

Algebraic NN Representation

TCTI-VKAAI-17: Applied Artificial Intelligence

Huib Aldewereld

Leerdoelen



- Na deze les kan de student:
 - Weloverwogen keuzes maken met betrekking tot netwerktopologie, initialisatie, gewichten en leeralgoritme in het bouwen van een neurale netwerk voor een specifiek doel.
 - Een neurale netwerk implementeren door middel van parallel computing op GPU-cores (bijv. d.m.v. OpenCL).

Inhoudsopgave



- Motivatie

- Lineaire algebra
 - Vectoren
 - Vector operaties
 - Matrices
- Algebraic representation
 - Meer lagen
 - Cost
- Parallellisatie

3

Motivatie



- Neural network representatie
 - Object-oriented (vorige week)
 - Elk neuron is een object
 - Netwerk is een object gebouwd uit neuronen
 - Etc...
- Qua idee handig, maar heeft zijn nadelen
 - Veel overhead
 - Moeilijk te paralleliseren

4



Motivatie, vervolg

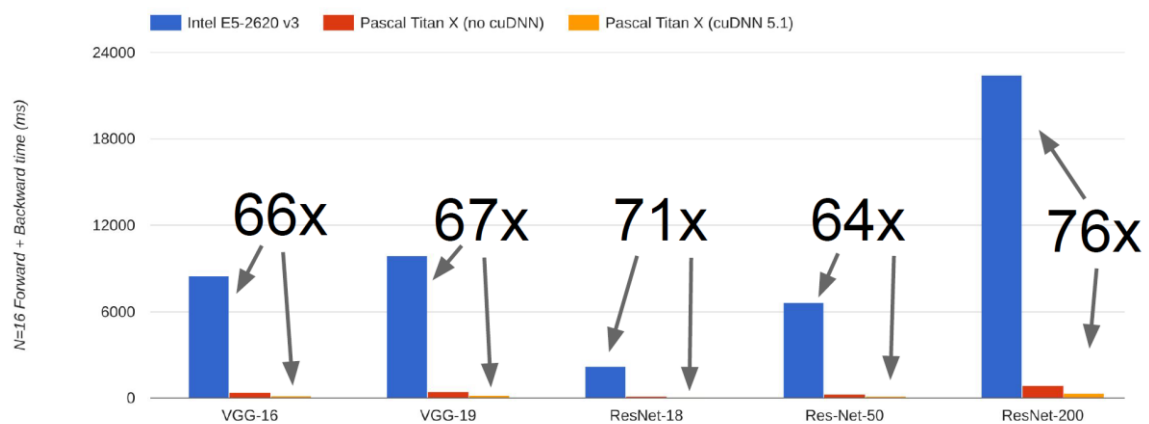
- Alternatief in wiskundige benadering
- Voordelen
 - Compact
 - Makkelijk(er) te paralleliseren, dmv gebruik GPU

	# Cores	Clock speed	Memory	Price
CPU (Intel Core i7-7700k)	4 (8 with hyperthreading)	4.4 GHz	Shared with system	~€ 340
CPU (Intel Core i7-6950k)	10 (20 with hyperthreading)	3.5 GHz	Shared with system	~€ 1723
GPU (NVIDIA Titan Xp)	3840	1.6 GHz	12 GB GDDR5X	~€ 1200
GPU (NVIDIA GTX 1070)	1920	1.68 GHz	8 GB GDDR5	~€ 399

5



CPU vs GPU in practice



Data from <https://github.com/jcjohnson/cnn-benchmarks>

6



Algebraic representatie

- Idee:
 - Invoer, hidden layers en output zijn (n -dimensionale) vectoren
 - Gewichten geven verband tussen waarden in vorige laag en de volgende: lineaire maps (matrices)
 - Matrix-vermenigvuldiging om deze maps toe te passen
 - Gradient descent om matrices te updaten

Maar eerst....
Opfrisser lineaire algebra!

