



UiT Norges arktiske universitet

CNN's - bildeklassifisering

SOK-3023 (ML for økonomer), 5 ECTS

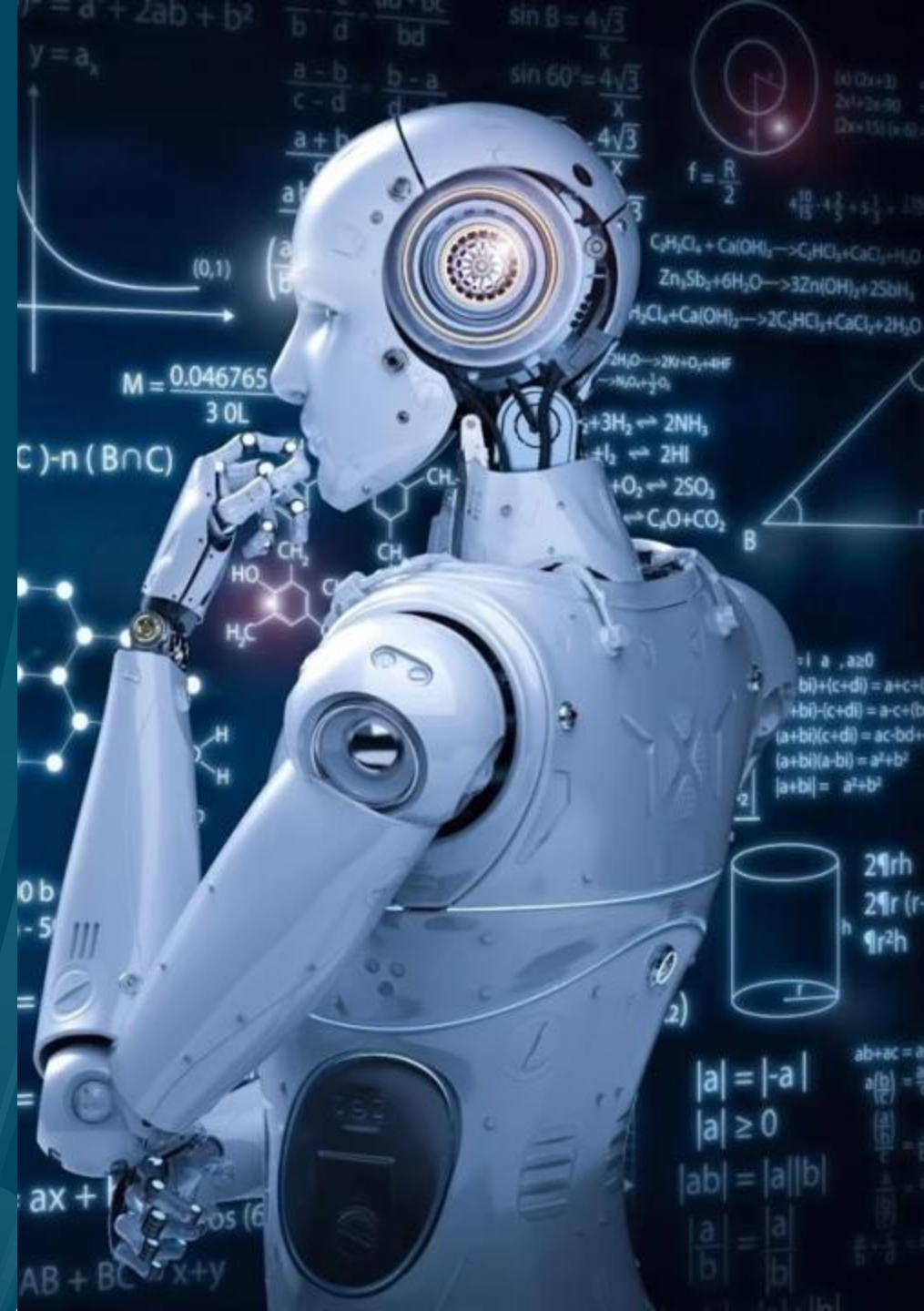
Markus J. Aase

markus.j.aase@uit.no, kontor 02.411

Universitetslektor i matematikk og statistikk

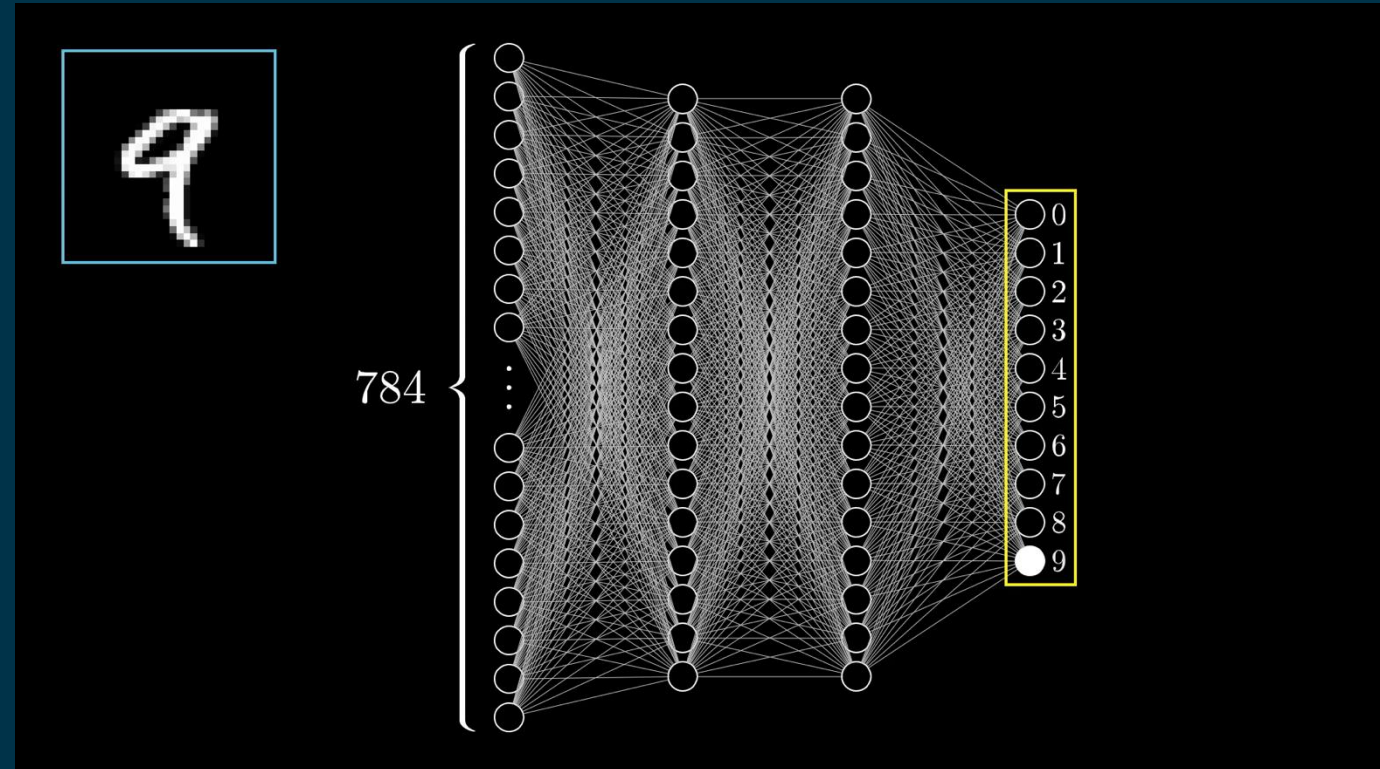
Handelshøgskolen, UiT

Master i samfunnsøkonomi med datavitenskap



Til nå

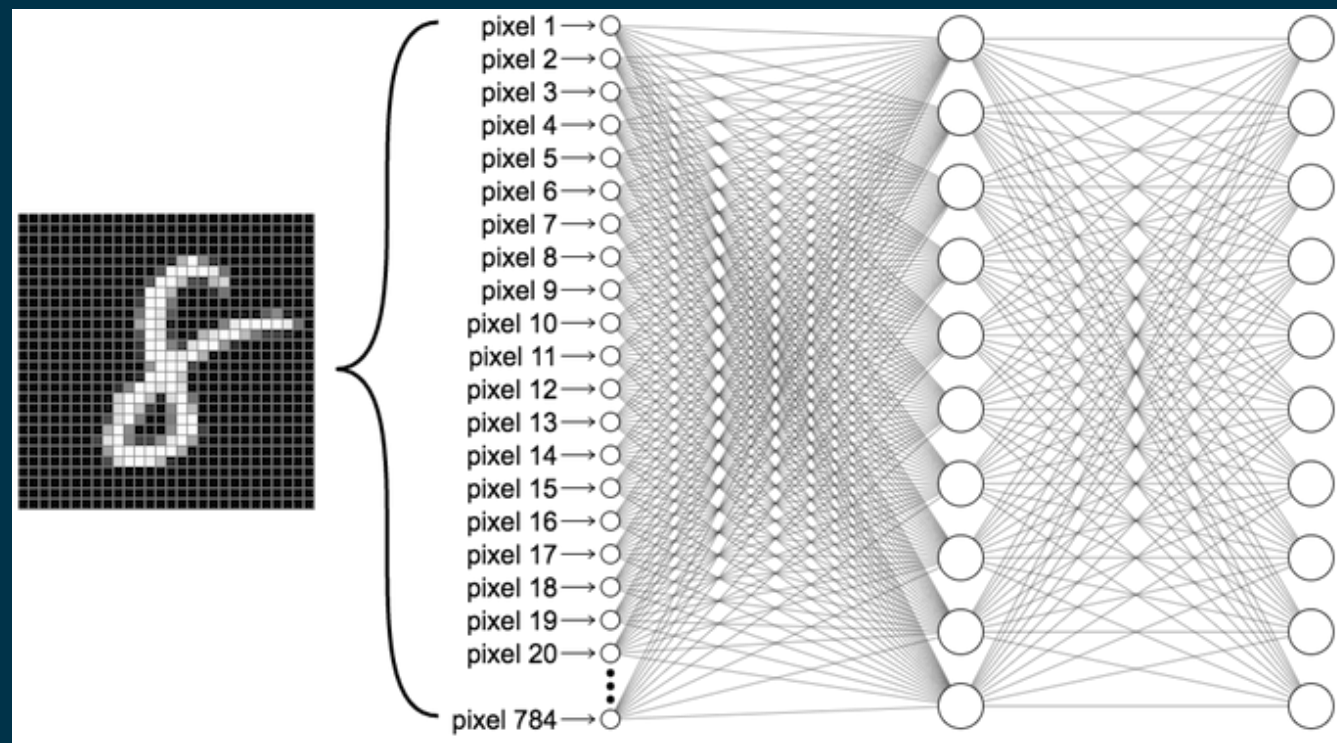
