

# Grundlagen der Datenwissenschaft und künstlicher Intelligenz für IoT-Entwickler

building IoT, 2024

Michal Harakal

# KANN DIE KI MEINEN ENTWICKLER JOB ÜBERNEHMEN?

building



2024

# AlphaGo

2016



# AlphaCode

## 2022



# Beispiele



# Beispiele

- Bilderkennung
- Spracherkennung und Sprachsynthese
- Übersetzungen
- Empfehlungsdienst

# Beispiele

## Ende Computing, IoT

- Visuelle und auditive Weckwörter
- Prädiktive Instandhaltung von Industriemaschinen mit Sensoren
- Gesten- und Aktivitätserkennung
- Datenschutz
- Begrenzte Bandbreite



**Andrej Karpathy** 

@karpathy

...

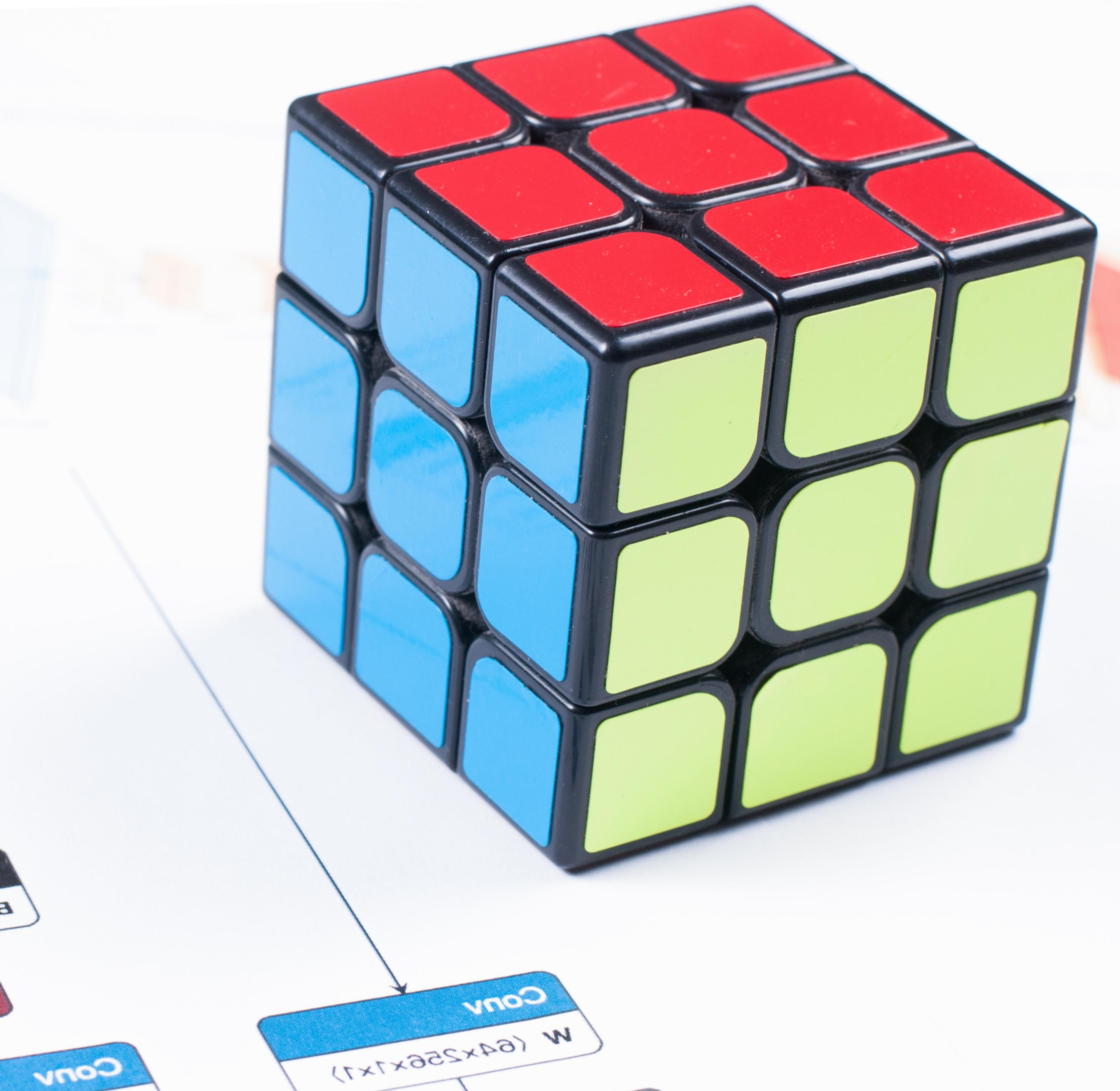
**Gradient descent can write code better than you. I'm sorry.**