7



Inhoudsopgave

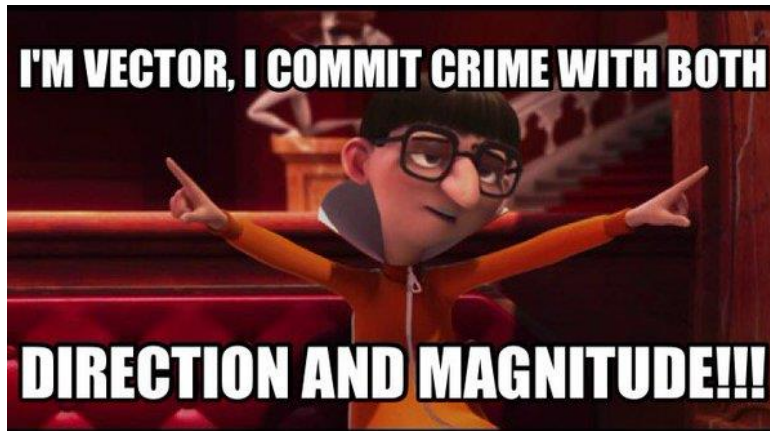
- Motivatie
- Lineaire algebra
 - Vectoren
 - Vector operaties
 - Matrices
- Algebraic representation
 - Meer lagen
 - Cost
- Parallellisatie

8

Lineaire algebra: vectoren

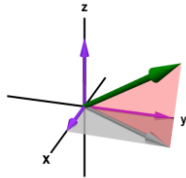
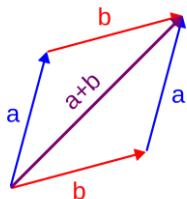


- Wat waren dat ook al weer



9

Vectoren – drie interpretaties



$$\begin{bmatrix} 1 & 2 & 4 \end{bmatrix} \begin{bmatrix} 1 \\ 2 \\ 4 \end{bmatrix}$$

Pijlen in een ruimte

(natuurkunde)

Lijsten van getallen

(informatica)

$$\vec{u}, \vec{v}, \vec{w} \in \mathbb{R}^3, c \in \mathbb{R}$$

$$\vec{u} + \vec{v} = \vec{v} + \vec{u}$$

$$\vec{u} + (\vec{v} + \vec{w}) = (\vec{u} + \vec{v}) + \vec{w}$$

$$c\vec{v} \in \mathbb{R}^3$$

$$c(\vec{u} + \vec{v}) = c\vec{u} + c\vec{v}$$

Objecten in een vectorruimte

(wiskunde)

10



Vectoren – drie interpretaties

- Elke interpretatie heeft een toepassing
 - Natuurkundige interpretatie is geschikt voor visualisatie (maar enkel in 2D en 3D)
 - Informatica interpretatie voor uitvoeren van operaties
 - Wiskundige interpretatie vertelt wat wel en niet mag
- Wat kunnen we met vectoren?
 - Optellen / aftrekken
 - Scalair vermenigvuldigen
 - Inwendig / dot product
 - ... rest (bijvoorbeeld kruisproduct), buiten scope cursus

11



Vectoren optellen en aftrekken

$$\vec{u}, \vec{v}, \vec{w} \in \mathbb{R}^3$$

$$\vec{u} = \begin{bmatrix} 1 \\ 2 \\ 3 \end{bmatrix} \quad \vec{v} = \begin{bmatrix} 4 \\ 5 \\ 6 \end{bmatrix} \quad \vec{w} = \begin{bmatrix} 7 \\ 8 \\ 9 \end{bmatrix}$$

$$\vec{u} + \vec{v} = \begin{bmatrix} 1 \\ 2 \\ 3 \end{bmatrix} + \begin{bmatrix} 4 \\ 5 \\ 6 \end{bmatrix} = \begin{bmatrix} 5 \\ 7 \\ 9 \end{bmatrix} = \begin{bmatrix} 4 \\ 5 \\ 6 \end{bmatrix} + \begin{bmatrix} 1 \\ 2 \\ 3 \end{bmatrix} = \vec{v} + \vec{u}$$

$$\vec{u} + (\vec{v} + \vec{w}) = (\vec{u} + \vec{v}) + \vec{w}$$

- (Reken zelf na op papier)