- Maskinlæring
 - Lineær algebra, statistikk og kalkulus
 - Prediksjon vs inferens
 - Regresjon vs klassifikasjon
 - Bias variance trade-off
 - Nødvendighet av god data
 - Reducible og irreducible error
 - Tensorer
 - Veiledet læring
 - Overfitting/underfitting
 - Trenings-, validerings- og testsett
 - Evalueringsmetrikker – accuracy, sensitivitet, precision osv.
 - Arkitektur i nevrale nettverk
 - Aktiveringsfunksjoner, loss/kost-funksjon
 - «Læring» = minimering av loss-funksjon
 - Epoch, batch, batch_size
 - Gradient descent



Kilde: 3b1b

Bildeklassifisering

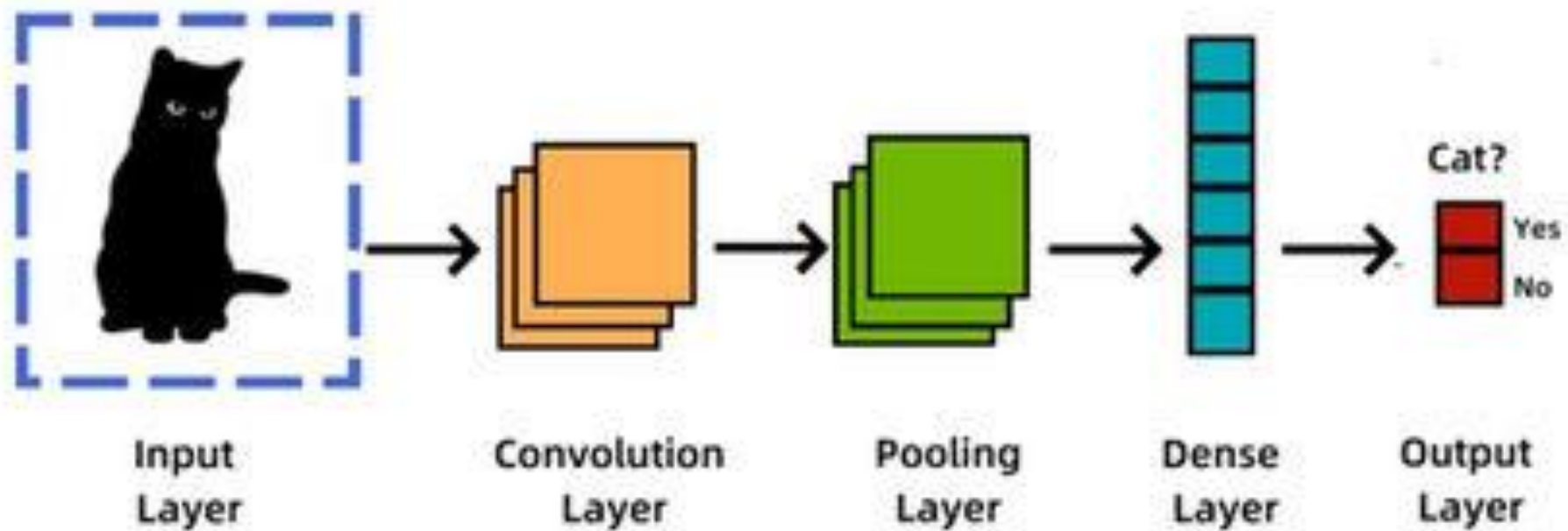
- I starten av kurset så vi på bilder, og brukte MLP til bildeklassifisering av håndskrevne tall.
- Da gjorde vi om 28x28-bilder til en lang vektor med 784 komponenter.