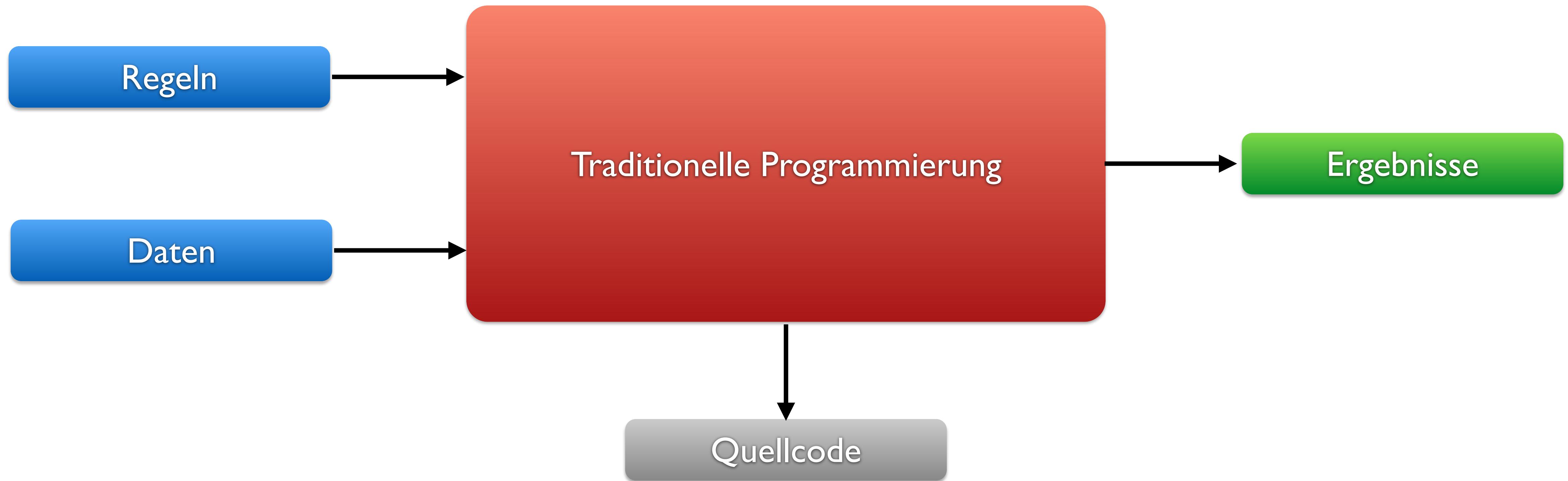
10:56 PM · Aug 4, 2017 · Twitter Web Client

---

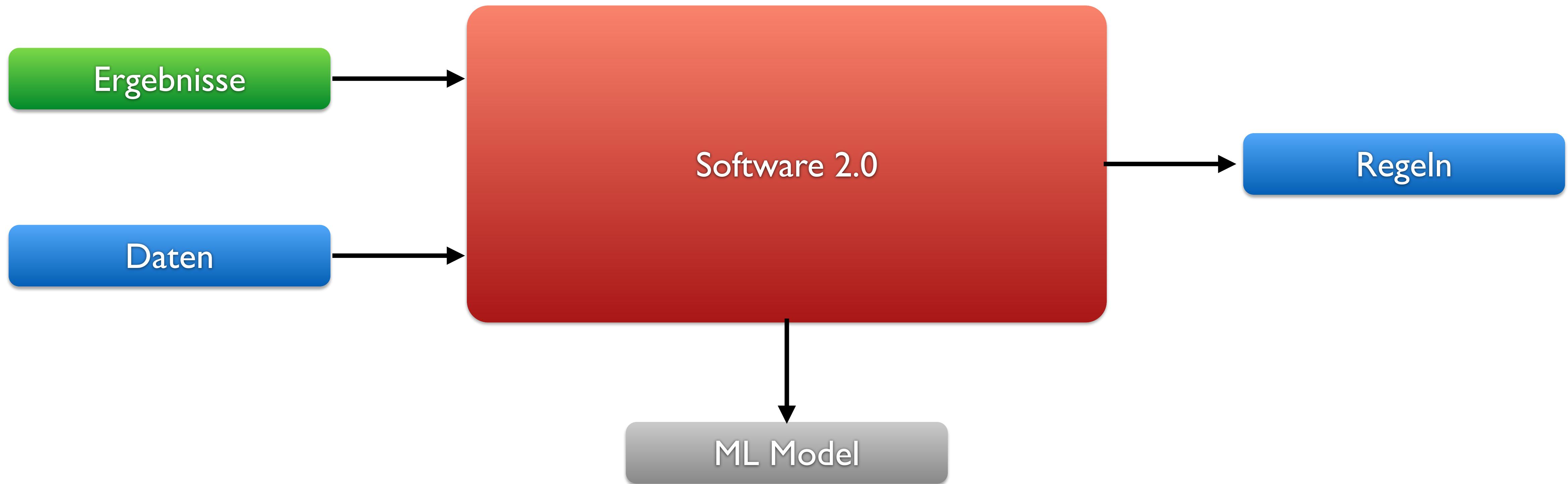
**493 Retweets**   **92 Quote Tweets**   **2,611 Likes**