12



Vector scalair vermenigvuldigen

$$\vec{u}, \vec{v} \in \mathbb{R}^3; c, d \in \mathbb{R}$$

$$\vec{u} = \begin{bmatrix} 1 \\ 2 \\ 3 \end{bmatrix}, \vec{v} = \begin{bmatrix} 4 \\ 5 \\ 6 \end{bmatrix}, c = 2, d = 3$$

$$c\vec{u} = \vec{u} + \vec{u} = \begin{bmatrix} 1 \\ 2 \\ 3 \end{bmatrix} + \begin{bmatrix} 1 \\ 2 \\ 3 \end{bmatrix} = \begin{bmatrix} 2 \times 1 \\ 2 \times 2 \\ 2 \times 3 \end{bmatrix} = \begin{bmatrix} 2 \\ 4 \\ 6 \end{bmatrix}$$

$$c(\vec{u} + \vec{v}) = 2 \left(\begin{bmatrix} 1 \\ 2 \\ 3 \end{bmatrix} + \begin{bmatrix} 4 \\ 5 \\ 6 \end{bmatrix} \right) = 2 \begin{bmatrix} 1 \\ 2 \\ 3 \end{bmatrix} + 2 \begin{bmatrix} 4 \\ 5 \\ 6 \end{bmatrix} = c\vec{u} + c\vec{v}$$

$$c(d\vec{u}) = (cd)\vec{u}$$

- (Reken zelf na op papier)

13



Dot product

$$\vec{u}, \vec{v} \in \mathbb{R}^3$$

$$\vec{u} = \begin{bmatrix} 1 \\ 2 \\ 3 \end{bmatrix}, \vec{v} = \begin{bmatrix} 4 \\ 5 \\ 6 \end{bmatrix}$$

$$\vec{u} \cdot \vec{v} = \vec{u}_0 \vec{v}_0 + \vec{u}_1 \vec{v}_1 + \vec{u}_2 \vec{v}_2$$

$$\begin{bmatrix} 1 \\ 2 \\ 3 \end{bmatrix} \cdot \begin{bmatrix} 4 \\ 5 \\ 6 \end{bmatrix} = 1 \times 4 + 2 \times 5 + 3 \times 6 \\ = 32$$

14



Inwendig product

- Generalisatie dot-product
- Notatie
 - Dot-product (alleen vectoren) $\vec{u} \cdot \vec{v}$
 - Algemeen inwendig product $\langle \vec{u}, \vec{v} \rangle$
 - Bra-Ket notatie (Dirac) $\langle \vec{u} | \vec{v} \rangle$
- Inwendig product werkt ook voor matrices

Wat was een matrix ook al weer?

15



Matrices

- Net als vectoren: collecties van getallen
- Maar: met zowel rijen als kolumnen $M = \begin{bmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \end{bmatrix}, \vec{v} = \begin{bmatrix} 1 \\ 2 \\ 3 \end{bmatrix}, \vec{v}^T = [1 \quad 2 \quad 3]$
- Rij en kolom-vectoren zijn matrices
 - Een rijvector is een matrix met één rij $M \in \mathbb{R}^{2 \times 3}, \vec{v} \in \mathbb{R}^3, \vec{v}^T \in \mathbb{R}^3$
 - Een kolomvector is een matrix met één kolom
- Vectoren: pijlen, matrices: transformaties
- $\mathbb{R}^{m \times n}$ is zelf ook een vectorruimte
 - Optellen als vectoren: puntsgewijs
 - Scalair product als vectoren: puntsgewijs

16



Matrix-producten

- We kunnen twee matrices vermenigvuldigen, notatie voor matrices M en N is MN
- Matrixproduct is niet symmetrisch: $MN \neq NM$
- Matrixproduct MN is alleen mogelijk als $M \in \mathbb{R}^{a \times b}$ en $N \in \mathbb{R}^{b \times c}$, het product behoort dan tot $\mathbb{R}^{a \times c}$

$$\begin{bmatrix} a & b \\ c & d \end{bmatrix} \begin{bmatrix} e & f \\ g & h \end{bmatrix} = \begin{bmatrix} ae + bg & af + bh \\ ce + dg & cf + dh \end{bmatrix}$$

- $AB = C \Rightarrow C_{x,y} = \langle A_{x,*} | B_{*,y} \rangle$
- Inwendig product bij vectoren komt overeen met matrixvermenigvuldiging van één rij en één kolom
- Uitwendig product precies andersom: vermenigvuldiging één kolom met één rij