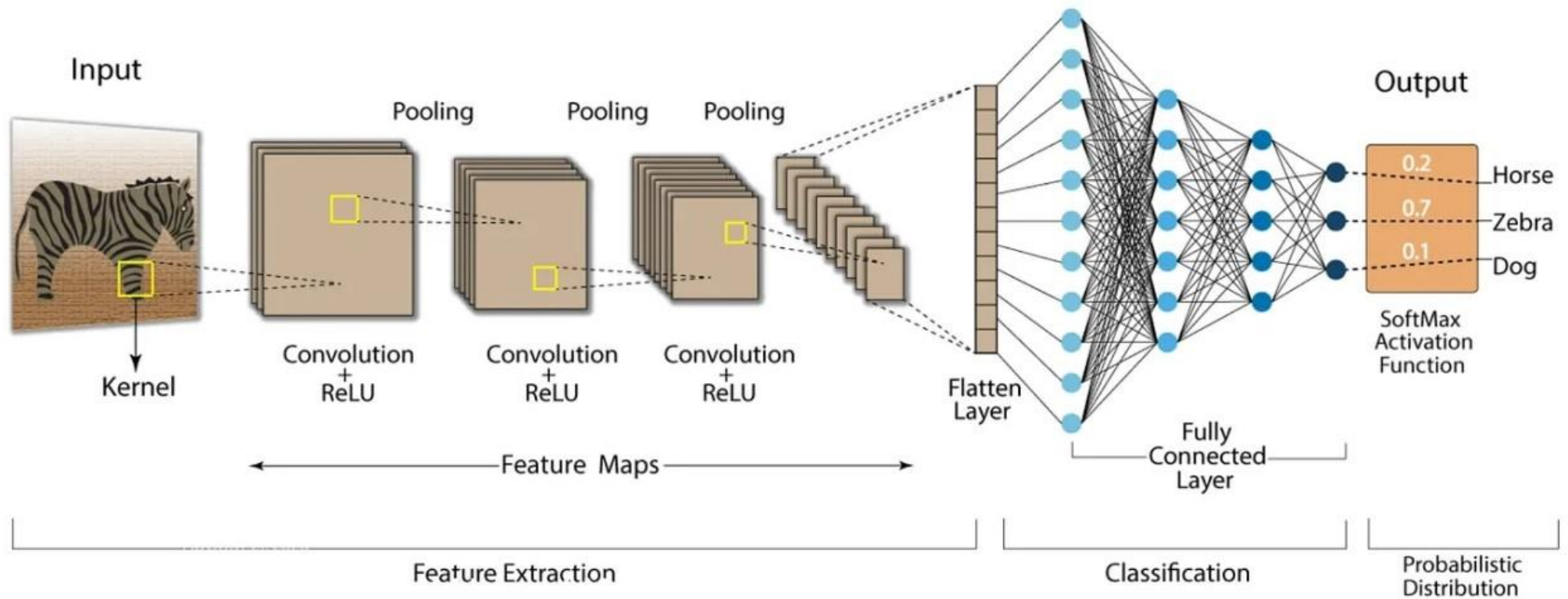
Kilde: zitaoshen.rbind.io/

Convolutional Neural Networks

- En type dyplæring som passer særs godt til bildegjenkjenning og finne underliggende mønstre (eng: *pattern recognition*).
 - Består av flere lag (filter)
 - Konvolusjonslag
 - Pooling-lag
 - Fully-connected lag
 - Aktiveringsfunksjoner
- } Dette kan vi allerede 😊