# Software 2.0





# Traditionelle Programmierung



# Software 2.0

# Wir fordern KI heraus



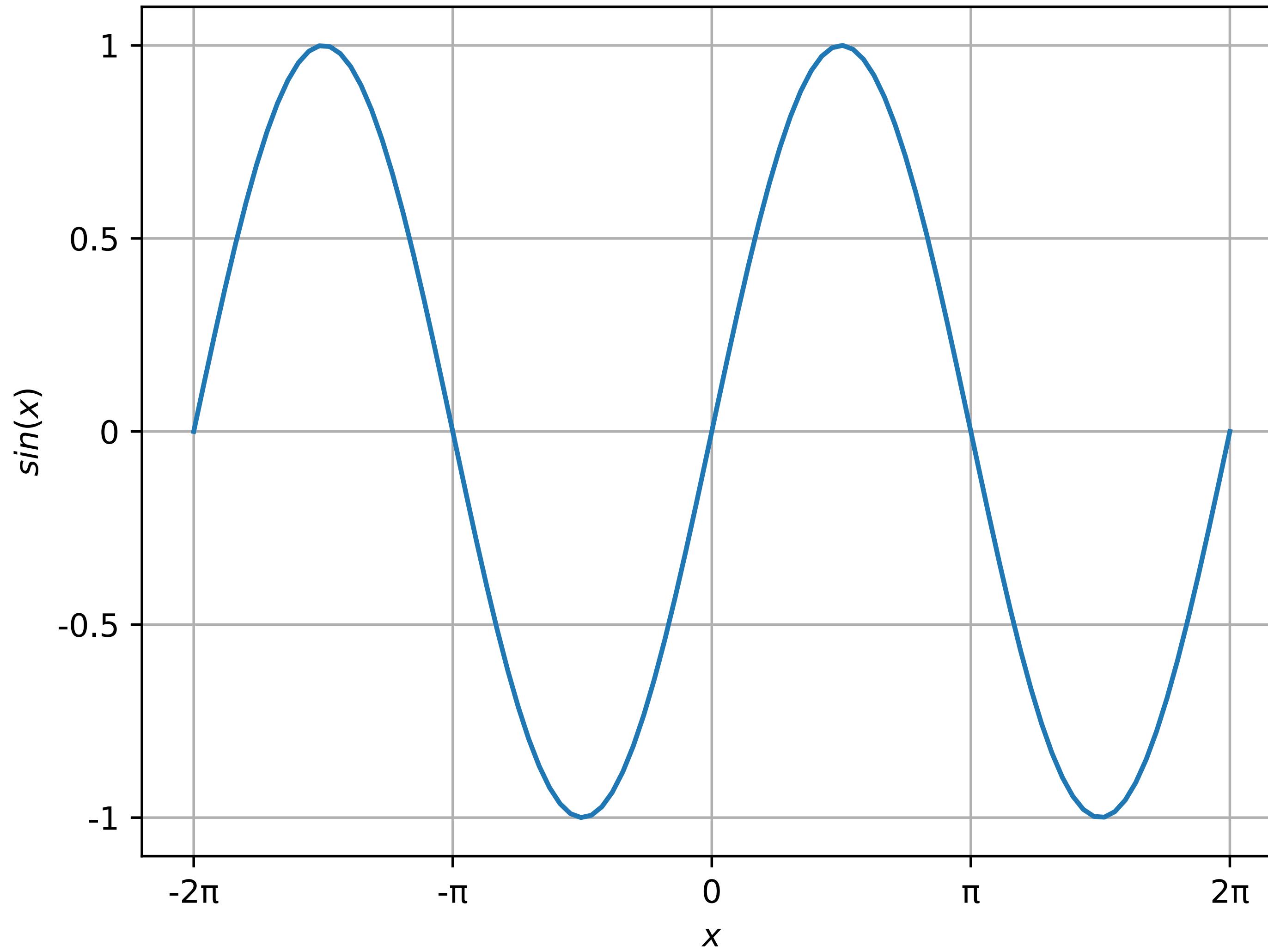
# **Apollo 11 Guidance Computer**

**1969**

- Navigationsrechner
- 4KiB RAM
- 64 KiB ROM
