

Introduksjon til maskinlæring

Bildegjenkjenning med Python og Tensorflow

Esten H. Leonardsen
23.09.24



UNIVERSITETET
I OSLO

Plan for dagen

Teori:

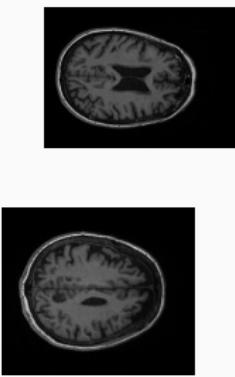
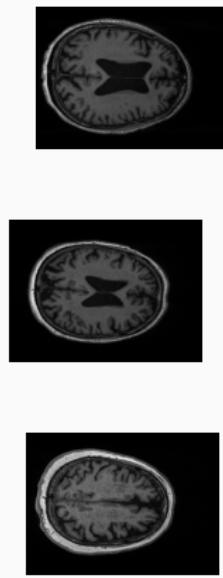
- Hva er en statistisk læringsmodell?
- Hva er en kost-funksjon?
- Hvordan trener vi en statistisk læringsmodell?
- Hvordan fungerer et (dypt) kunstig nevralgt nettverk?
- Hvordan fungerer et konvolusjonelt nevralgt nettverk (for klassifikasjon)?
- Hva er transfer learning?
- Hva er overtilpasning, og hvordan unngår vi det?

Praktisk workshop:

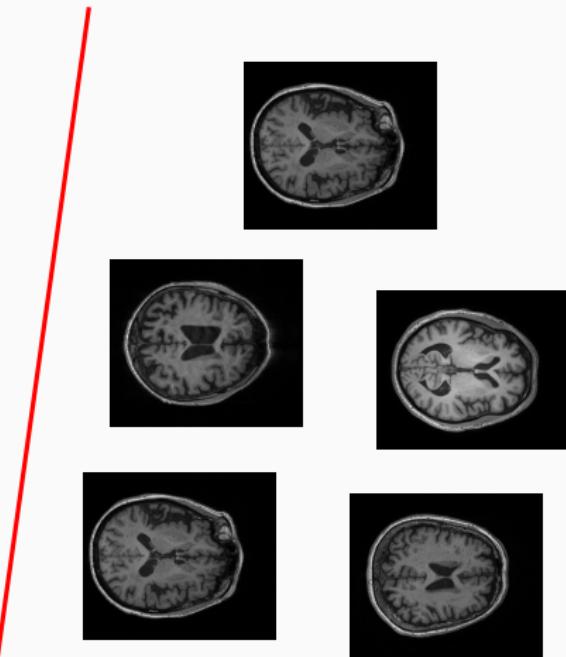
1. Sette opp et Python-miljø i Google Colab
2. Predikere med et ferdigrent konvolusjonelt nevralgt nettverk
3. Retrene en klassifikator for blomsterarter
4. Hvis tid, forbedre klassifikatoren



Motivasjon



Kontroller



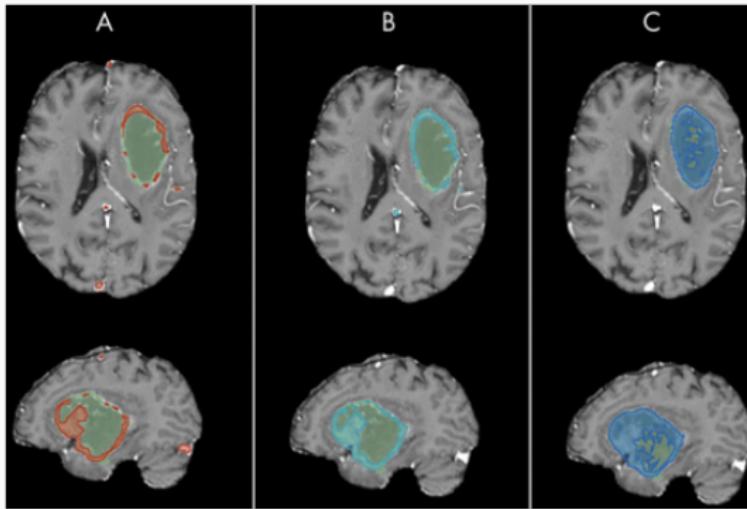
Pasienter

https://drive.google.com/file/d/10bPNMY0E8puFZaiJvejpZb9y_X3nw1c4/view?usp=sharing



Motivasjon





Motivasjon



Imagen

Statistisk læring



UNIVERSITETET
I OSLO

Statistisk læring: Modeller

Finn Mulighetens marked

Vælgere | My profile | Log in | Log out

Søgefelt

- [Vælgere og boliger \(3\)](#)
- [Nyboliger \(1\)](#)
- [Sælge \(1\)](#)

Søgesort

- [Til sælge \(750\)](#)
- [Sælges i 3 dager \(9\)](#)
- [Kommer for sælge \(1\)](#)

Nyhedsliste

- [Brugt bolig \(127\)](#)
- [Nyhøjt \(217\)](#)

Prisinterval

- [Ingen \(1\)](#)
- [10 M \(1\)](#)

Søk

Tidspunkt

- [I dag \(1\)](#)
- [10 d \(1\)](#)

Søk

Familiegrupper per måned

- [Ingen \(1\)](#)
- [10 M \(1\)](#)

Søk

Størrelse

- [Ingen \(1\)](#)
- [10 M \(1\)](#)

Søk

Antal soverom

- [1 \(1\)](#)
- [2 \(1\)](#)
- [3+ \(1\)](#)
- [4+ \(1\)](#)
- [5+ \(1\)](#)

Søk

Bælgar

- [Ingen \(1\)](#)
- [10 M \(1\)](#)

Søk

Boligtyper

- [Lejlighed \(544\)](#)
- [Garage/Parkeringsplads \(11\)](#)
- [Rekkehus \(2\)](#)
- [Endebolig \(1\)](#)
- [Terrassebolig \(2\)](#)
- [Lægenbolig \(2\)](#)
- [Andre \(2\)](#)

Utsikt

Kategorier

- [Aktie \(5\)](#)
- [Andel \(124\)](#)
- [Car \(Gåvobolig\) \(795\)](#)

Læres - Tilgangen 3-roms med god planlæsning og flott beliggenhet - Sørlig...

73 år • 6400 000 kr

Tilgang: 0 100 200 m • Tidspunkt: 10 d (1) • Øvr. (Sælger) • Lejlighed - 2 værelser

Varing - 22 januar kl 10:30

Læres - Et tilgangen 3-roms med god planlæsning og flott beliggenhet - Sørlig...

73 år • 6 400 000 kr

Tilgang: 0 100 200 m • Tidspunkt: 10 d (1) • Øvr. (Sælger) • Lejlighed - 2 værelser

Varing - 22 januar kl 10:30

SP - Storleks / Grønne lokale - Lys, luftig 3-roms hjemmelejl med indre gård ...

67 år • 5 500 000 kr

Tilgang: 0 600 127 m • Tidspunkt: 4 002 d (1) • Øvr. (Sælger) • Lejlighed - 2 værelser

Varing - 22 januar kl 15:30

Rødekket - Rødekket - 2-roms lejlighed med stor terrasse og god beliggenhet - Hels...

46 m² • 4 100 000 kr

Tilgang: 0 210 742 m • Tidspunkt: 3 474 d (1) • Øvr. (Sælger) • Lejlighed - 1 værelse

Varing - 22 januar kl 15:30

Grønne lokale - Lys 2-roms med stort potensial Nordvest balkong - Hels...

46 m² • 4 100 000 kr

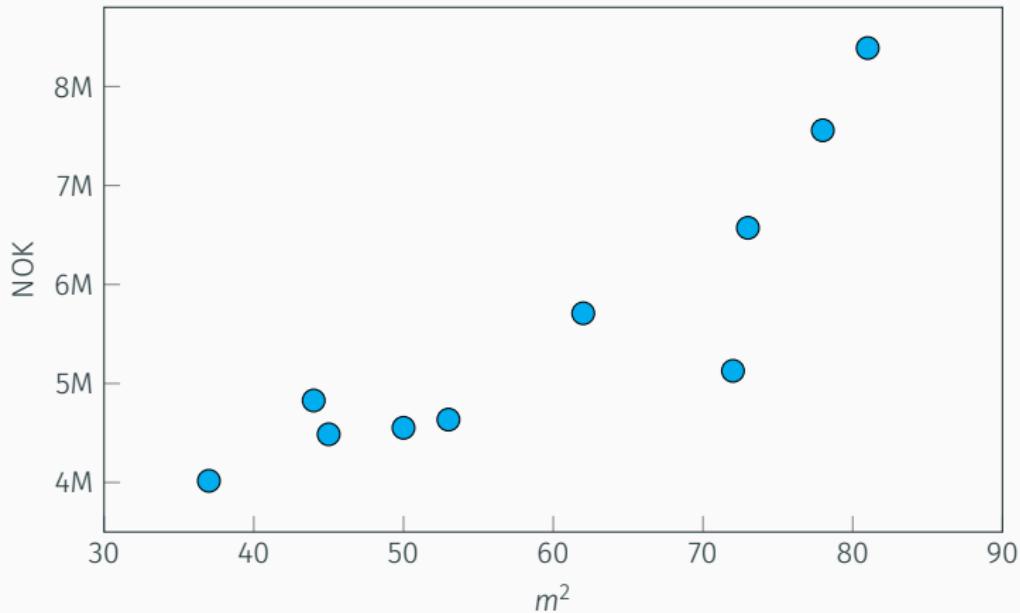
Tilgang: 0 210 742 m • Tidspunkt: 3 474 d (1) • Øvr. (Sælger) • Lejlighed - 1 værelse

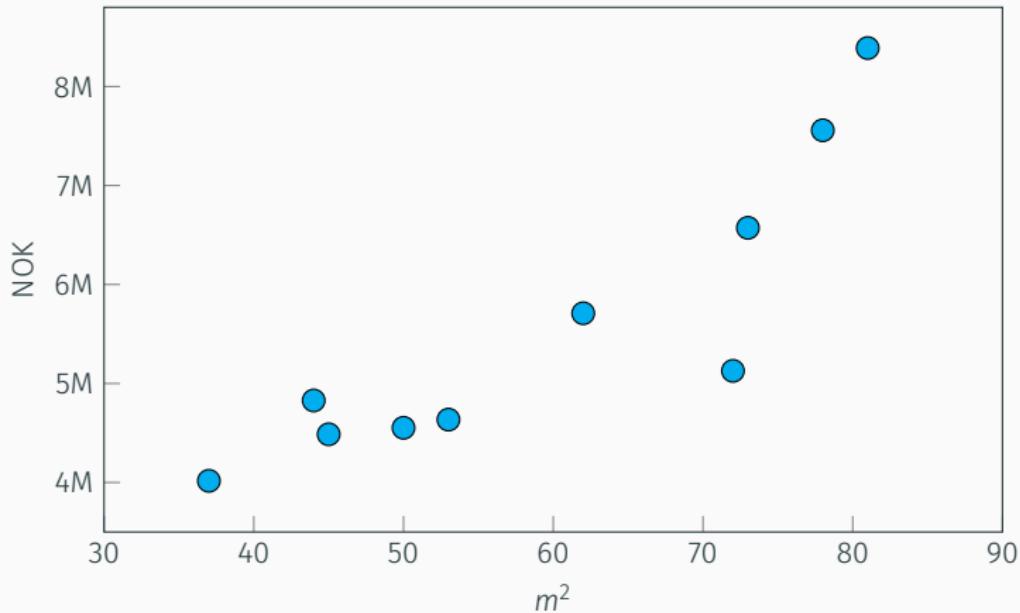
Varing - 22 januar kl 15:30



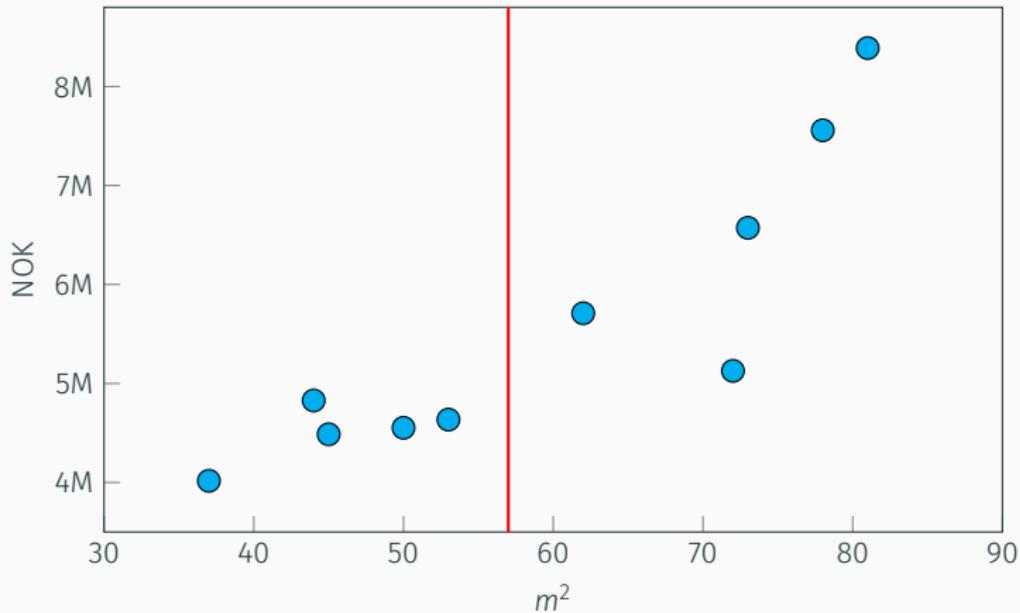
m^2	Price
72	5.127.379
50	4.552.170
45	4.486.654
62	5.709.276
53	4.634.912
81	8.388.570
44	4.828.170
78	7.557.770
37	4.016.520
73	6.572.351







$$\hat{y} = f(x)$$



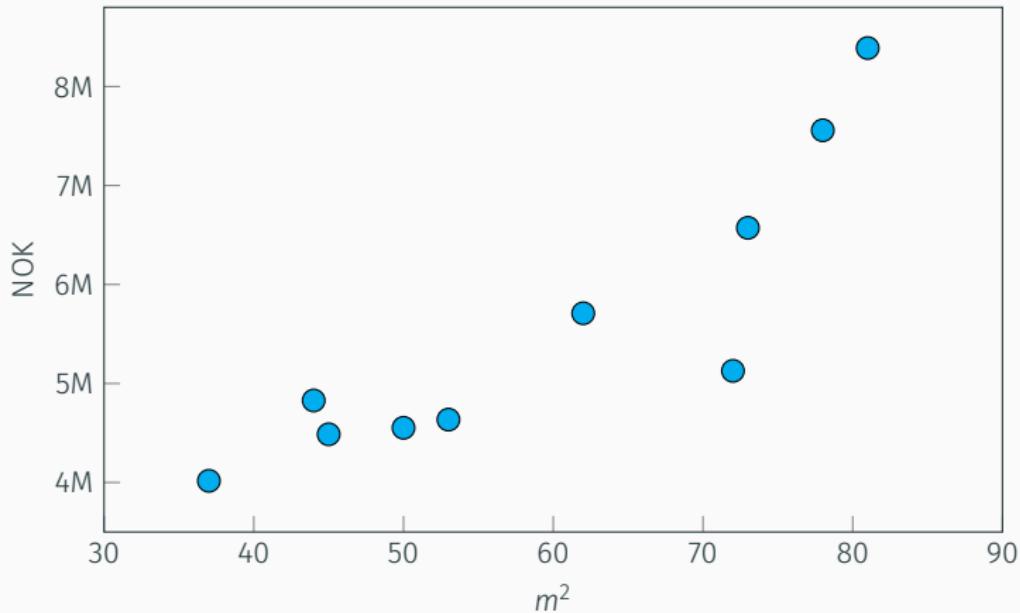
$$\hat{y} = f(\textcolor{red}{x})$$

Hva er en statistisk læringsmodell?

