_a^m f(x) dx - \int_a^m p_2(x) dx - k \int_a^m q(x) dx - \frac{E}{\int_a^b xq(x) dx} \int_a^m xq(x) dx,$$

som kan løses for k ved dele med $\int_a^m q(x) dx \neq 0$ (som er $\neq 0$ siden q ikke har noe fortegnsskifte på (a, m)).

Exercise 15(d). Siden $\int_a^b \phi(x) dx = \int_a^m \phi(x) dx = 0$, så må vi ha at $\int_m^b \phi(x) dx = 0$. Siden ϕ er kontinuerlig, så finnes det en $u \in (a, m)$ slik at $\phi(u) = 0$, og en $v \in (m, b)$ slik at $\phi(v) = 0$. ϕ har derfor minst fem forskjellige nullpunkter i $[a, b]$, siden vi også viste at $\phi(a) = \phi(m) = \phi(b) = 0$.

Exercise 15(e). Du kan argumentere som følger

1. Bruker du Rolles teorem på de fire intervallene (a, u) , (u, m) , (m, v) , og (v, b) , så får vi fire punkter $a_1 < a_2 < a_3 < a_4$, der alle $\phi'(a_i) = 0$.

2. Bruker du Rolles teorem igjen, men nå på de tre intervallene (a_1, a_2) , (a_2, a_3) , og (a_3, a_4) , så får vi tre punkter $b_1 < b_2 < b_3$, der alle $\phi''(b_i) = 0$.
3. Bruker du Rolles teorem en tredje gang, nå på de to intervallene (b_1, b_2) , (b_2, b_3) , så får vi to punkter $c_1 < c_2$ slik at $\phi'''(c_1) = \phi'''(c_2) = 0$.
4. Bruker du Rolles teorem en fjerde og siste gang så får vi en $\xi \in (c_1, c_2)$ slik at $\phi^{(4)}(\xi) = 0$,

Exercise 15(f). Evaluering av (12.15) i ξ gir

$$0 = f^{(4)}(\xi) + 120 \frac{4!E}{(b-a)^5},$$

slik at $E = -f^{(4)}(\xi)(b-a)^5/2880$. En øvre grense for feilen på ett intervall blir derfor $M_4(b-a)^5/2880$.

Exercise 16(a). Newtonformen får vi ved å løse likningene

$$f(a) = c_0$$

$$f(m) = c_0 + c_1 h$$

$$f(b) = c_0 + 2c_1 h + 2c_2 h^2.$$

Vi får $c_0 = f(a)$, $c_1 = \frac{f(m)-f(a)}{h}$, $c_2 = \frac{f(a)-2f(m)+f(b)}{2h^2}$. Dermed blir det interpolerende polynomet

$$p_2(x) = f(a) + \frac{f(m)-f(a)}{h}(x-a) + \frac{f(a)-2f(m)+f(b)}{2h^2}(x-a)(x-m).$$

Exercise 16(b). Her kan det være til hjelp først å regne ut $\int_a^b (x-a)(x-m) dx$ ved hjelp av delvis integrasjon:

$$\int_a^b (x-a)(x-m) dx = \left[\frac{1}{2}(x-a)^2(x-m) \right]_a^b - \frac{1}{2} \int_a^b (x-a)^2 dx = 2h^3 - \frac{4}{3}h^3 = \frac{2}{3}h^3.$$

Integrerer vi så $p_2(x)$ får vi

$$\begin{aligned} & \left[f(a)(x-a) + \frac{f(m)-f(a)}{2h}(x-a)^2 \right]_a^b + \frac{f(a)-2f(m)+f(b)}{2h^2} \frac{2}{3}h^3 \\ &= 2f(a)h + 2(f(m)-f(a))h + \frac{f(a)-2f(m)+f(b)}{h} \frac{1}{3}h \\ &= h \left(2f(m) + \frac{1}{3}(f(a)-2f(m)+f(b)) \right) \\ &= \frac{h}{3} (6f(m) + f(a) - 2f(m) + f(b)) \\ &= \frac{h}{3} (f(a) + 4f(m) + f(b)). \end{aligned}$$

Exercise 17. Vi kan skrive den om som følger.

$$I_{\text{trap}}(h) = h \left(\frac{f(a) + f(b)}{2} + \sum_{i=1}^{n-1} f(x_i) \right).$$