- Assembler

# Apollo 11 Guidance Computer

$$x = \sin(x)$$



# Apollo 11 Guidance Computer

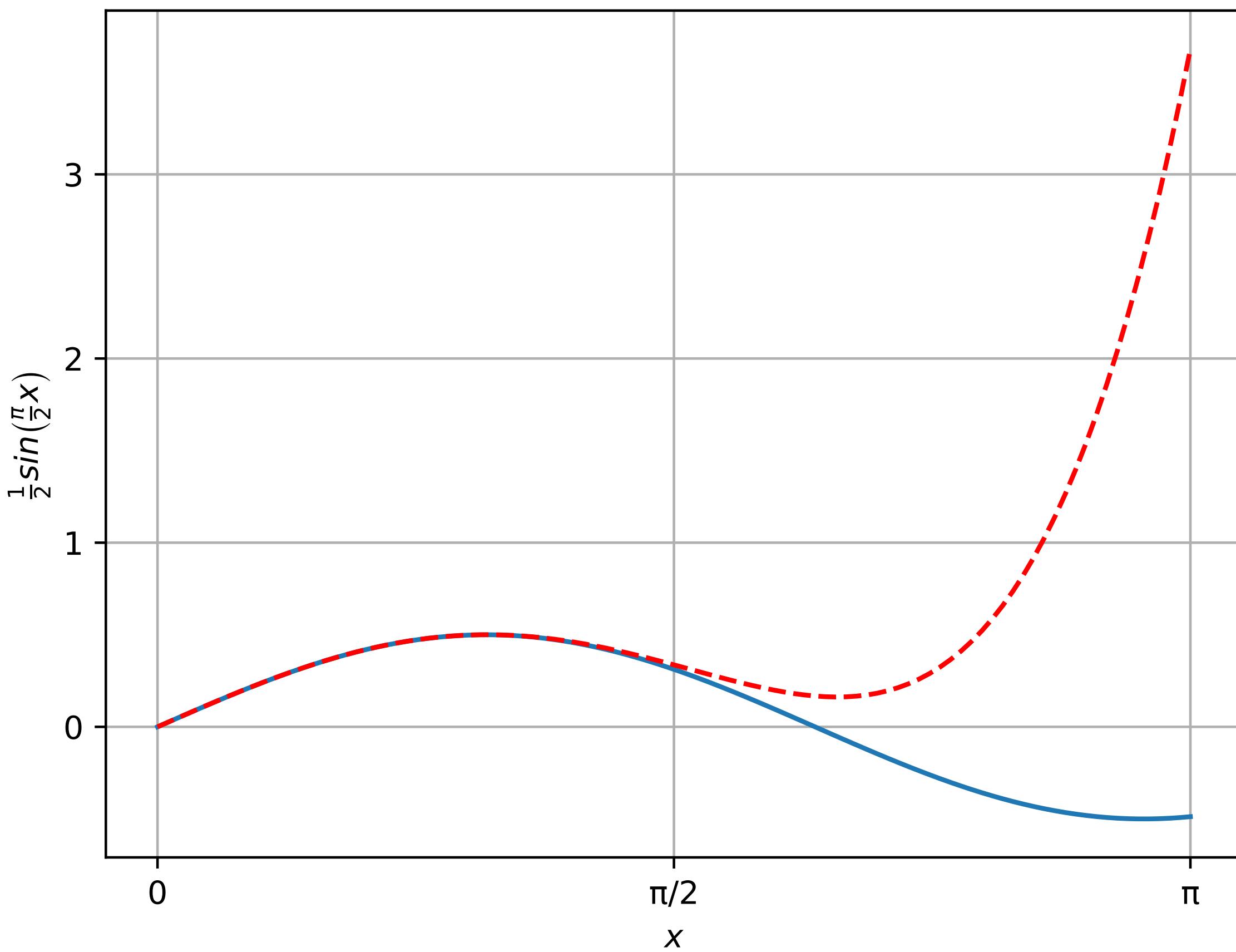
# SINGLE PRECISION SINE AND COSINE

	COUNT*	\$\$/INTER	# ARGUMENTS
SPCOS	AD	HALF	
SPSIN	TS	TEMK	
	TCF	SPT	