- En matematisk konstruksjon (ofte en formel) som representerer forholdet mellom input x og output y
- En funksjonell enhet som, gitt ny data x , er i stand til å predikere nye utfall y



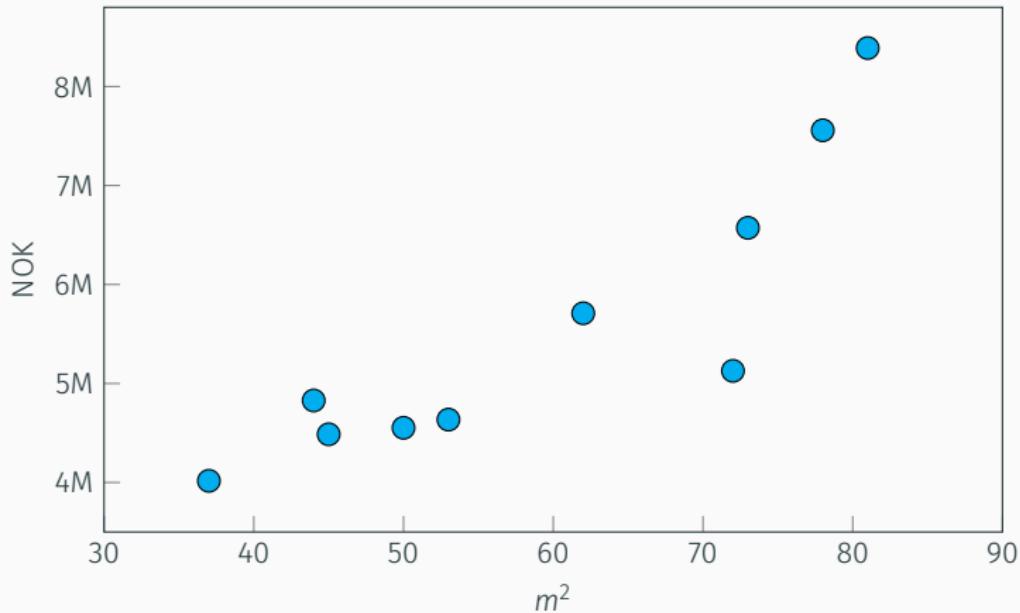
Statistisk læring: Kost-funksjoner



$$\hat{y} = b + wx$$

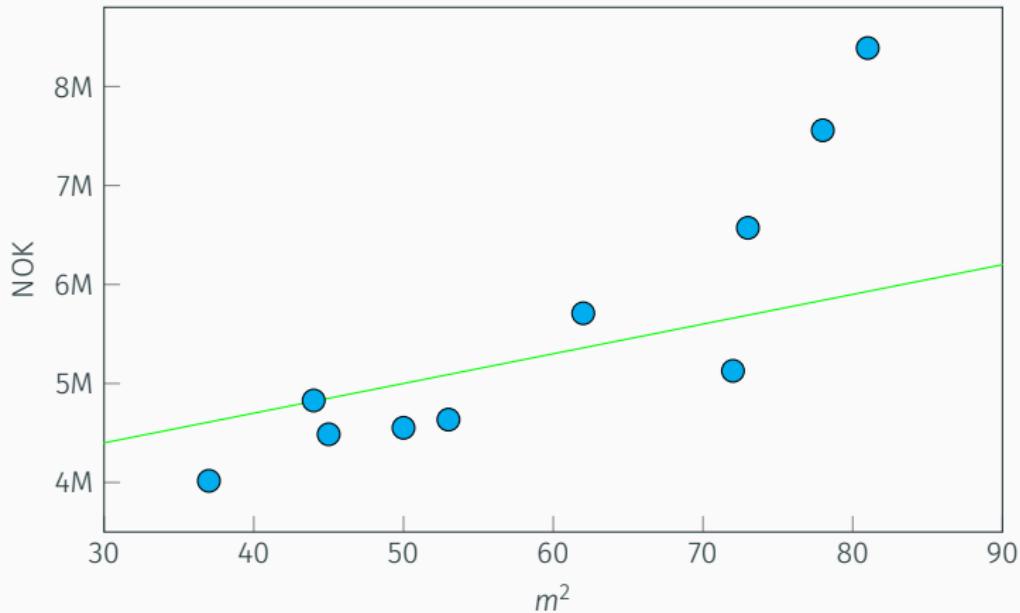


Statistisk læring: Kost-funksjoner



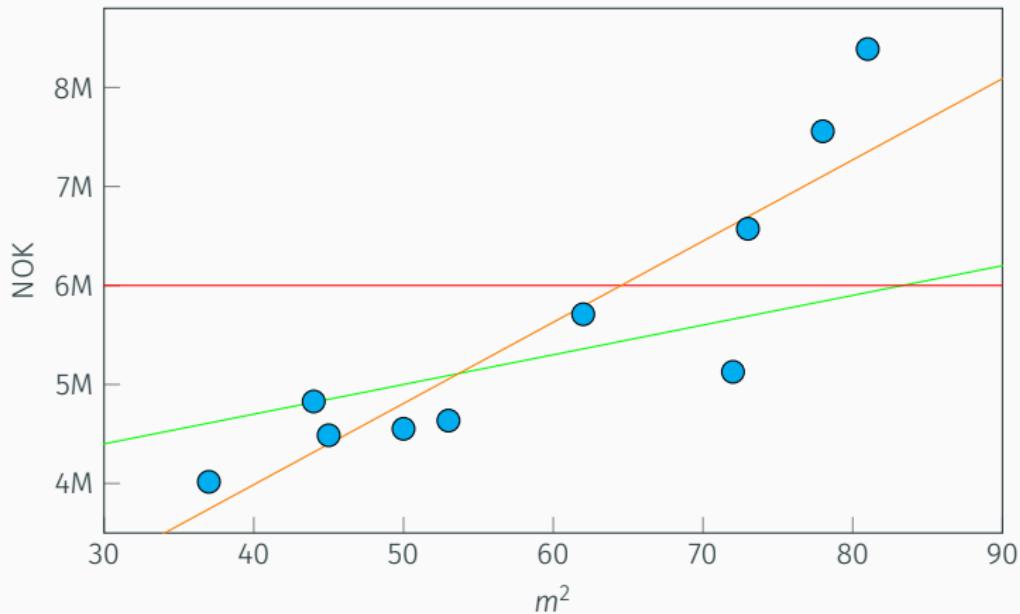
$$\hat{y} = \textcolor{red}{b} + \textcolor{red}{w}x$$