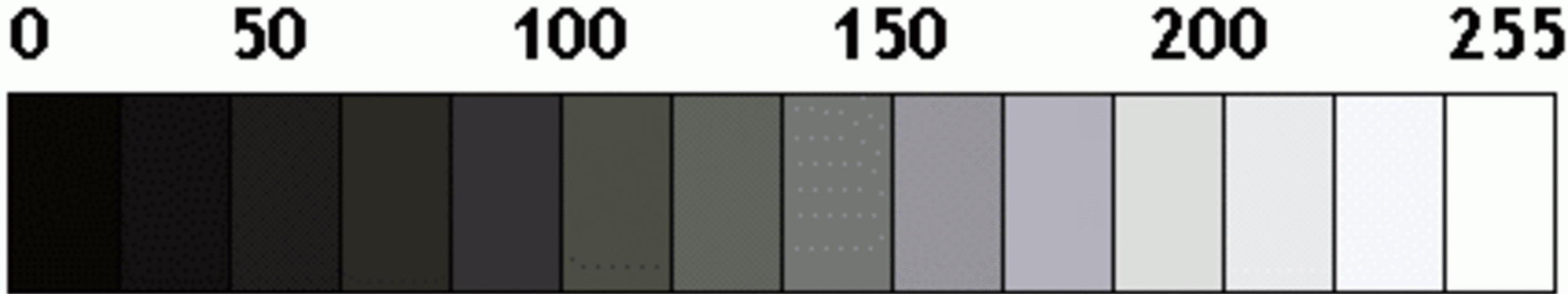
Convolution Neural Network (CNN)



Kilde: ingoampt.com

Viktig konsept - pikselverdier

Pixel values

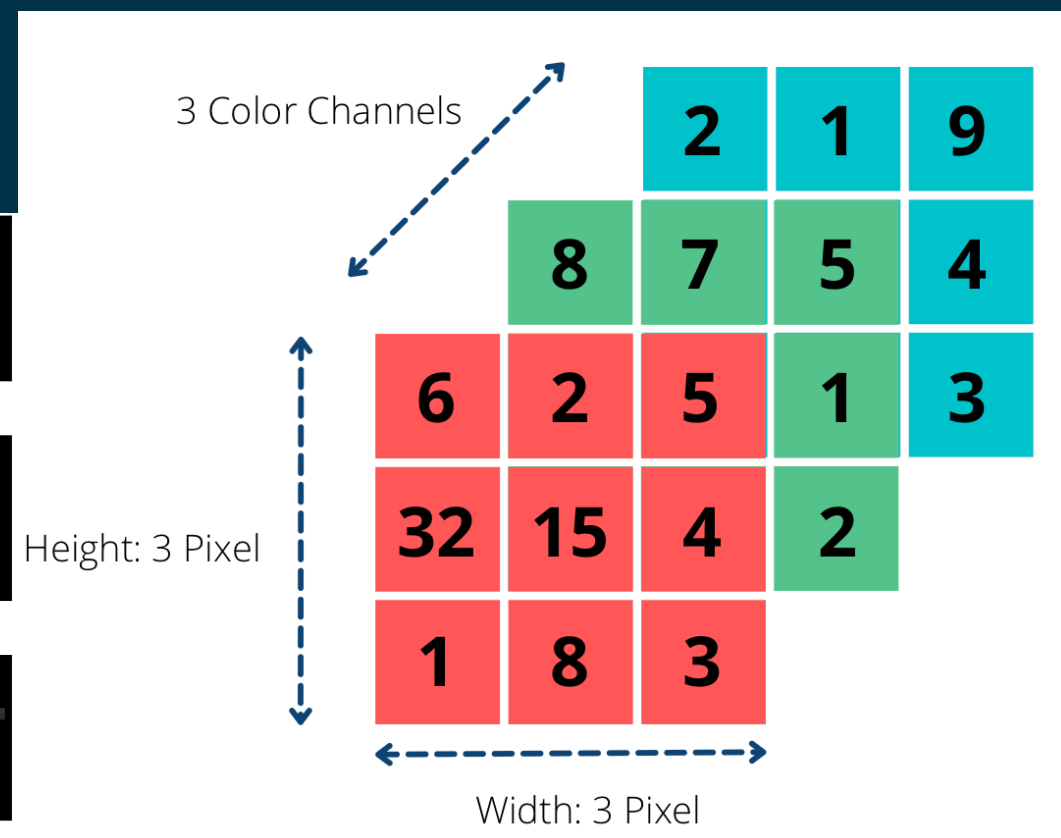


CNN's- input

- Input
 - Husker du rank-3 tensor?
 - Fargebilde = 3 kanaler
 - Verdier 0 - 255
- Rank-2 tensor
 - Sort-hvitt bilde