Exercise 18. Vi har at

$$\begin{aligned} \int_a^{a+nh} p_n(x) dx &= \int_a^{a+nh} \sum_{k=0}^n \frac{p_n^{(k)}(a)}{k!} (x-a)^k dx = \left[\sum_{k=0}^n \frac{p_n^{(k)}(a)}{(k+1)!} (x-a)^{k+1} \right]_a^{a+nh} \\ &= \sum_{k=0}^n \frac{p_n^{(k)}(a)}{(k+1)!} (nh)^{k+1} \\ &= hn \left(f(a) + \left(\frac{n}{2} \quad \frac{n^2}{3} \quad \dots \quad \frac{n^n}{n+1} \right) \begin{pmatrix} h^1 p_n^{(1)}(a)/1! \\ h^2 p_n^{(2)}(a)/2! \\ \vdots \\ h^n p_n^{(n)}(a)/n! \end{pmatrix} \right) \end{aligned}$$

som er uttrykket vi skulle fram til. For $n = 2$ gir Teorem 11.4

$$\begin{pmatrix} h^1 p_2^{(1)}(a)/1! \\ h^2 p_2^{(2)}(a)/2! \end{pmatrix} = \begin{pmatrix} 1 & 1 \\ 2 & 4 \end{pmatrix}^{-1} \begin{pmatrix} f(a+h) - f(a) \\ f(a+2h) - f(a) \end{pmatrix} = \begin{pmatrix} 2 & -1/2 \\ -1 & 1/2 \end{pmatrix} \begin{pmatrix} f(a+h) - f(a) \\ f(a+2h) - f(a) \end{pmatrix}$$

Setter vi inn dette i uttrykket over får vi

$$\begin{aligned} &2h \left(f(a) + \begin{pmatrix} 1 & 4/3 \end{pmatrix} \begin{pmatrix} 2 & -1/2 \\ -1 & 1/2 \end{pmatrix} \begin{pmatrix} f(a+h) - f(a) \\ f(a+2h) - f(a) \end{pmatrix} \right) \\ &= 2h \left(f(a) + \begin{pmatrix} 2/3 & 1/6 \end{pmatrix} \begin{pmatrix} f(a+h) - f(a) \\ f(a+2h) - f(a) \end{pmatrix} \right) \\ &= 2h \left(\left(1 - \frac{2}{3} - \frac{1}{6} \right) f(a) + \frac{2}{3} f(a+h) + \frac{1}{6} f(a+2h) \right) \\ &= \frac{h}{3} (f(a) + 4f(a+h) + f(a+2h)). \end{aligned}$$

For $n = 3$ gir Teorem 11.4

$$\begin{aligned} \begin{pmatrix} h^1 p_3^{(1)}(a)/1! \\ h^2 p_3^{(2)}(a)/2! \\ h^3 p_3^{(3)}(a)/3! \end{pmatrix} &= \begin{pmatrix} 1 & 1 & 1 \\ 2 & 4 & 8 \\ 3 & 9 & 27 \end{pmatrix}^{-1} \begin{pmatrix} f(a+h) - f(a) \\ f(a+2h) - f(a) \\ f(a+3h) - f(a) \end{pmatrix} \\ &= \begin{pmatrix} 3 & -3/2 & 1/3 \\ -5/2 & 2 & -1/2 \\ 1/2 & -1/2 & 1/6 \end{pmatrix} \begin{pmatrix} f(a+h) - f(a) \\ f(a+2h) - f(a) \\ f(a+3h) - f(a) \end{pmatrix} \end{aligned}$$

Setter vi inn dette i uttrykket får vi

$$\begin{aligned}
 & 3h \left(f(a) + \begin{pmatrix} \frac{3}{2} & 3 & \frac{27}{4} \end{pmatrix} \begin{pmatrix} 3 & -3/2 & 1/3 \\ -5/2 & 2 & -1/2 \\ 1/2 & -1/2 & 1/6 \end{pmatrix} \begin{pmatrix} f(a+h) - f(a) \\ f(a+2h) - f(a) \\ f(a+3h) - f(a) \end{pmatrix} \right) \\
 &= 3h \left(f(a) + \begin{pmatrix} \frac{3}{8} & \frac{3}{8} & \frac{1}{8} \end{pmatrix} \begin{pmatrix} f(a+h) - f(a) \\ f(a+2h) - f(a) \\ f(a+3h) - f(a) \end{pmatrix} \right) \\
 &= 3h \left(\left(1 - \frac{3}{8} - \frac{3}{8} - \frac{1}{8} \right) f(a) + \frac{3}{8} f(a+h) + \frac{3}{8} f(a+2h) + \frac{1}{8} f(a+3h) \right) \\
 &= \frac{3h}{8} (f(a) + 3f(a+h) + 3f(a+2h) + f(a+3h)).
 \end{aligned}$$