$$\hat{y} = 3500000 + 30000x$$

Statistisk læring: Kost-funksjoner

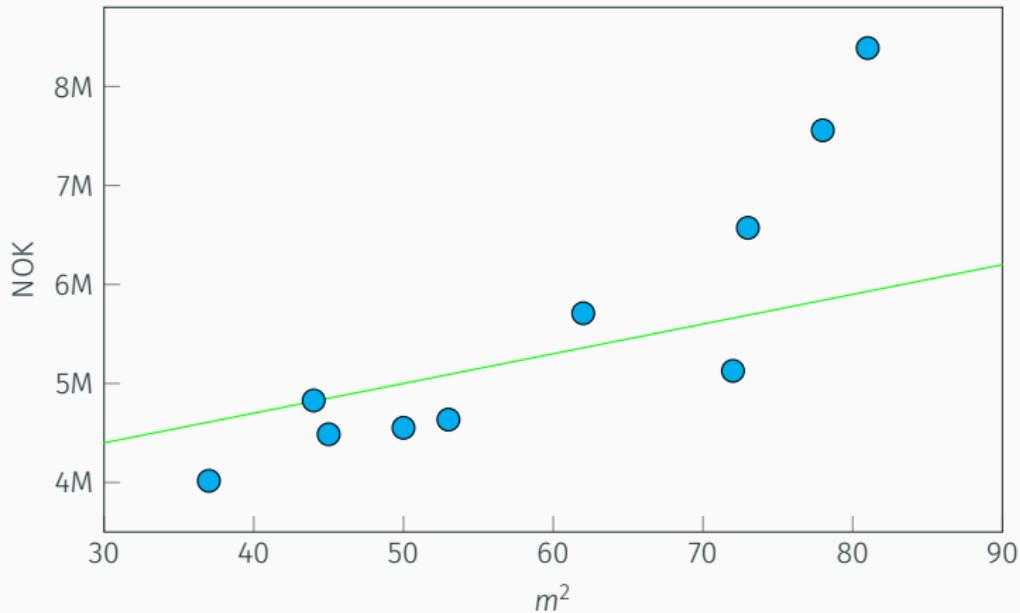


$$\hat{y} = 6000000 + 0x$$

$$\hat{y} = 3500000 + 30000x$$

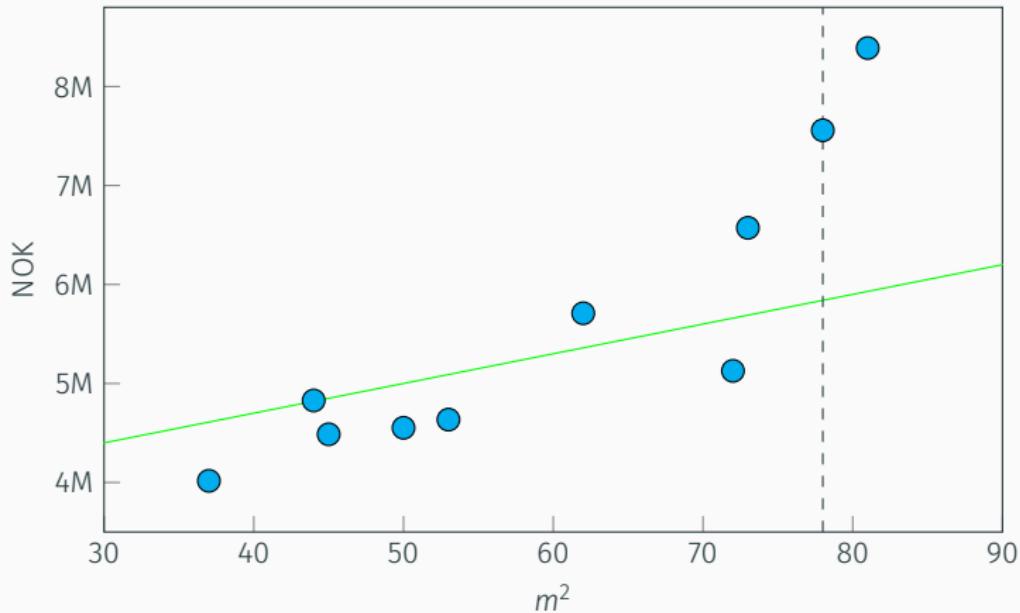
$$\hat{y} = 706495 + 82031x$$





$$\hat{y} = 3500000 + 30000x$$

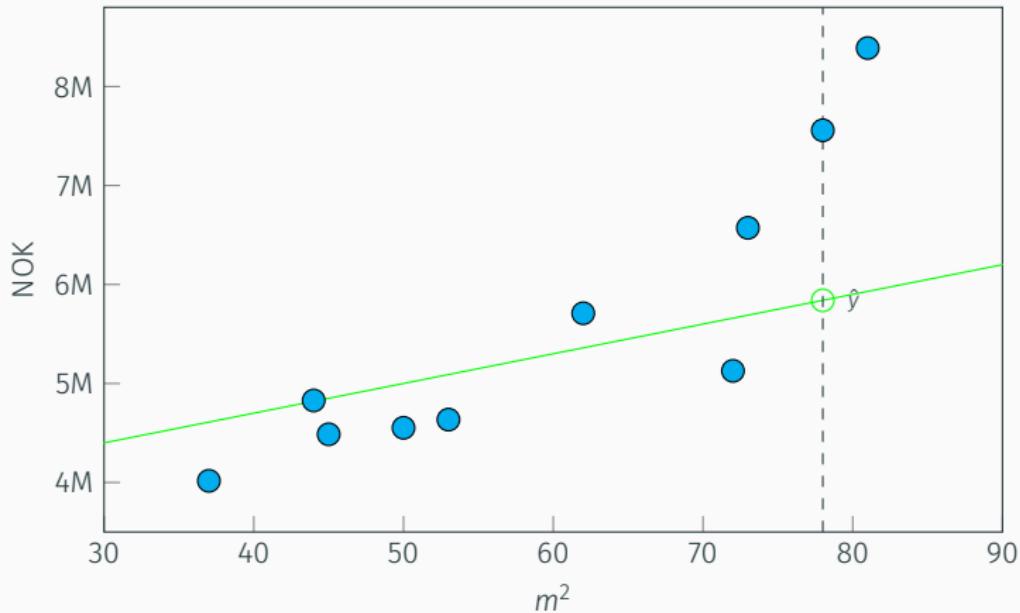
Statistisk læring: Kost-funksjoner



$$\hat{y} = 3500000 + 30000 * 78$$



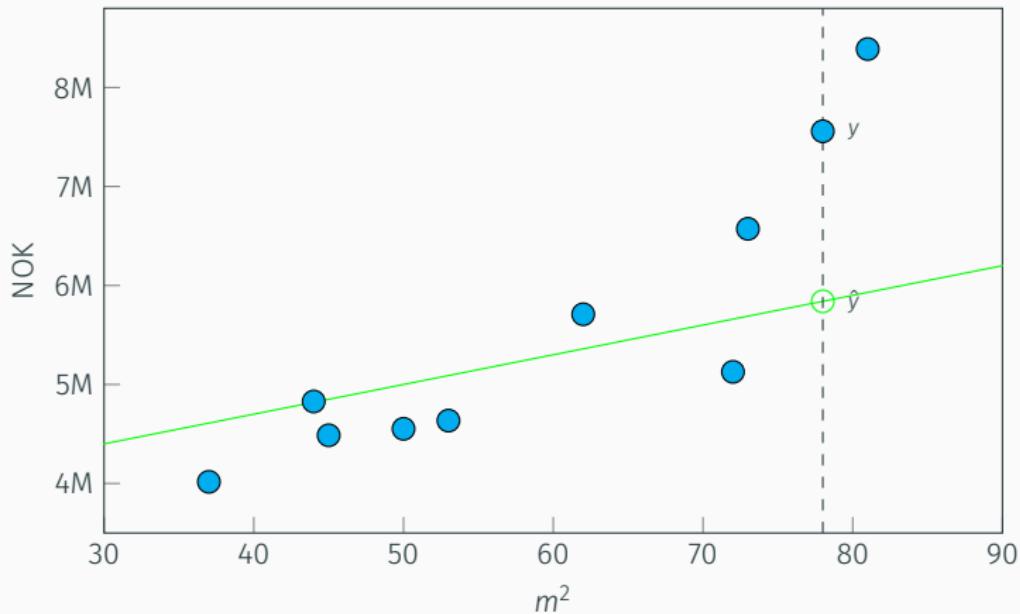
Statistisk læring: Kost-funksjoner



$$\hat{y} = 3500000 + 30000 * 78$$

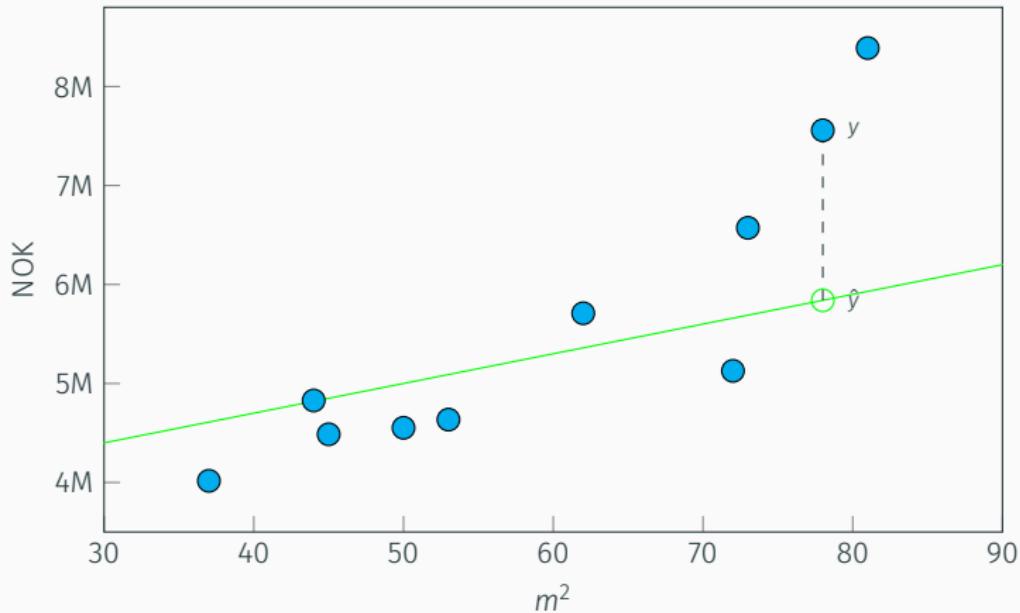


Statistisk læring: Kost-funksjoner



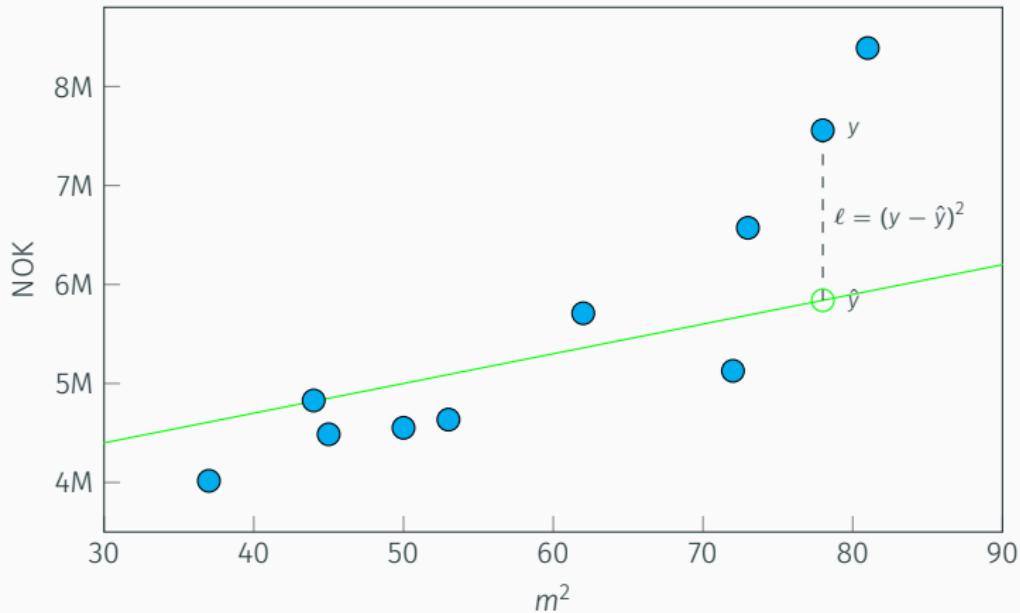
$$\hat{y} = 3500000 + 30000 * 78$$





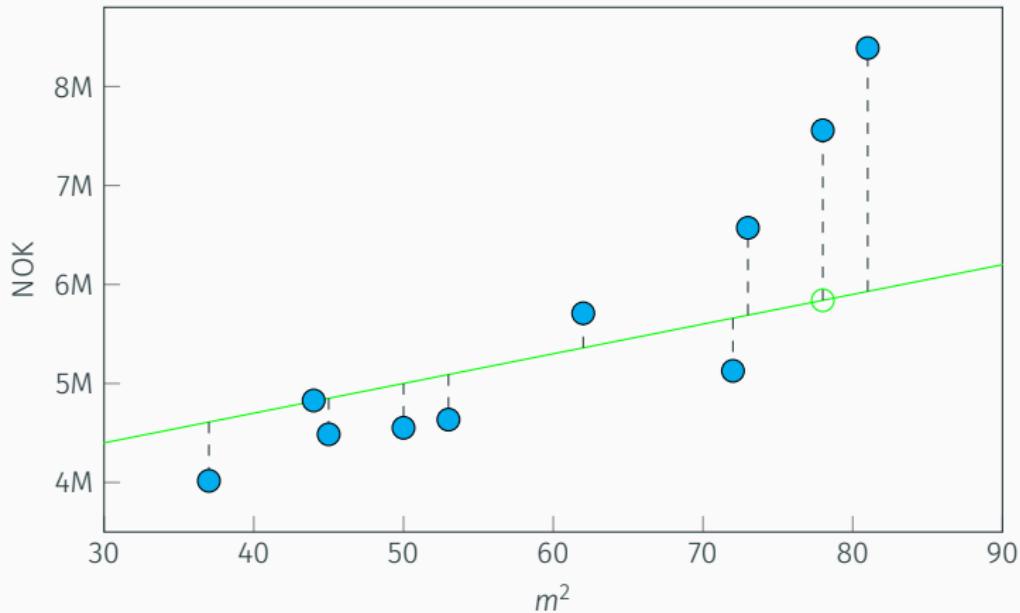
$$\hat{y} = 3500000 + 30000 * 78$$

Statistisk læring: Kost-funksjoner



$$\hat{y} = 3500000 + 30000 * 78$$

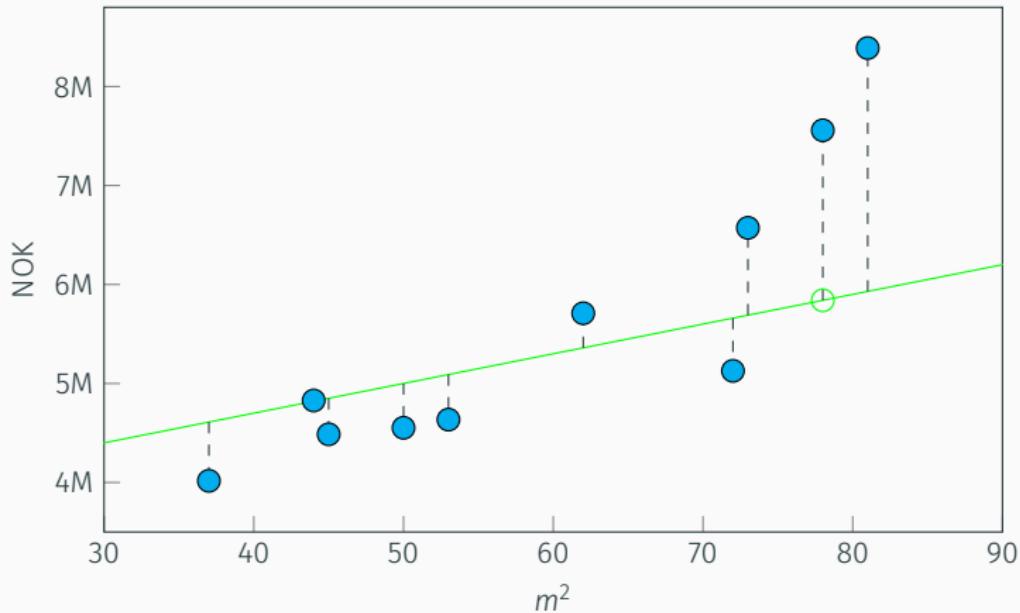




$$\hat{y} = 3500000 + 30000x$$

$$\ell = \frac{1}{N} \sum (y - \hat{y})^2$$

Statistisk læring: Kost-funksjoner

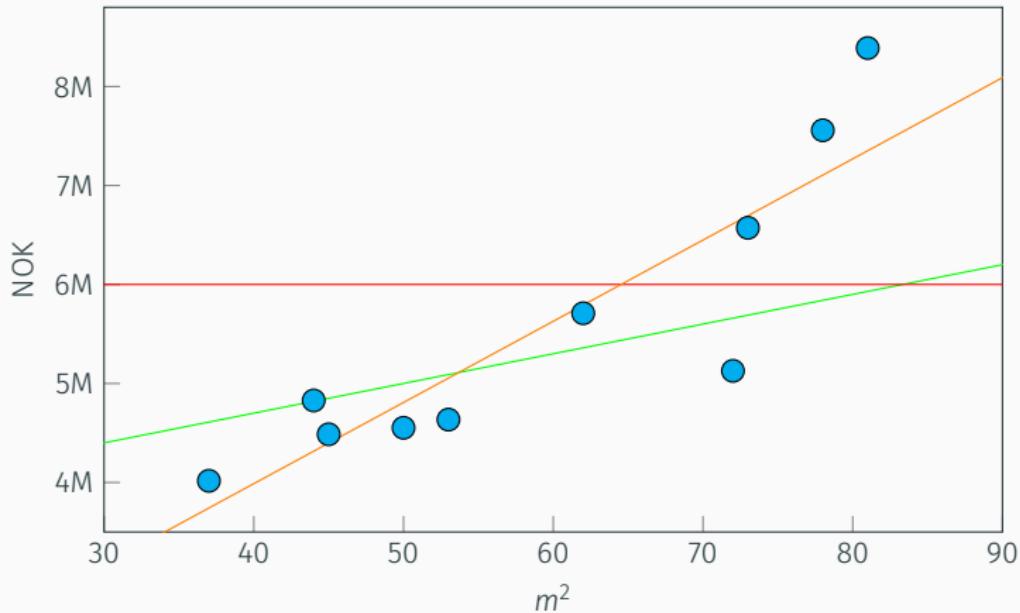


$$\hat{y} = 3500000 + 30000x$$

$$\ell = 1.10 \times 10^{13}$$



Statistisk læring: Kost-funksjoner



$$\hat{y} = 6000000 + 0x$$
$$\ell = 2.08 \times 10^{13}$$

$$\hat{y} = 3500000 + 30000x$$
$$\ell = 1.10 \times 10^{13}$$

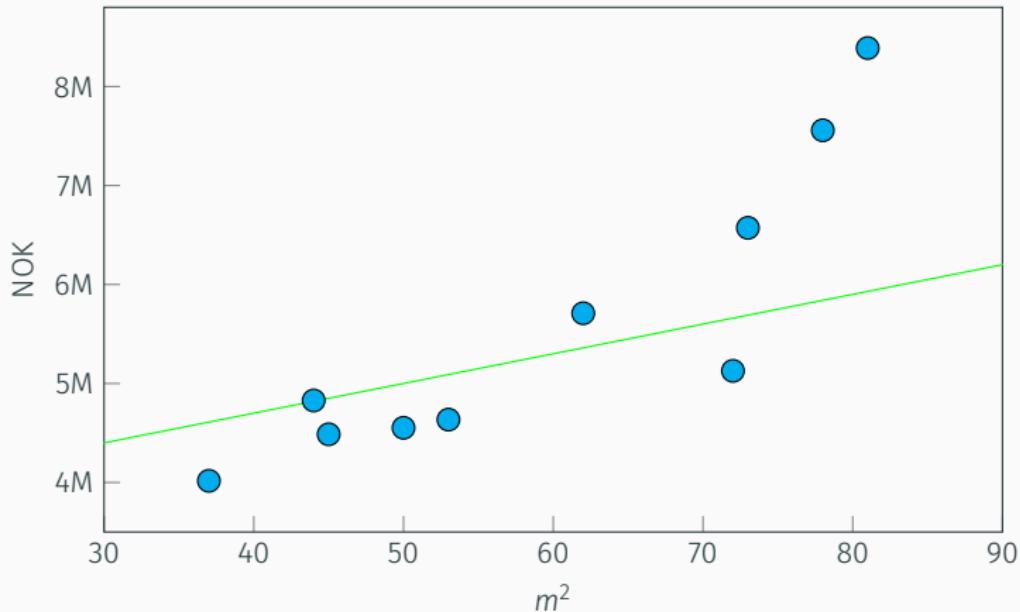
$$\hat{y} = 706495 + 82031x$$
$$\ell = 4.09 \times 10^{12}$$