17



Samenvatting notatie

- y is een getal (scalar)
- \vec{b} of \mathbf{b} is een vector
- \vec{b}_1 is een getal in \vec{b} (een scalar)
- $\vec{b}^{(2)}$ is vector (niet te verwarren met $\vec{b}^{(1)}$ of $\vec{b}^{(3)}$)
- A is een matrix
- $A_{x,y}$ of A_{xy} is een getal in A (rij x , kolom y)
- $A_{x,*}$ is een vector opgebouwd uit de rij x van A
- θ en Θ worden gebruikt voor de gewichten ("theta")
- $\langle \vec{a} | \vec{b} \rangle$ is het inwendig product van \vec{a} en \vec{b}
- $c\vec{v}$ en cA zijn scalaire producten
- $\vec{v}A$, $A\vec{v}$ en AB zijn matrixvermenigvuldigingen

18

Inhoudsopgave



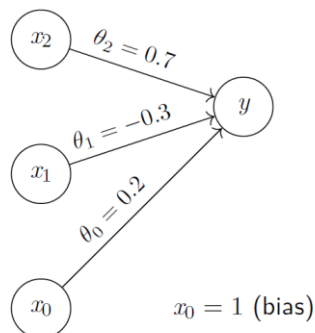
- Motivatie
- Lineaire algebra
 - Vectoren
 - Vector operaties
 - Matrices
- Algebraic representation
 - Meer lagen
 - Cost
- Parallelisatie

19

Neurale netwerken als matrixproducten



- Voldoende achtergrond om een neuraal netwerk te vertalen:



- Waar doet dit aan denken?

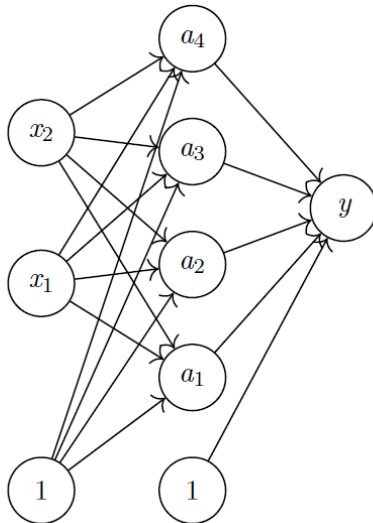
$$y = \sigma(\theta_0 x_0 + \theta_1 x_1 + \theta_2 x_2)$$

Inwendig product!

$$y = \sigma\langle \theta | \vec{x} \rangle$$

20

Meer lagen

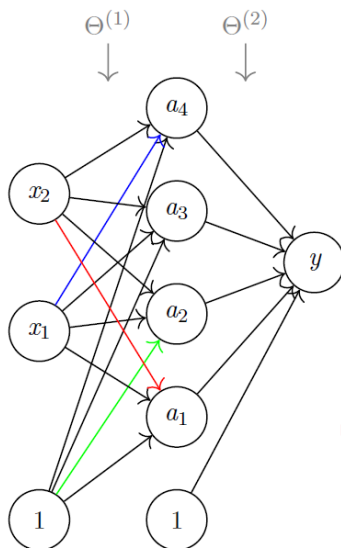


$$a_1 = \langle \vec{x} | \theta^{(1)} \rangle, a_2 = \langle \vec{x} | \theta^{(2)} \rangle, \dots$$

$$\vec{a} = \sigma(\Theta^{(1)} \vec{x}), \quad \Theta^{(1)} = \begin{bmatrix} - & \vec{\theta}^1 & - \\ - & \vec{\theta}^2 & - \\ - & \vec{\theta}^3 & - \\ - & \vec{\theta}^4 & - \end{bmatrix}$$

21

Meer lagen



$$\Theta^{(1)} = \begin{bmatrix} 0.1 & -0.2 & -0.3 \\ 0.4 & -0.5 & 0.6 \\ -0.7 & 0.8 & 0.9 \\ -0.1 & 0.2 & 0.3 \end{bmatrix}$$

$\Theta_{ij}^{(1)}$ is het gewicht van x_j naar a_i

Rijen tellen vanaf 1, kolommen van 0;
kolom 0 is voor de bias

$$\Theta^{(2)} = [0.1 \quad 0.1 \quad 0.2 \quad 0.3 \quad 0.5]$$

$$\vec{a} = \sigma(\Theta^{(1)} \vec{x}), \quad \vec{a}' = [1 \quad \vec{a}], \quad y = \sigma(\Theta^{(2)} \vec{a}')$$