Kilde: mnist



Kilde: medium.com

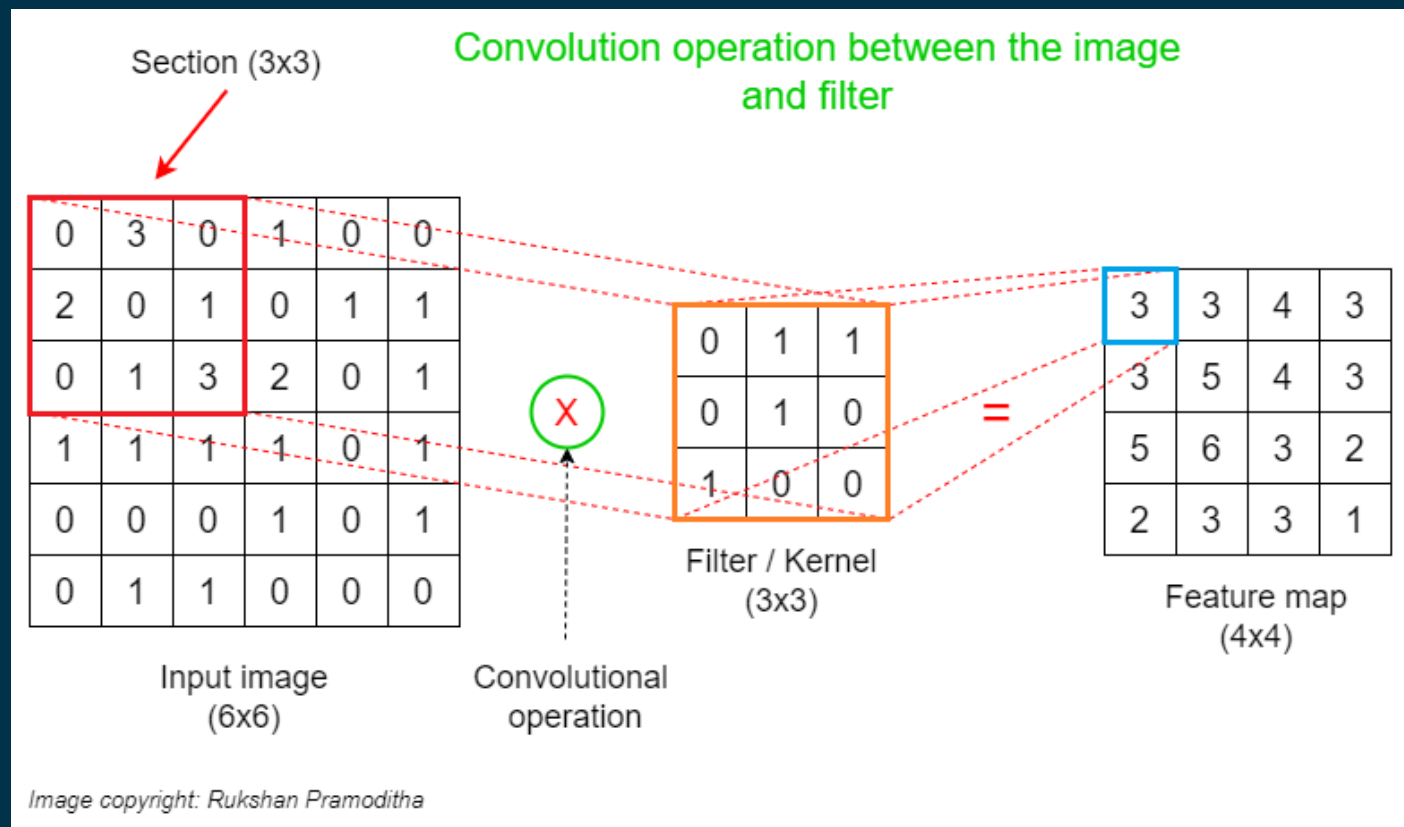
Viktig konsept - pikselverdier

	Red	Green	Blue	Hexadecimal code
	0	0	0	#000000
	255	255	255	#FFFFFF
	255	0	0	#FF0000
	0	255	0	#00FF00
	0	0	255	#0000FF
	255	128	0	#FF8000
	255	255	0	#FFFF00
	128	128	128	#808080

CNN's - Konvolusjonslag

- Konvolusjon
 - En matematisk operasjon på to funksjoner (f og g), som gir en tredje funksjon som output ($f * g$).
- Implementeres som
 - Kernel
 - Filter
 - Feature detector

} Samme ting
- Dette lager en ny matrise, vi kaller et *feature map* av input-laget.
 - Her brukes det aktiveringsfunksjoner, for å kapre ikke-lineære sammenhenger.