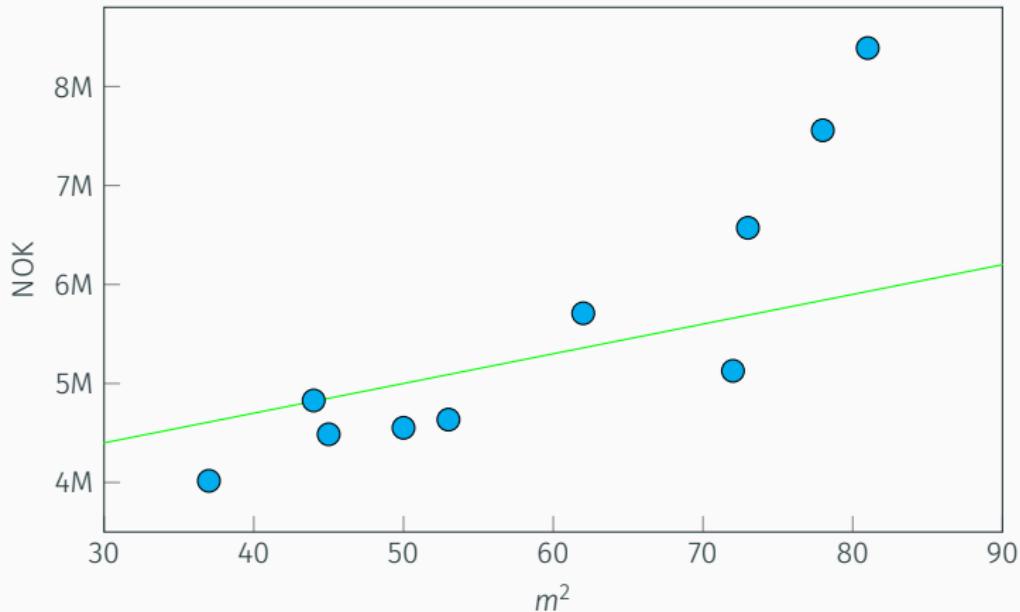
Hva er en kost-funksjon?

- En matematisk funksjon som beregner kostnaden av å bruke en gitt modell for å forklare et gitt datasett
- En formalisering av oppgaven vi ønsker at modellen skal løse



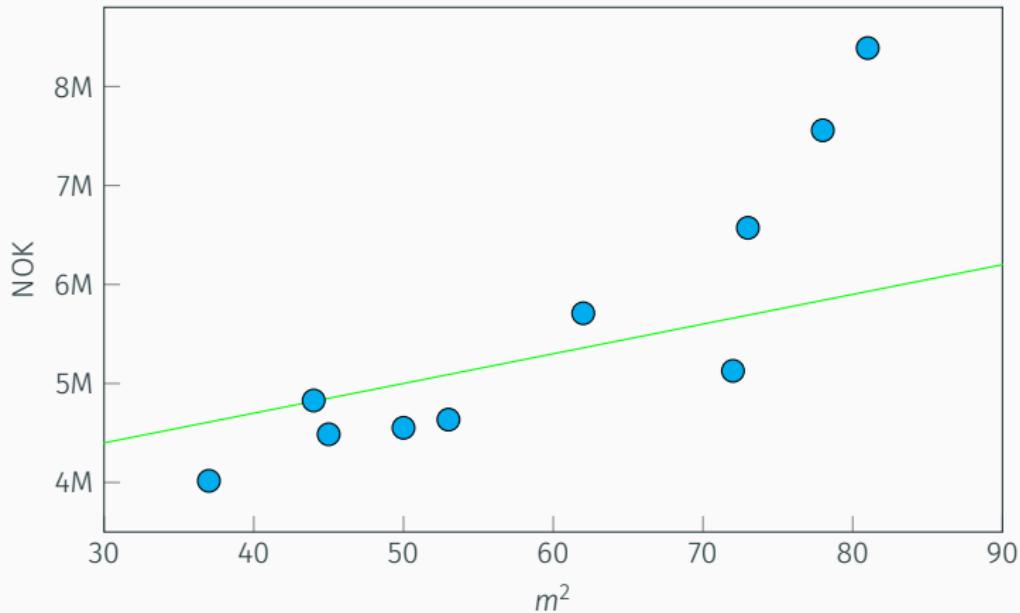


$$\hat{y} = b + wx$$
$$\ell = \frac{1}{N} \sum (y - \hat{y})^2$$

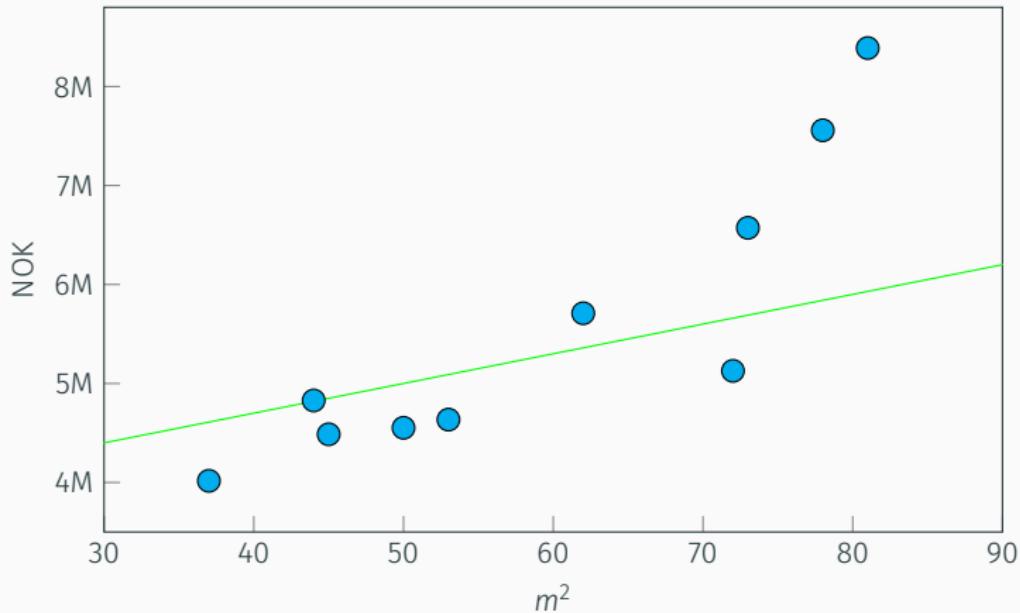


$$\hat{y} = b + wx$$

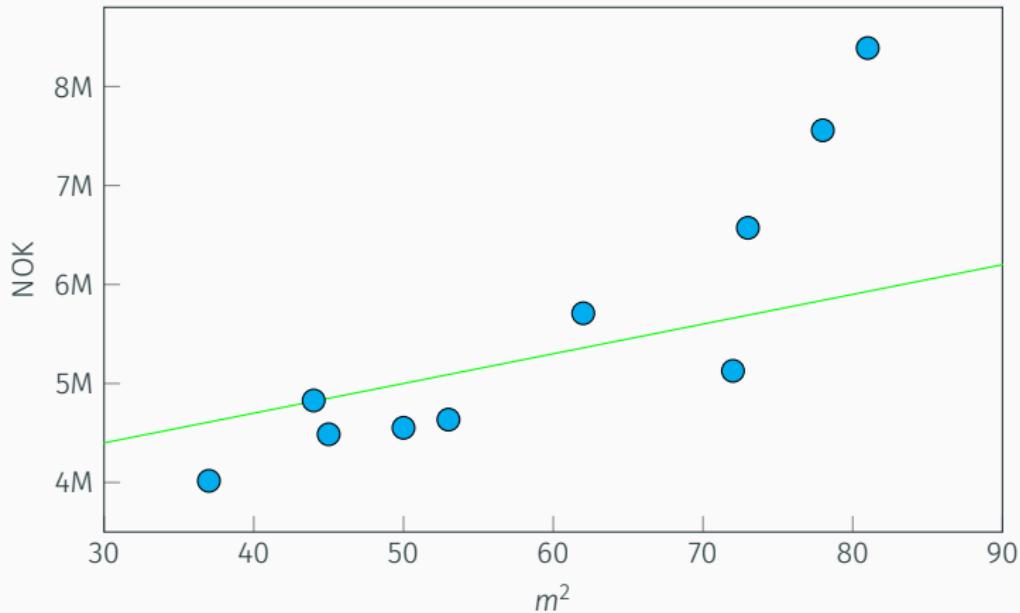
$$\ell = \frac{1}{N} \sum (y - \hat{y})^2$$



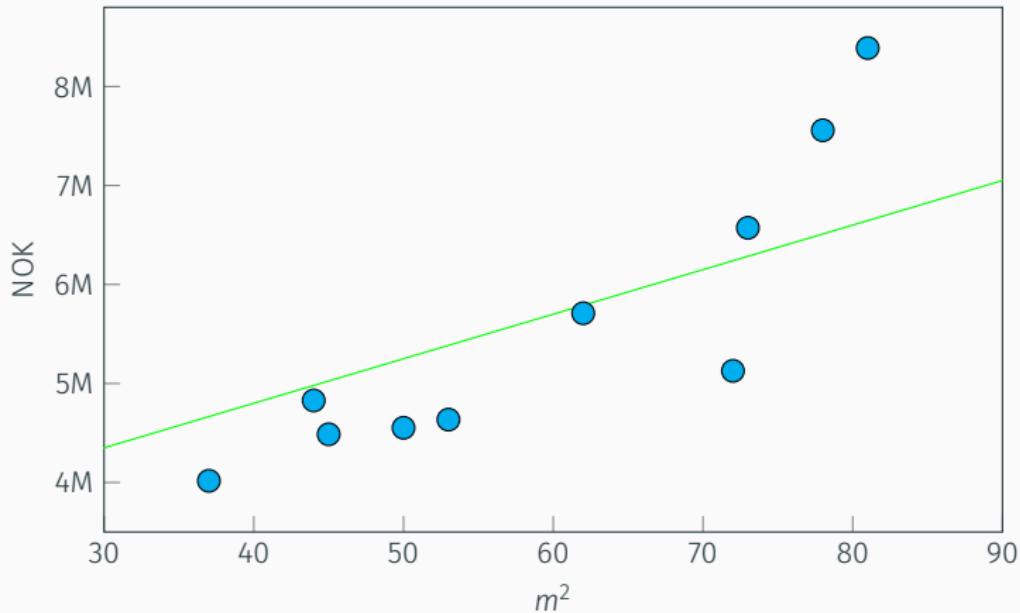
$$\ell = \frac{1}{N} \sum (y - (b + wx))^2$$



$$1.10 \times 10^{13} = \frac{1}{N} \sum (y - (3500000 + 30000x))^2$$



$$1.10 \times 10^{13} = \frac{1}{N} \sum (y - (3500000 + 30000x))^2$$



$$7.24 \times 10^{12} = \frac{1}{N} \sum (y - (3000000 + 45000x))^2$$



Hvordan trener vi en statistisk læringsmodell?