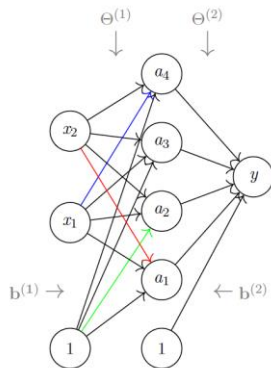
- Om verwarring met rijen van $\Theta^{(1)}$ te voorkomen is de laatste laag de 1×4 -matrix $\Theta^{(2)}$ en duiden we deze niet aan als de vector $\theta^{(2)}$

22



Bias

- In plaats van bias in gewichten te coderen ook mogelijk om hier een losse vector voor te gebruiken, die we bij het product optellen:



$$\Theta^{(1)} = \begin{bmatrix} -0.2 & -0.3 \\ -0.5 & 0.6 \\ 0.8 & 0.9 \\ 0.2 & 0.3 \end{bmatrix}, \quad \vec{b}^{(1)} = \begin{bmatrix} 0.1 \\ 0.4 \\ -0.7 \\ -0.1 \end{bmatrix}$$

$\Theta_{ij}^{(1)}$ is het gewicht van x_j naar a_i

Rijen en kolommen tellen vanaf 1

$$\Theta^{(2)} = [0.1 \quad 0.2 \quad 0.3 \quad 0.5], \quad \vec{b}^{(2)} = [0.1]$$

$$\vec{a} = \sigma(\Theta^{(1)}\vec{x} + \vec{b}^{(1)}), \quad y = \sigma(\Theta^{(2)}\vec{a} + \vec{b}^{(2)})$$

23



Need to go deeper!

- 2 Hidden Layers
- 1024 Input Neurons
- 42 in Hidden Layer 1
- 28 in Hidden Layer 2
- 12 Output Neurons

- Inputvector
- Activatievector HL1
- Activatievector HL2
- Outputvector

$$\vec{x} \in \mathbb{R}^{1024}$$

$$\vec{a}^{(1)} \in \mathbb{R}^{42}$$

$$\vec{a}^{(2)} \in \mathbb{R}^{28}$$

$$\vec{y} \in \mathbb{R}^{12}$$

24



Need to go deeper!

$$\vec{x} \in \mathbb{R}^{1024}, \vec{a}^{(1)} \in \mathbb{R}^{42}, \vec{a}^{(2)} \in \mathbb{R}^{28}, \vec{y} \in \mathbb{R}^{12}$$

- Hieruit kunnen we de dimensies van de weights en biases opmaken:

$$\Theta^{(1)} \in \mathbb{R}^{42 \times 1024}, \Theta^{(2)} \in \mathbb{R}^{28 \times 42}, \Theta^{(3)} \in \mathbb{R}^{12 \times 28}$$

$$\vec{b}^{(1)} \in \mathbb{R}^{42}, \vec{b}^{(2)} \in \mathbb{R}^{28}, \vec{b}^{(3)} \in \mathbb{R}^{12}$$

- En komen we tot de volgende formules voor de verschillende lagen:

$$\vec{a}^{(1)} = \sigma(\Theta^{(1)}\vec{x} + \vec{b}^{(1)})$$

$$\vec{a}^{(2)} = \sigma(\Theta^{(2)}\vec{a}^{(1)} + \vec{b}^{(2)})$$

$$\vec{y} = \sigma(\Theta^{(3)}\vec{a}^{(2)} + \vec{b}^{(3)})$$

- Totaal: $\vec{y} = \sigma(\Theta^{(3)}\sigma(\Theta^{(2)}\sigma(\Theta^{(1)}\vec{x} + \vec{b}^{(1)}) + \vec{b}^{(2)}) + \vec{b}^{(3)})$

25



Cost

- De kostenfunctie is gebaseerd op het verschil tussen de voorspelde output y en de correcte output y' . Meestal kiezen we de sum of squared errors:

$$J = (\vec{y}_1 - \vec{y}'_1)^2 + (\vec{y}_2 - \vec{y}'_2)^2 + \dots + (\vec{y}_{12} - \vec{y}'_{12})^2$$

- Dit doet erg aan het inproduct van twee vectoren denken. Kunnen we dit vectoriseren?

$$J = \langle \vec{y} - \vec{y}' | \vec{y} - \vec{y}' \rangle \quad \text{of} \quad J = (\vec{y} - \vec{y}')^2$$

- De kostenfunctie is het inwendig product van de verschilvector met zichzelf

26



Inhoudsopgave

- Motivatie
- Lineaire algebra
 - Vectoren
 - Vector operaties
 - Matrices
- Algebraic representation
 - Meer lagen
 - Cost