Input image

Shape(1, 5, 5)

9	4	1	5	3
1	1	1	1	3
1	2	1	4	5
1	5	3	5	2
0	4	3	1	2

Filter
3x3 matrix

0	2	1
4	1	0
1	0	1

$$9 \cdot 0 + 4 \cdot 2 + 1 \cdot 1 \dots 1 \cdot 1 = 16$$

Dot product

16		

Output
3x3 matrix

Stride 1: Filter moves only one from left to right

1	2	3	5	3
3	2	3	1	3
5	4	3	4	5
1	5	3	5	2
0	1	3	5	2

0	2	1
4	1	0
1	0	1

1	2	3	5	3
3	2	3	1	3
5	4	3	4	5
1	5	3	5	2
0	1	3	5	2

0	2	1
4	1	0
1	0	1

16	30	

16	30	34

original



convolved



CNN's - Pooling

- Pooling-lag
 - Down-sampler feature map ved å bare beholde det nødvendige, og forkaste resten.
- Max-pooling
 - Down-sampler basert på maksimalverdi i en viss matrisestørrelse – dette gir et nytt, mindre feature map.
- Hvorfor?
 - Reduserer størrelse, fanger mønstre

Max Pooling

29	15	28	184
0	100	70	38
12	12	7	2
12	12	45	6

2 x 2
pool size

100	184
12	45

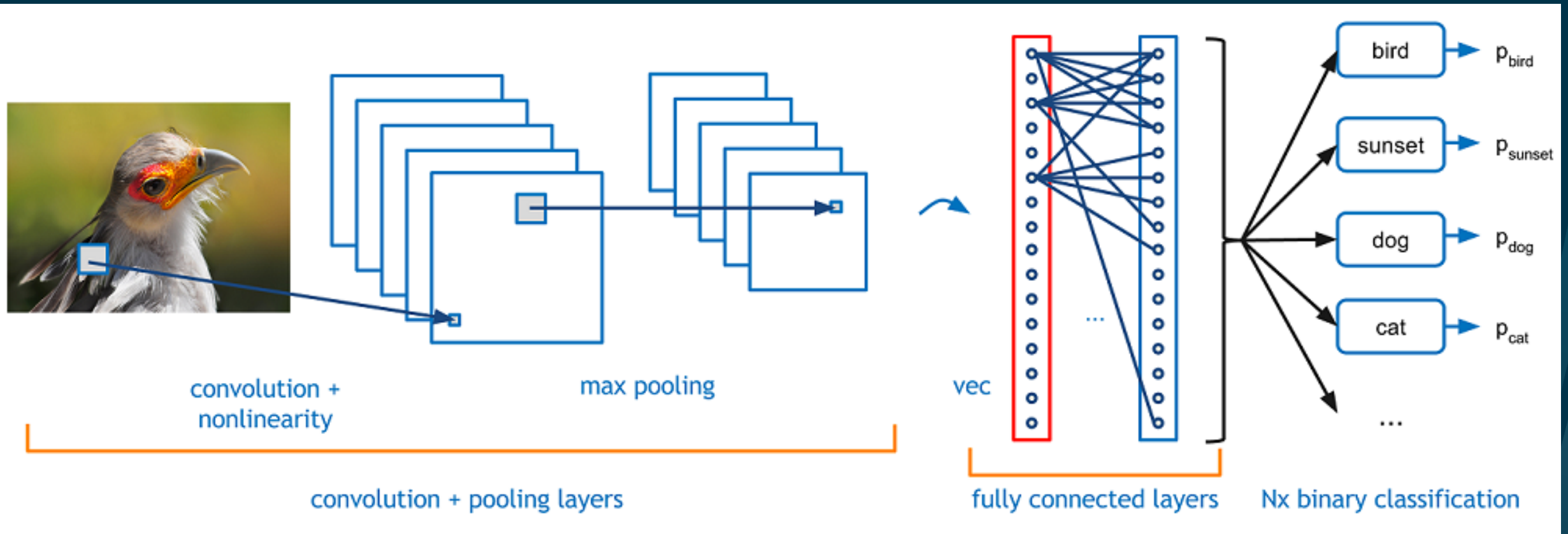
Average Pooling

31	15	28	184
0	100	70	38
12	12	7	2
12	12	45	6

2 x 2
pool size

36	80
12	15

En gang til



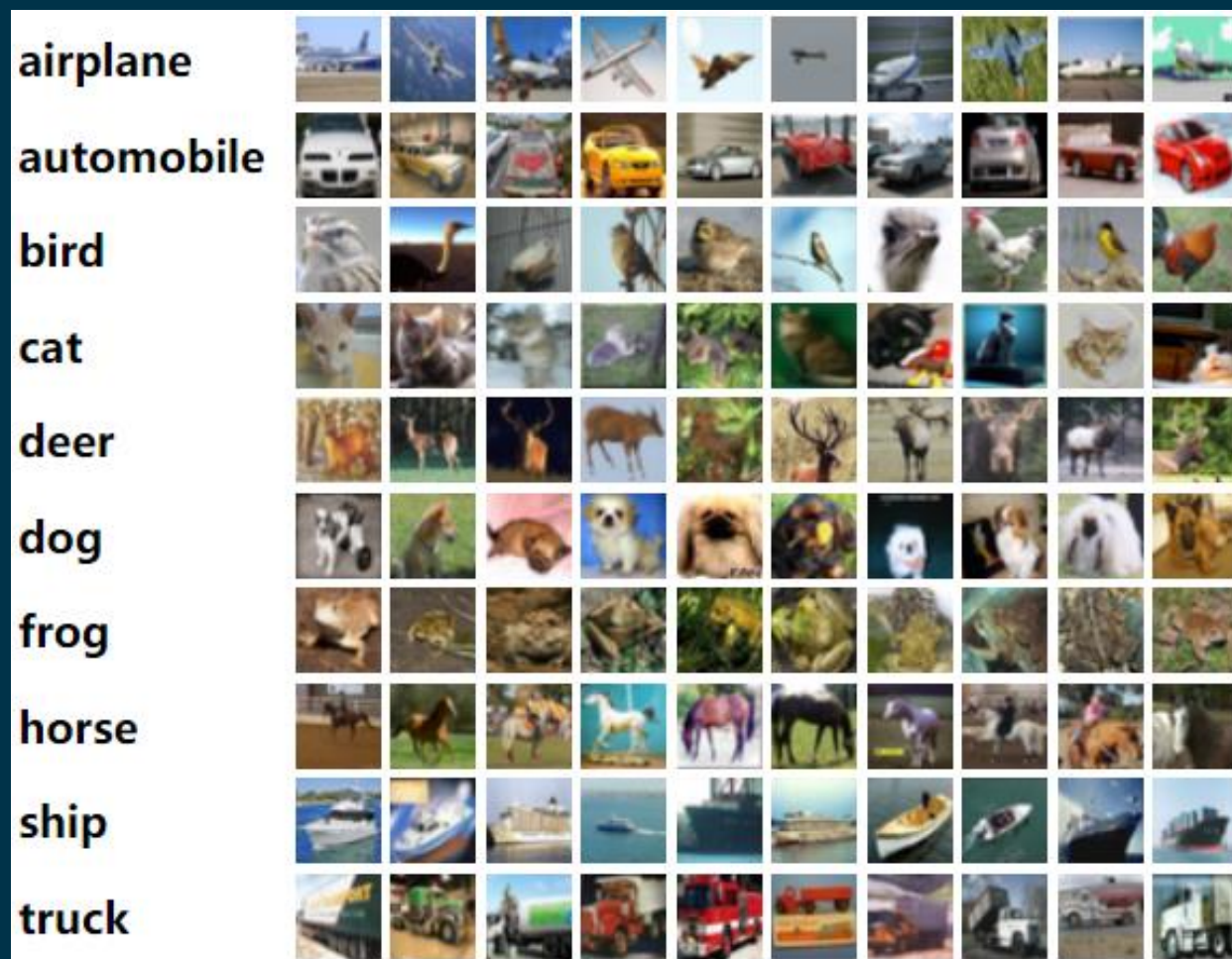
Kilde: opendatascience.com

Prøv selv!

- https://adamharley.com/nn_vis/cnn/2d.html

Vi prøver sammen!

- CIFAR-10 datasett



Sjekk ut!

- <https://www.youtube.com/watch?v=QzY57FaENXg>