1. Estimer prediksjoner for alle datapunkter
2. Beregn kostnaden av disse prediksjonene (ved å sammenligne med fasit)
3. Kalkuler hvordan parametrene i modellen påvirker kostnaden
4. Oppdater parametrene i den retningen som minsker kostnaden

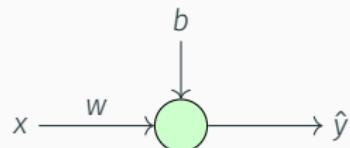


Kunstige nevrale nettverk



UNIVERSITETET
I OSLO

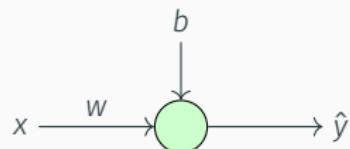
Kunstige nevrale nettverk: Oppbygning



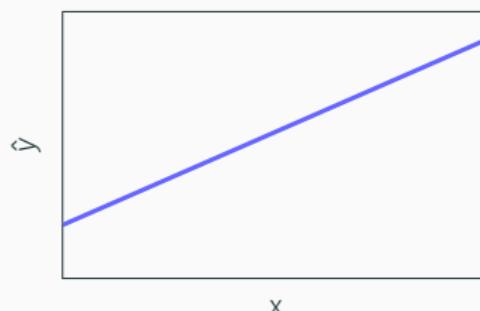
$$\hat{y} = wx + b$$



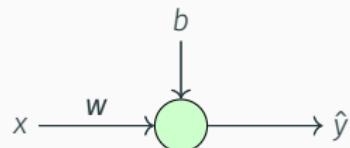
Kunstige nevrale nettverk: Oppbygning



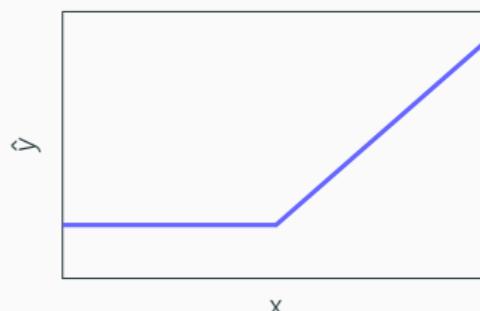
$$\hat{y} = wx + b$$



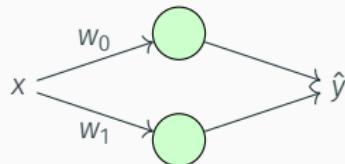
Kunstige nevrale nettverk: Oppbygning



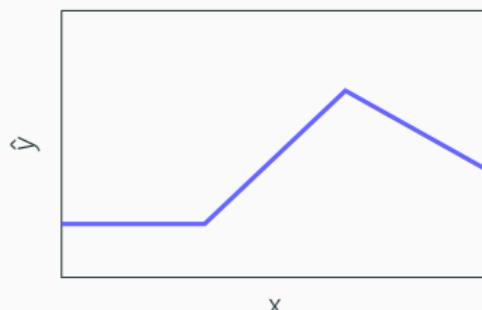
$$\hat{y} = \max(0, wx + b)$$



Kunstige nevrale nettverk: Oppbygning



$$\hat{y} = \max(0, w_0x + b_0) + \max(0, w_1x + b_1)$$

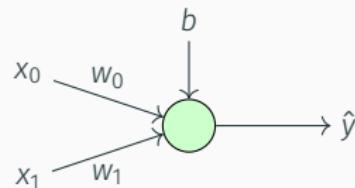


Universal approximation theorem:

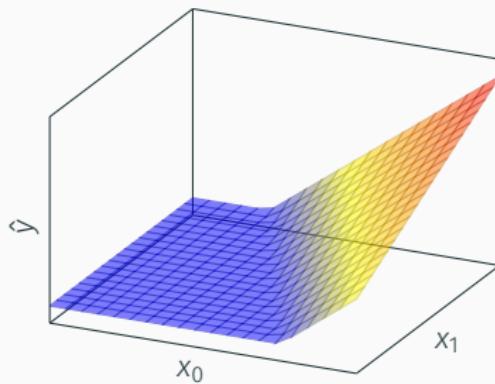
"Any relationship that can be described with a polynomial function can be approximated by a neural network with a single hidden layer."



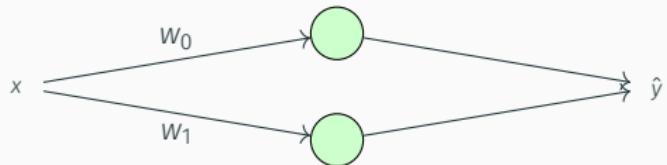
Kunstige nevrale nettverk: Oppbygning



$$\hat{y} = \max(0, w_0x_0 + w_1x_1 + b)$$



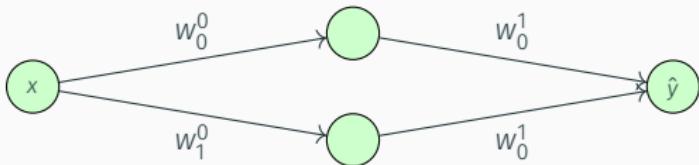
Kunstige nevrale nettverk: Oppbygning



$$\hat{y} = \max(0, w_0x + b_0) + \max(0, w_1x + b_1)$$



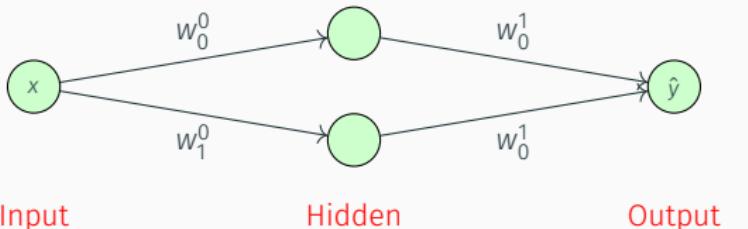
Kunstige nevrale nettverk: Oppbygning



$$\hat{y} = \max(0, w_{0,0}^1 * \max(0, w_{0,0}^0 * x + b_{0,0}) + w_{1,0}^1 * \max(0, w_{0,1}^0 * x + b_{1,0}) + b_1)$$



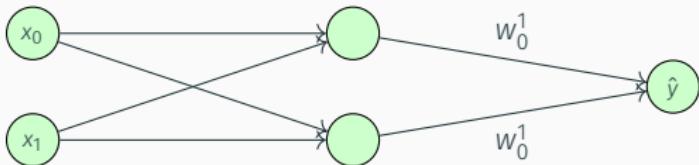
Kunstige nevrale nettverk: Oppbygning



$$\hat{y} = \max(0, w_{0,0}^1 * \max(0, w_{0,0}^0 * x + b_{0,0}) + w_{1,0}^1 * \max(0, w_{0,1}^0 * x + b_{1,0}) + b_1)$$



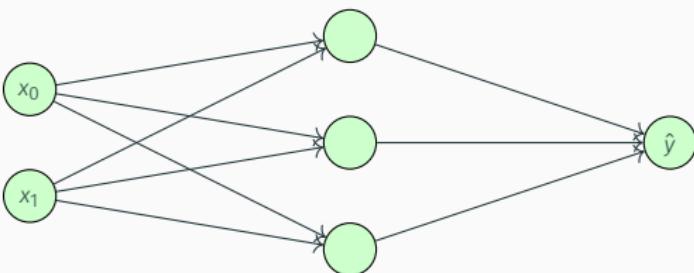
Kunstige nevrale nettverk: Oppbygning



$$\begin{aligned}\hat{y} = & \max(0, w_{0,0}^1 * \max(0, w_{0,0}^0 * x_0 + w_{1,0}^0 * x_1 + b_{0,0}) + \\ & w_{1,0}^1 * \max(0, w_{0,1}^0 * x_0 + w_{1,1}^0 * x_1 + b_{0,1}) + \\ & b_1)\end{aligned}$$



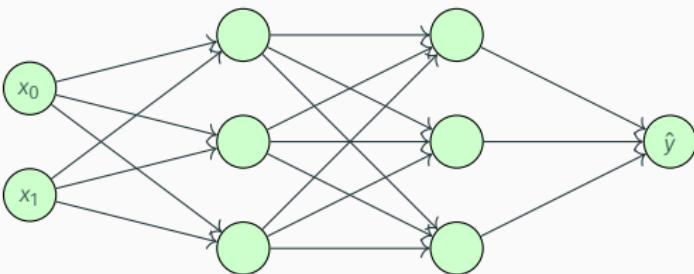
Kunstige nevrale nettverk: Oppbygning



$$\begin{aligned}\hat{y} = & \max(0, w_{0,0}^1 * \max(0, w_{0,0}^0 * x_0 + w_{1,0}^0 * x_1 + b_{0,0}) + \\& w_{1,0}^1 * \max(0, w_{0,1}^0 * x_0 + w_{1,1}^0 * x_1 + b_{0,1}) + \\& w_{2,0}^1 * \max(0, w_{0,2}^0 * x_0 + w_{1,2}^0 * x_1 + b_{0,2}) + \\& b_1)\end{aligned}$$



Kunstige nevrale nettverk: Oppbygning



$$\hat{y} = \max(0, w_{0,0}^2 * \max(0, w_{0,0}^1 * \max(0, w_{0,0}^0 * x_0 + w_{1,0}^0 * x_1 + b_{0,0}) + \\ w_{1,0}^1 * \max(0, w_{0,1}^0 * x_0 + w_{1,1}^0 * x_1 + b_{0,1}) + \\ w_{2,0}^1 * \max(0, w_{0,2}^0 * x_0 + w_{1,2}^0 * x_1 + b_{0,2}) + \\ b_{1,0}) + \\ w_{1,0}^2 * \max(0, w_{0,1}^1 * \max(0, w_{0,0}^0 * x_0 + w_{1,0}^0 * x_1 + b_{0,0}) + \\ w_{1,1}^1 * \max(0, w_{0,1}^0 * x_0 + w_{1,1}^0 * x_1 + b_{0,1}) + \\ w_{2,1}^1 * \max(0, w_{0,2}^0 * x_0 + w_{1,2}^0 * x_1 + b_{0,2}) + \\ b_{1,1}) + \\ w_{2,0}^2 * \max(0, w_{0,2}^1 * \max(0, w_{0,0}^0 * x_0 + w_{1,0}^0 * x_1 + b_{0,0}) + \\ w_{1,2}^1 * \max(0, w_{0,1}^0 * x_0 + w_{1,1}^0 * x_1 + b_{0,1}) + \\ w_{2,2}^1 * \max(0, w_{0,2}^0 * x_0 + w_{1,2}^0 * x_1 + b_{0,2}) + \\ b_{1,2}) + \\ b_2)$$



Hvordan fungerer et (dypt) kunstig nevralt nettverk?

- Kunstige neuroner som hver for seg beregner en enkel, ikke-lineær funksjon kombineres i en komputasjonell graf
- Grafen som helhet kan beregne arbitrært komplekse sammenhenger mellom input og output

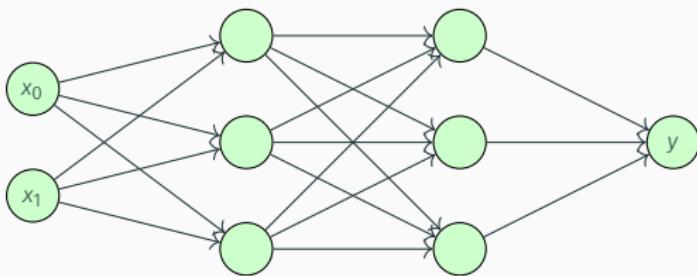


Konvolusjonelle nevrale nettverk

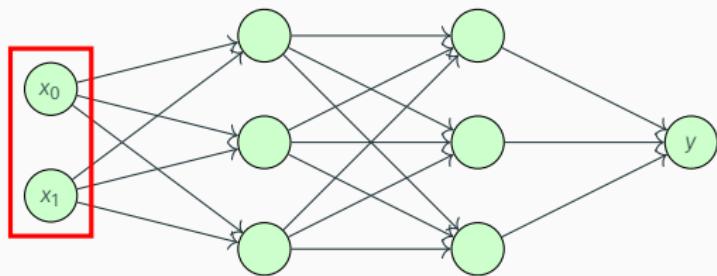


UNIVERSITETET
I OSLO

Konvolusjonelle nevrale nettverk: Bildeinput



Konvolusjonelle nevrale nettverk: Bildeinput

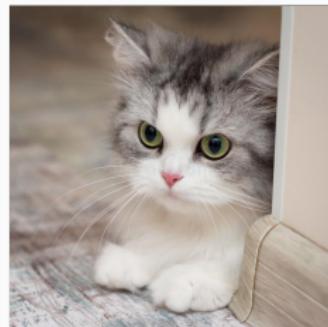


Konvolusjonelle nevrale nettverk: Bildeinput

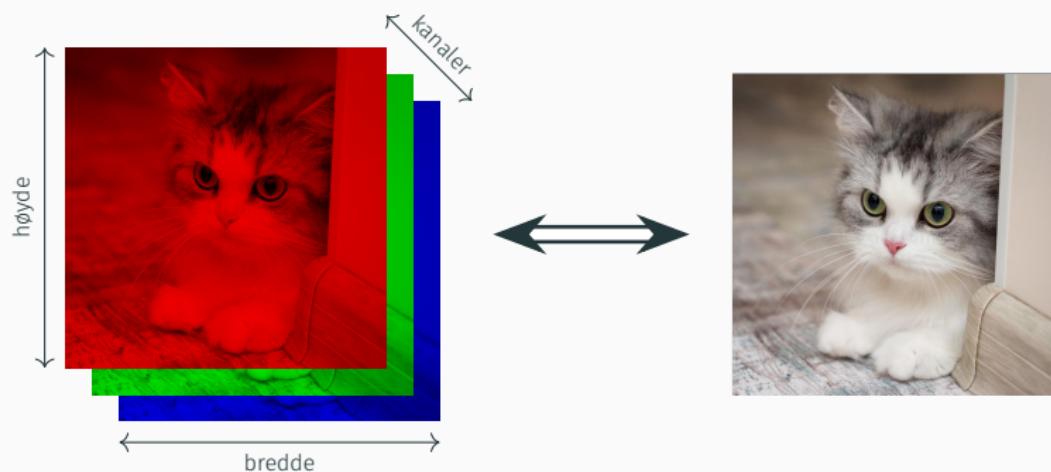
x_0

x_1

?