■ Parallellisme

27



Parallel I

- Maar voor het trainen willen we niet de kostenfunctie voor ieder voorbeeld apart uitrekenen; kunnen we vectoriseren?
Met ons diepe voorbeeld van zojuist:

$$\vec{x} \in \mathbb{R}^{1024}, a^{(\vec{1})} \in \mathbb{R}^{42}, a^{(\vec{2})} \in \mathbb{R}^{28}, \vec{y} \in \mathbb{R}^{12}$$

$$\Theta^{(1)} \in \mathbb{R}^{42 \times 1024}, \Theta^{(2)} \in \mathbb{R}^{28 \times 42}, \Theta^{(3)} \in \mathbb{R}^{12 \times 28}$$

$$b^{(\vec{1})} \in \mathbb{R}^{42}, b^{(\vec{2})} \in \mathbb{R}^{28}, b^{(\vec{3})} \in \mathbb{R}^{12}$$

$$a^{(\vec{1})} = \sigma(\Theta^{(1)} \vec{x} + b^{(\vec{1})})$$

$$a^{(\vec{2})} = \sigma(\Theta^{(2)} a^{(\vec{1})} + b^{(\vec{2})})$$

$$\vec{y} = \sigma(\Theta^{(3)} a^{(\vec{2})} + b^{(\vec{3})})$$

28



Parallel II

- Maar voor het trainen willen we niet de kostenfunctie voor ieder voorbeeld apart uitrekenen; kunnen we vectoriseren?

Met ons diepe voorbeeld van zojuist:

$$X \in \mathbb{R}^{1024 \times 1}, A^{(1)} \in \mathbb{R}^{42 \times 1}, A^{(2)} \in \mathbb{R}^{28 \times 1}, Y \in \mathbb{R}^{12 \times 1}$$

$$\Theta^{(1)} \in \mathbb{R}^{42 \times 1024}, \Theta^{(2)} \in \mathbb{R}^{28 \times 42}, \Theta^{(3)} \in \mathbb{R}^{12 \times 28}$$

$$B^{(1)} \in \mathbb{R}^{42 \times 1}, B^{(2)} \in \mathbb{R}^{28 \times 1}, B^{(3)} \in \mathbb{R}^{12 \times 1}$$

$$A^{(1)} = \sigma(\Theta^{(1)} X + B^{(1)})$$

$$A^{(2)} = \sigma(\Theta^{(2)} A^{(1)} + B^{(2)})$$

$$Y = \sigma(\Theta^{(3)} A^{(2)} + B^{(3)})$$

Kunnen we vectoren als matrices interpreteren ...

29



Parallel III

- Maar voor het trainen willen we niet de kostenfunctie voor ieder voorbeeld apart uitrekenen; kunnen we vectoriseren?

Met ons diepe voorbeeld van zojuist:

$$X \in \mathbb{R}^{1024 \times n}, A^{(1)} \in \mathbb{R}^{42 \times n}, A^{(2)} \in \mathbb{R}^{28 \times n}, Y \in \mathbb{R}^{12 \times n}$$

$$\Theta^{(1)} \in \mathbb{R}^{42 \times 1024}, \Theta^{(2)} \in \mathbb{R}^{28 \times 42}, \Theta^{(3)} \in \mathbb{R}^{12 \times 28}$$

$$B^{(1)} \in \mathbb{R}^{42 \times n}, B^{(2)} \in \mathbb{R}^{28 \times n}, B^{(3)} \in \mathbb{R}^{12 \times n}$$

$$A^{(1)} = \sigma(\Theta^{(1)} X + B^{(1)})$$

$$A^{(2)} = \sigma(\Theta^{(2)} A^{(1)} + B^{(2)})$$

$$Y = \sigma(\Theta^{(3)} A^{(2)} + B^{(3)})$$

Kunnen we vectoren als matrices interpreteren ...

... en net zo makkelijk n voorbeelden tegelijk uitrekenen!

30

Huiswerk



- Bekijk filmpjes van 3Blue1Brown:
 - 1: **“But what *is* a Neural Network”**
 - (2: “Gradient descent, how neural networks learn” [optioneel])
 - 3: **“What is backpropagation really doing”**
 - (3a: “Backpropagation calculus” [optioneel])