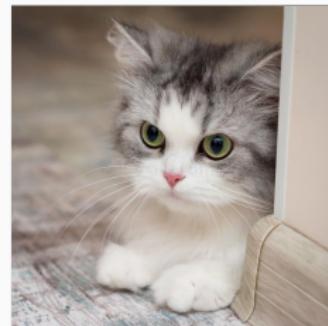


Konvolusjonelle nevrale nettverk: Bildeinput

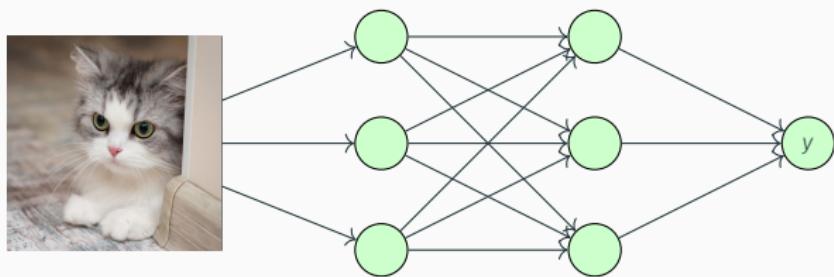


Konvolusjonelle nevrale nettverk: Bildeinput

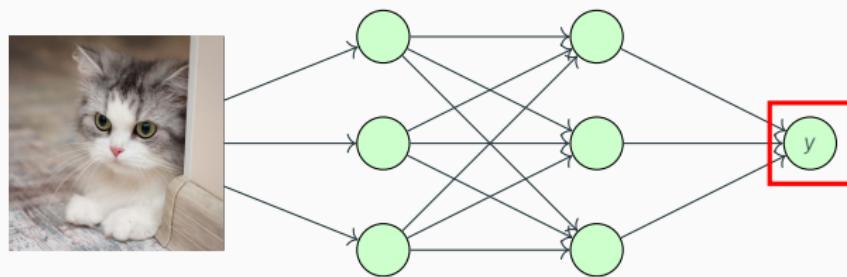
X_{000}	X_{010}	X_{020}	X_{030}	X_{040}	
X_{100}	X_{110}	X_{120}	X_{130}	X_{140}	
X_{200}	X_{210}	X_{220}	X_{230}	X_{240}	
X_{300}	X_{310}	X_{320}	X_{330}	X_{340}	
X_{400}	X_{410}	X_{420}	X_{430}	X_{440}	
					1
					2
					1
					2
					1
					2



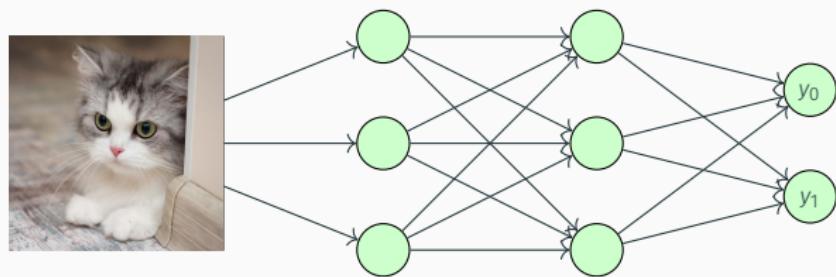
Konvolusjonelle nevrale nettverk: Bildeinput



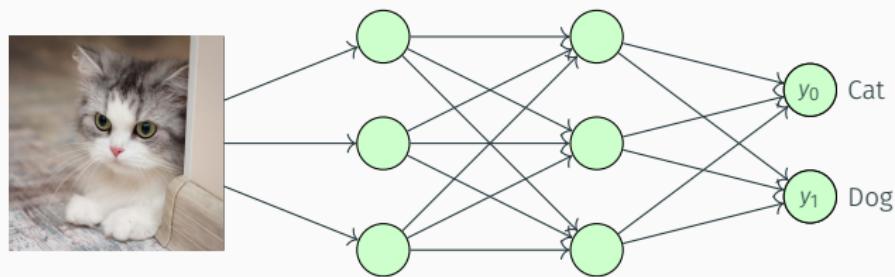
Konvolusjonelle nevrale nettverk: Klassifikasjonsoutput



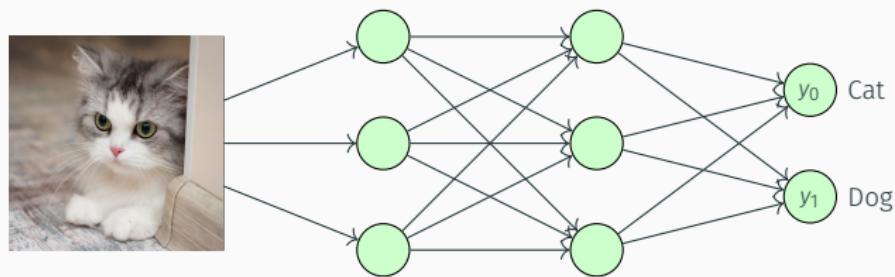
Konvolusjonelle nevrale nettverk: Klassifikasjonsoutput



Konvolusjonelle nevrale nettverk: Klassifikasjonsoutput

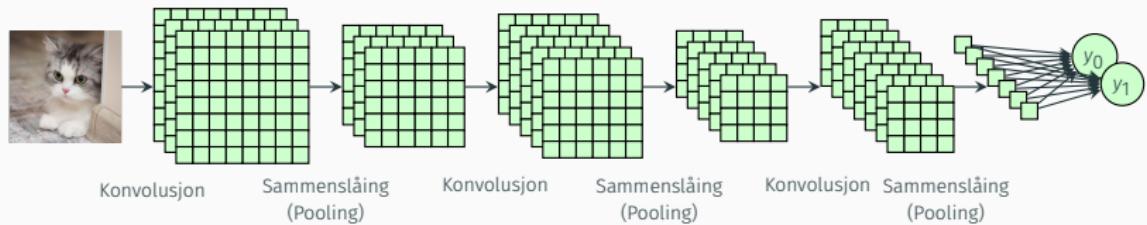


Konvolusjonelle nevrale nettverk: Klassifikasjonsoutput



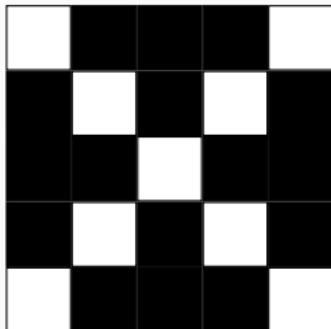
$$\text{kost} = - \sum_i y_i \log(\hat{y}_i)$$

Konvolusjonelle nevrale nettverk: Arkitektur



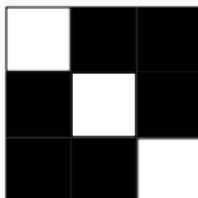
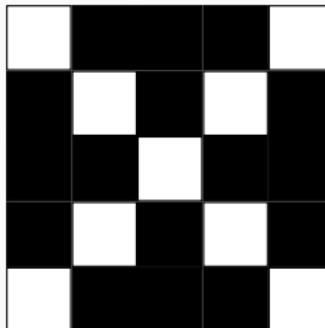
Konvolusjonelle nevrale nettverk: Konvolusjon

Bilde



Konvolusjonelle nevrale nettverk: Konvolusjon

Bilde

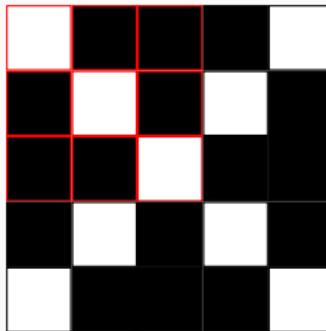


Mønster 1

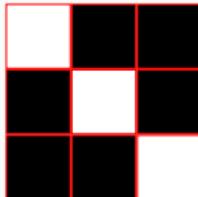


Konvolusjonelle nevrale nettverk: Konvolusjon

Bilde



3

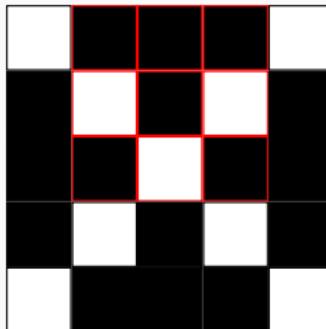


Mønster 1

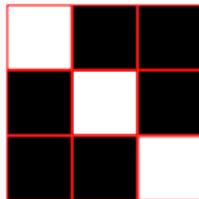


Konvolusjonelle nevrale nettverk: Konvolusjon

Bilde



3	0
---	---

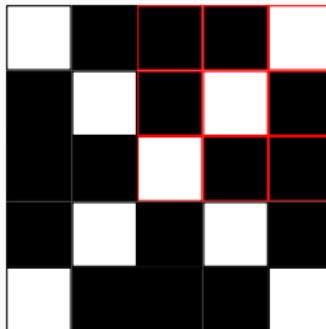


Mønster 1

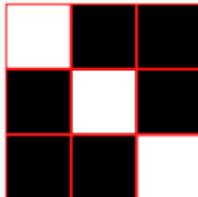


Konvolusjonelle nevrale nettverk: Konvolusjon

Bilde



3	0	1
---	---	---

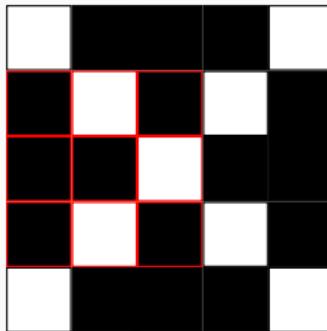


Mønster 1

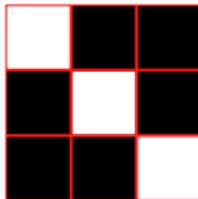


Konvolusjonelle nevrale nettverk: Konvolusjon

Bilde



3	0	1
0		

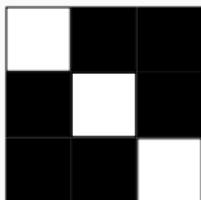
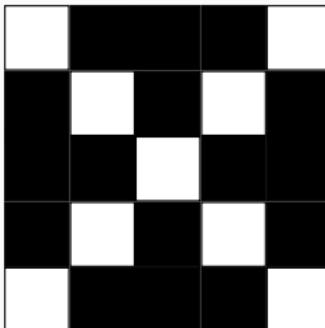


Mønster 1



Konvolusjonelle nevrale nettverk: Konvolusjon

Bilde



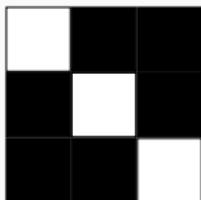
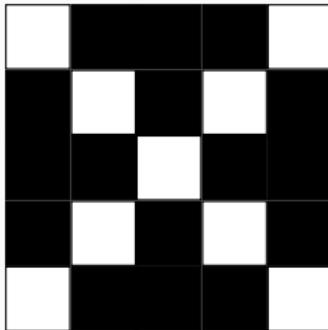
Mønster 1

3	0	1
0	3	0
1	0	3

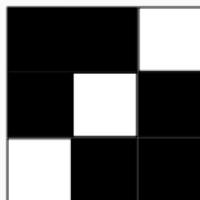


Konvolusjonelle nevrale nettverk: Konvolusjon

Bilde



Mønster 1

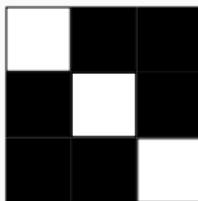
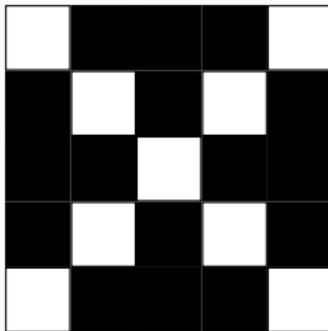


Mønster 2

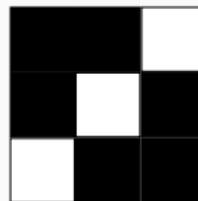
1	0	3
0	3	0
3	0	1

Konvolusjonelle nevrale nettverk: Konvolusjon

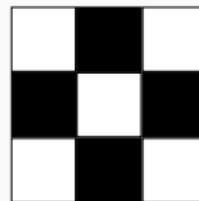
Bilde



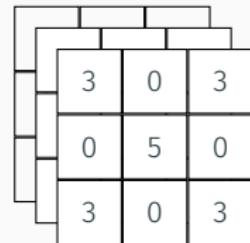
Mønster 1



Mønster 2

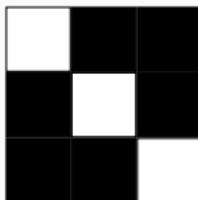
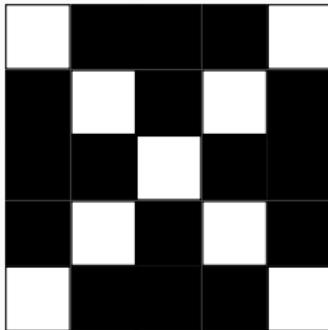


Mønster 3

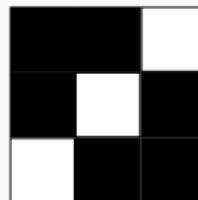


Konvolusjonelle nevrale nettverk: Konvolusjon

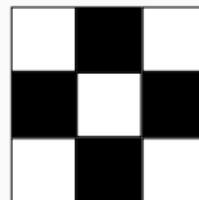
Bilde



Mønster 1

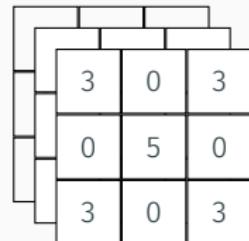


Mønster 2



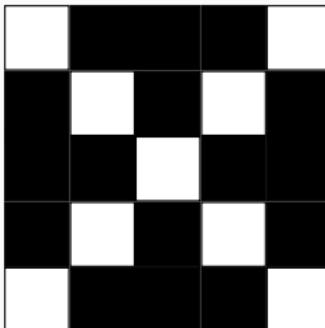
Mønster 3

Feature-kart

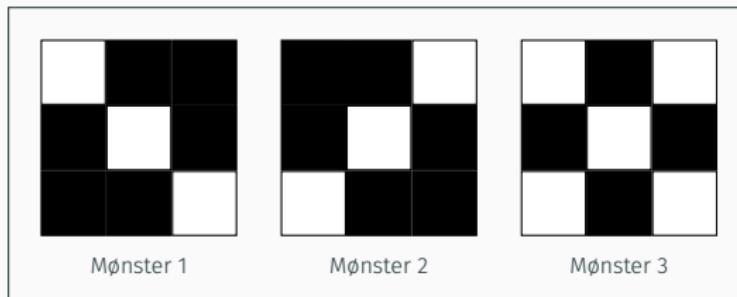
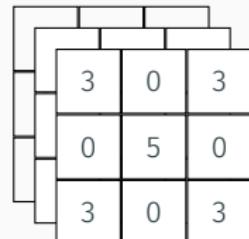


Konvolusjonelle nevrale nettverk: Konvolusjon

Bilde



Feature-kart

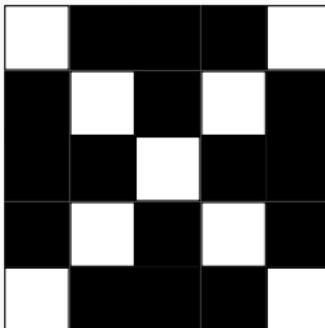


Vekter/parametre

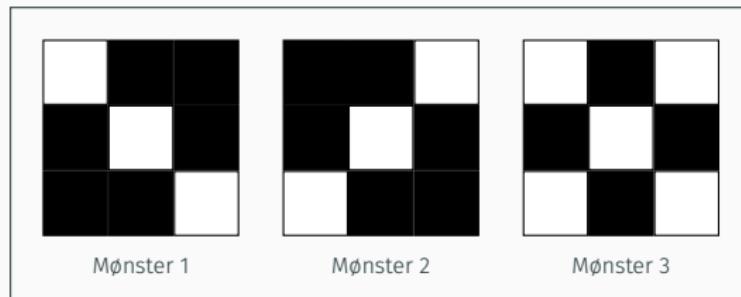
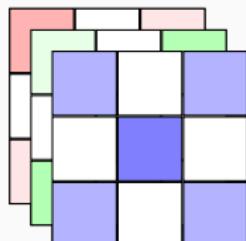


Konvolusjonelle nevrale nettverk: Konvolusjon

Bilde



Feature-kart



Vekter/parametre



Konvolusjonelle nevrale nettverk: Pooling

Feature-kart

0	1	2	3
4	5	6	7
8	9	10	1
12	13	14	15



Konvolusjonelle nevrale nettverk: Pooling

Feature-kart

0	1	2	3
4	5	6	7
8	9	10	1
12	13	14	15

5



Konvolusjonelle nevrale nettverk: Pooling

Feature-kart

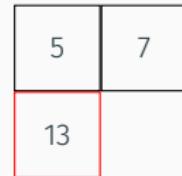
0	1	2	3
4	5	6	7
8	9	10	1
12	13	14	15



Konvolusjonelle nevrale nettverk: Pooling

Feature-kart

0	1	2	3
4	5	6	7
8	9	10	1
12	13	14	15



Konvolusjonelle nevrale nettverk: Pooling

Feature-kart

0	1	2	3
4	5	6	7
8	9	10	11
12	13	14	15

5	7
13	15



Konvolusjonelle nevrale nettverk: Pooling

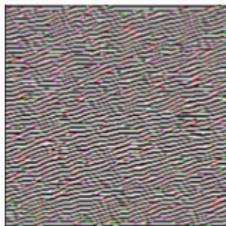
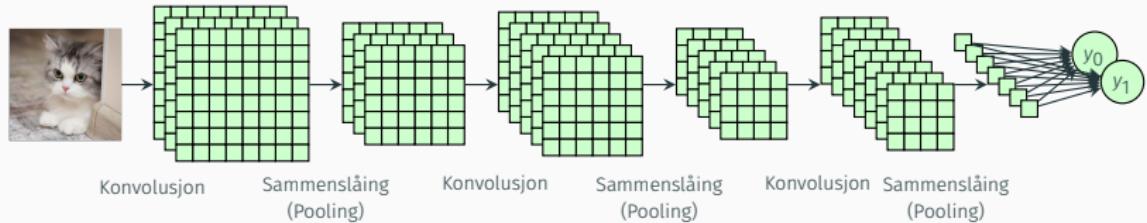
Feature-kart

0	1	2	3
4	5	6	7
8	9	10	1
12	13	14	15

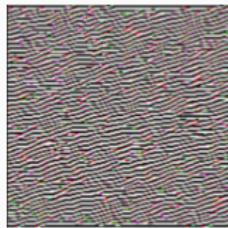
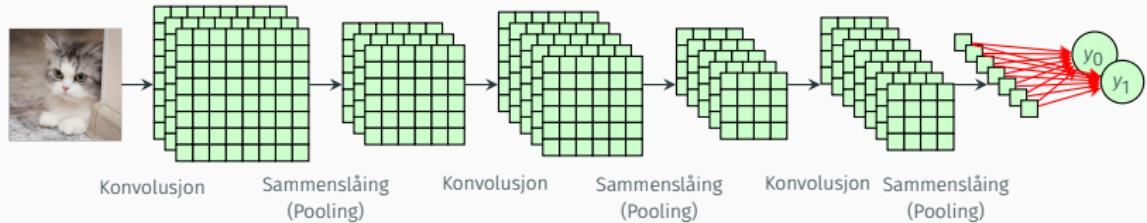
5	7
13	15



Konvolusjonelle nevrale nettverk: Arkitektur



Konvolusjonelle nevrale nettverk: Arkitektur

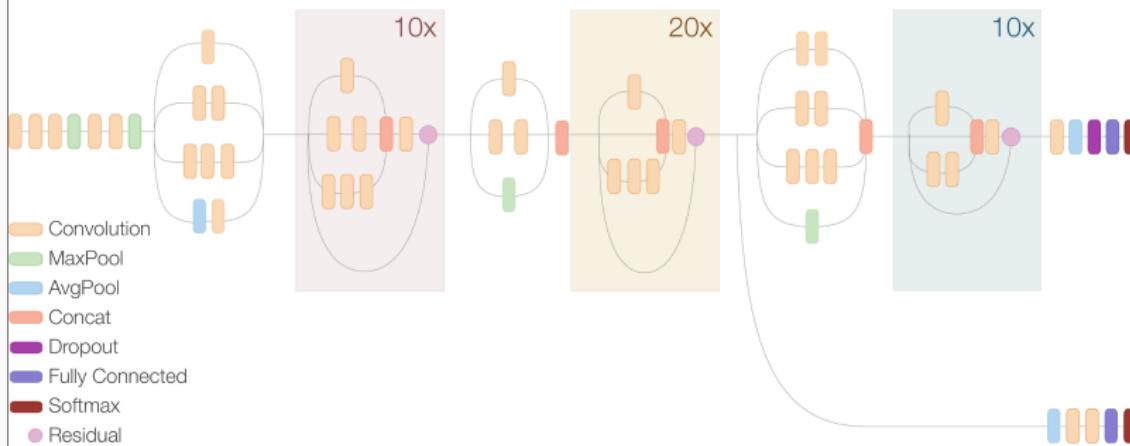


Konvolusjonelle nevrale nettverk: Arkitektur

Inception Resnet V2 Network



Compressed View



Hvordan fungerer et konvolusjonelt nevralt nettverk (for klassifikasjon)?

- Bilder representeres ved 3-dimensjonale arrayer i en datamaskin, som kan mates direkte som input til nettverket
- For å oppnå klassifikasjon gir vi nettverket flere output-nevroner, en for hver klasse vi er interessert i, også tolker vi outputen slik at nevronet med høyest verdi sammensvarer med den predikerte klassifikatoren
- Inne i nettverket utføres alternerende konvolusjoner og pooling som i praksis lar oss gjenkjenne større og mer abstrakte mønstre jo dypere i modellen vi kommer
- Til slutt kobles mønstrene opp mot klassene ved et "vanlig" nevralt nett (alle-til-alle koblinger)
- Parametrene som trenes ligger i konvolusjonene og sier hvilke mønstre som må læres, og i det siste laget

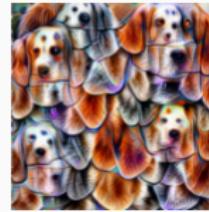
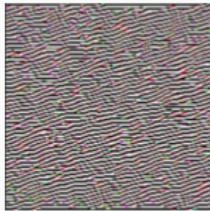
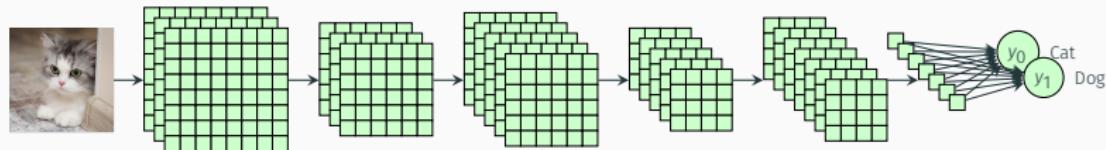


Praktiske tips

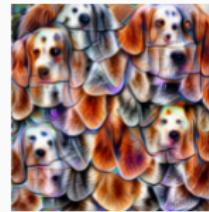
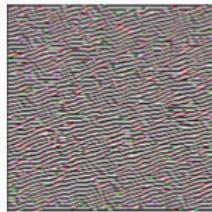
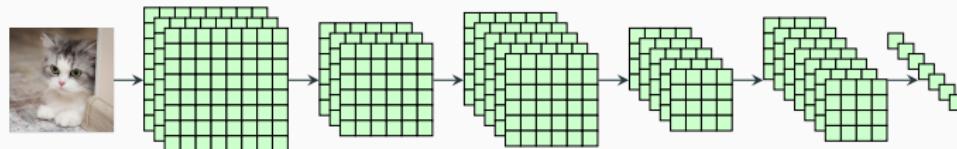


UNIVERSITETET
I OSLO

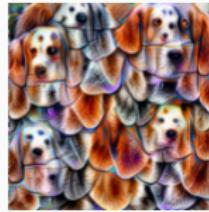
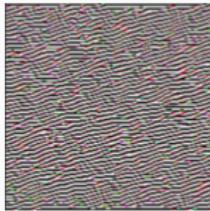
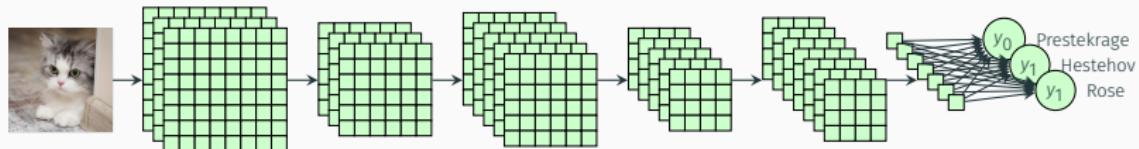
Praktiske tips: Transfer learning



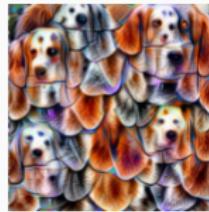
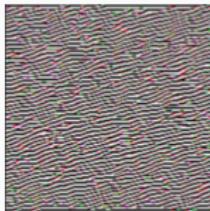
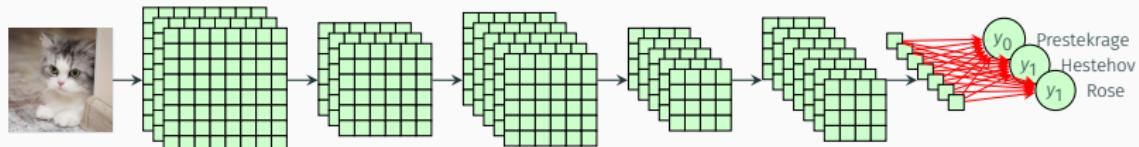
Praktiske tips: Transfer learning



Praktiske tips: Transfer learning



Praktiske tips: Transfer learning

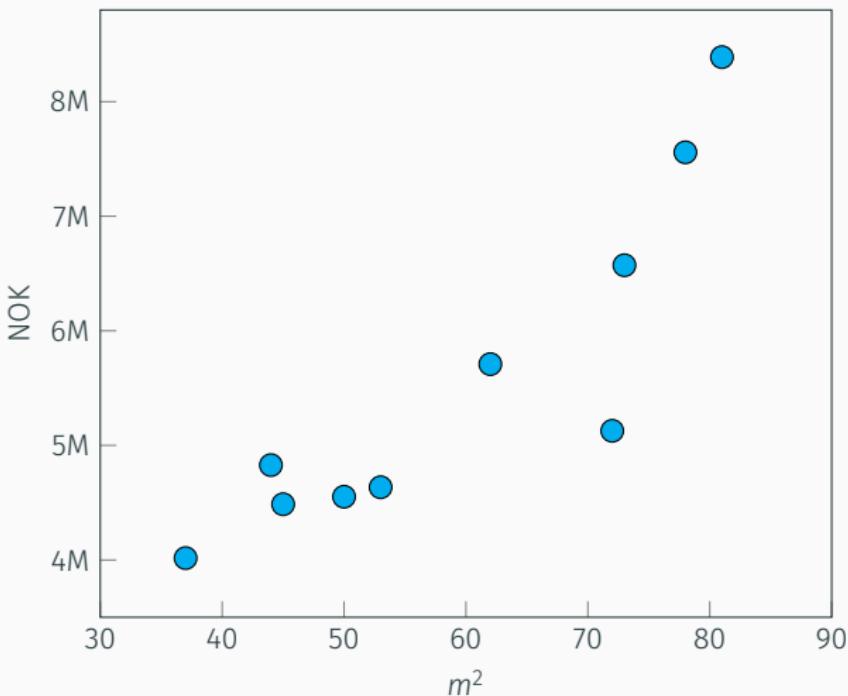


Hva er transfer learning?

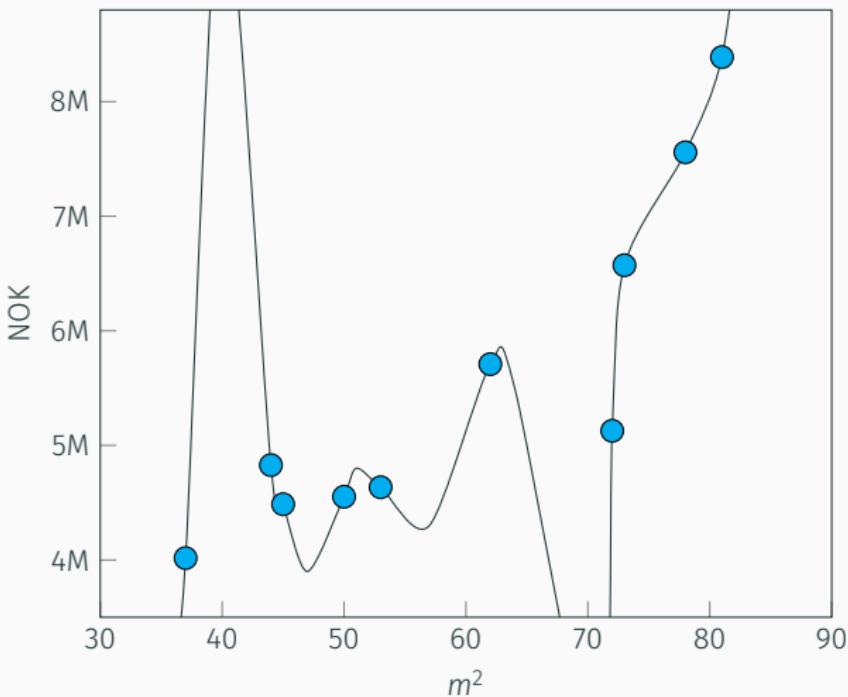
- En teknikk for å utnytte at det er overlapp mellom visuelle problemer, og at et nettverk som har blitt trent til å løse en oppgave antageligvis har lært noe som er nyttig også for en annen
- I praksis: Vi beholder de første lagene av en ferdigtrent modell (som gjenkjenner generiske visuelle mønstre, kanter, farger, geometri etc) og trener nye lag på toppen av disse



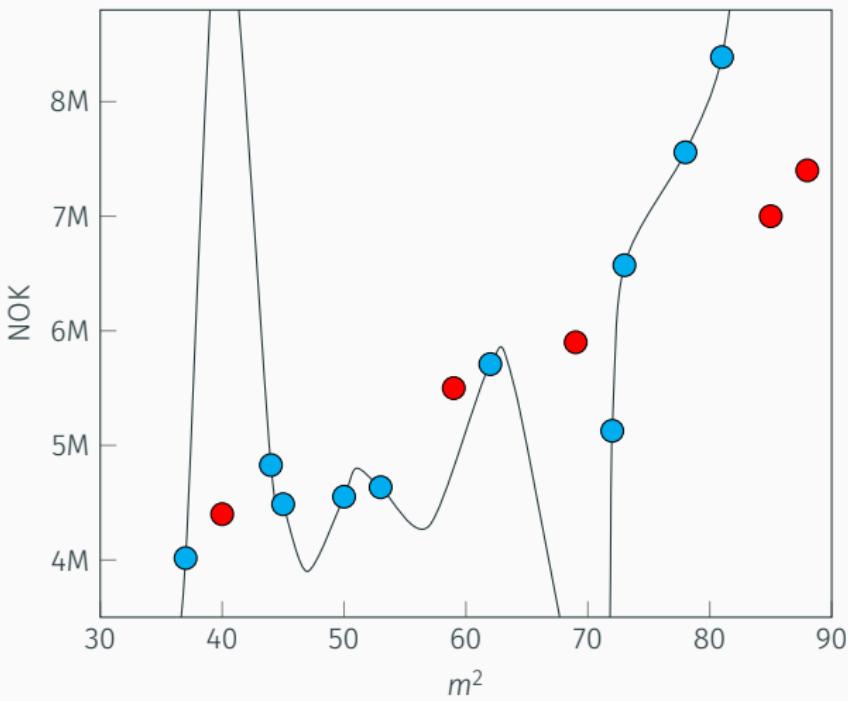
Praktiske tips: Overtilpasning



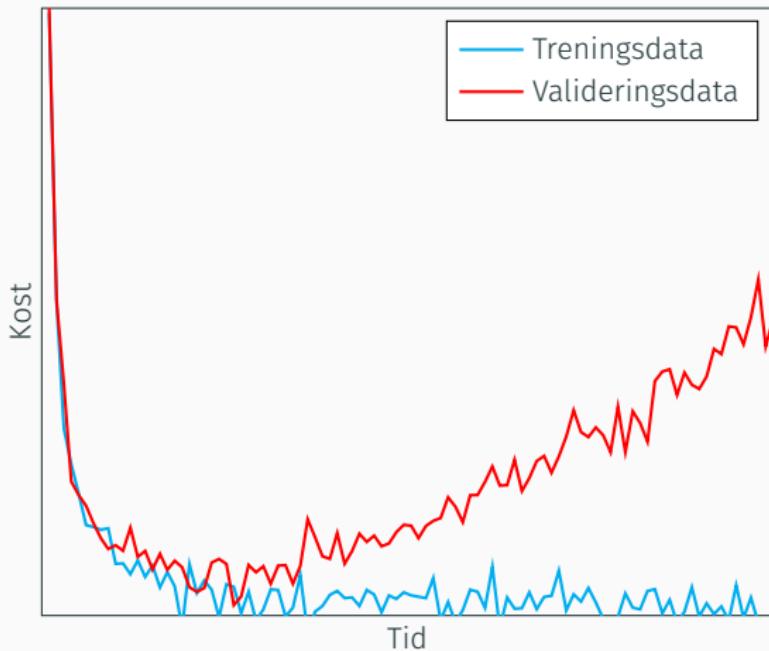
Praktiske tips: Overtilpasning



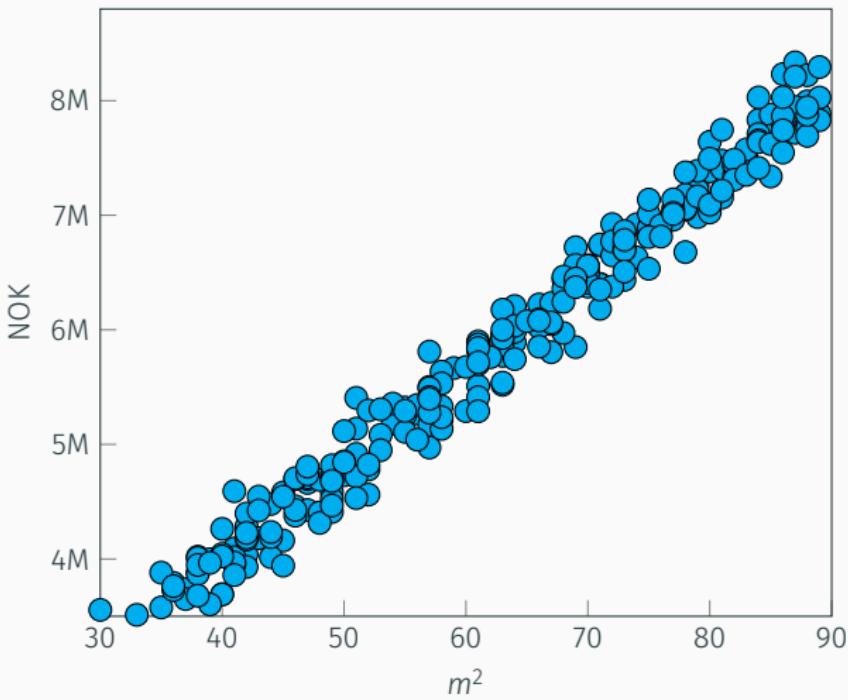
Praktiske tips: Overtilpasning



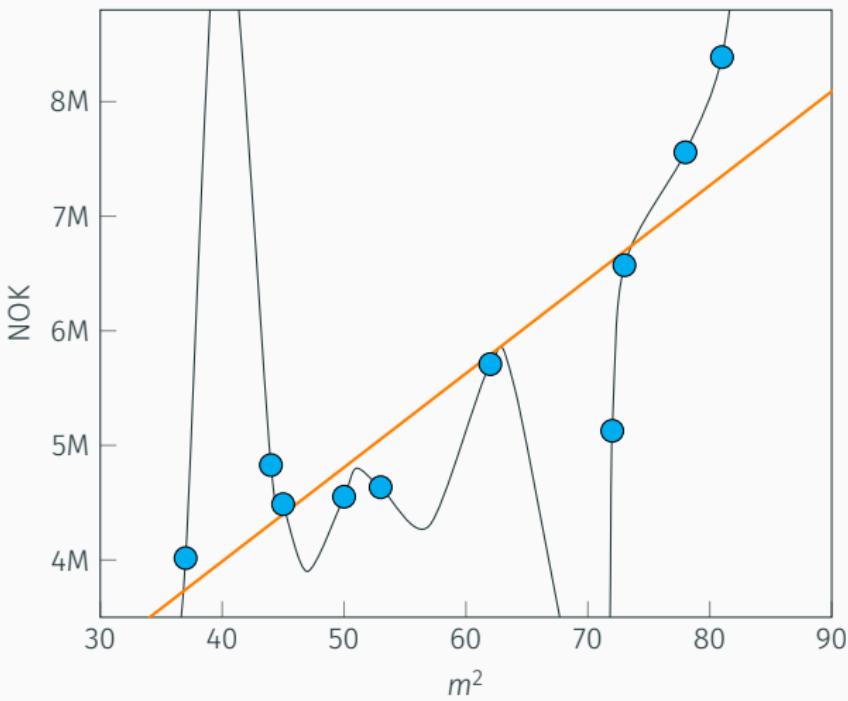
Praktiske tips: Overtilpasning

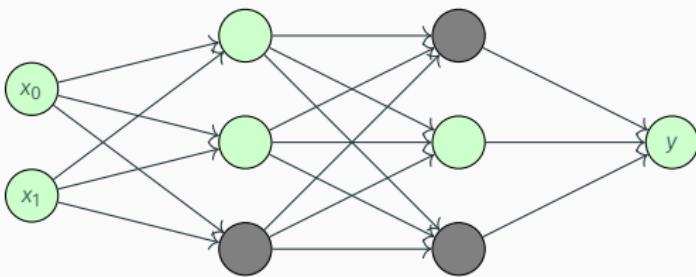


Praktiske tips: Overtilpasning

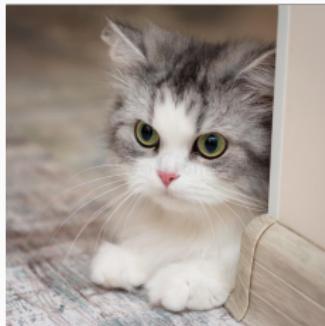


Praktiske tips: Overtilpasning

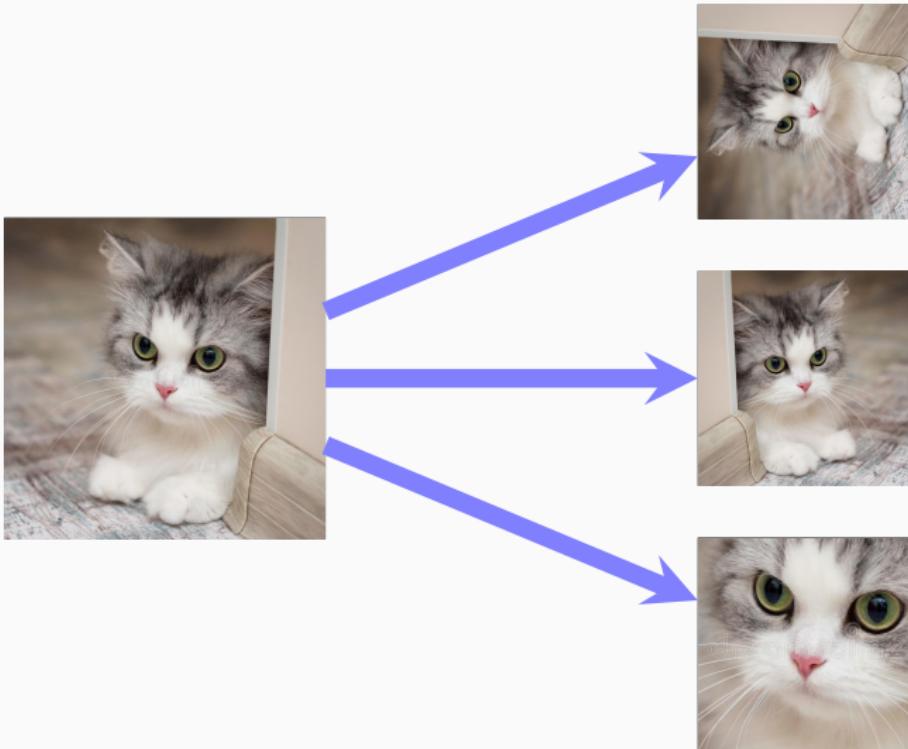




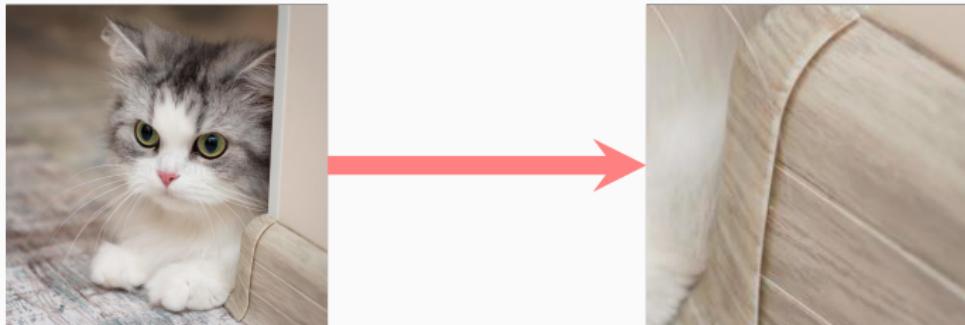
Praktiske tips: Overtilpasning



Praktiske tips: Overtilpasning



Praktiske tips: Overtilpasning



Hva er overtilpasning, og hvordan unngår vi det?

- Modellen har lært å gjenkjenne mønstre i treningsdataen som **ikke holder** i det generelle tilfellet (e.g. i ny data)
- Vi unngår det ved: Store datamengder, testing (!!!), regularisering, augmentering, ...