	CS	TEMK	
SPT	DOUBLE		
	TS	TEMK	
	TCF	POLLEY	

# Apollo 11 Implementierung

$$y(x) = 0.7853134 \cdot x - 0.3216147^3 + 0.036551 \cdot x^5$$

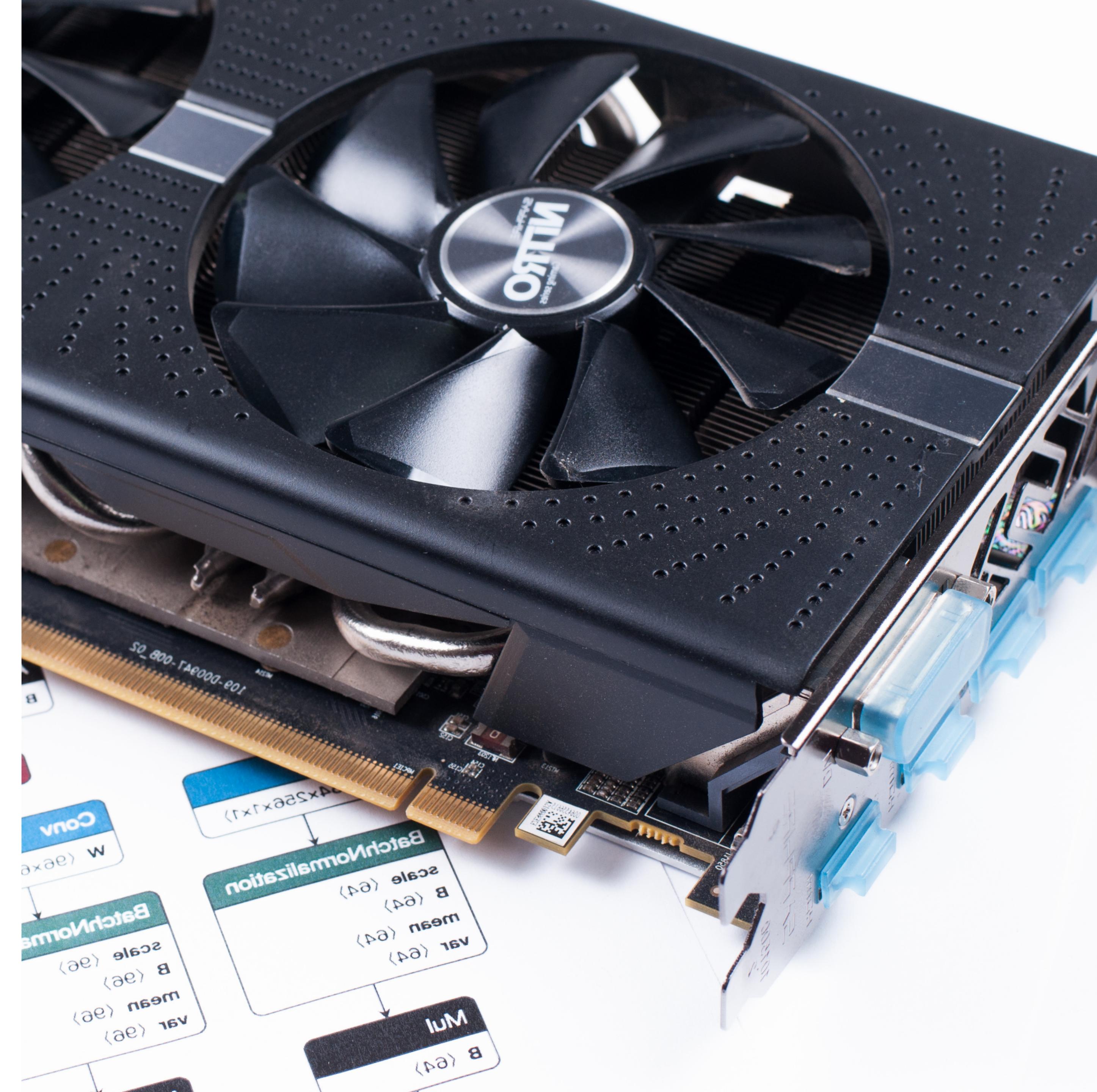
$$\frac{1}{2} \sin\left(\frac{\pi}{2}x\right)$$

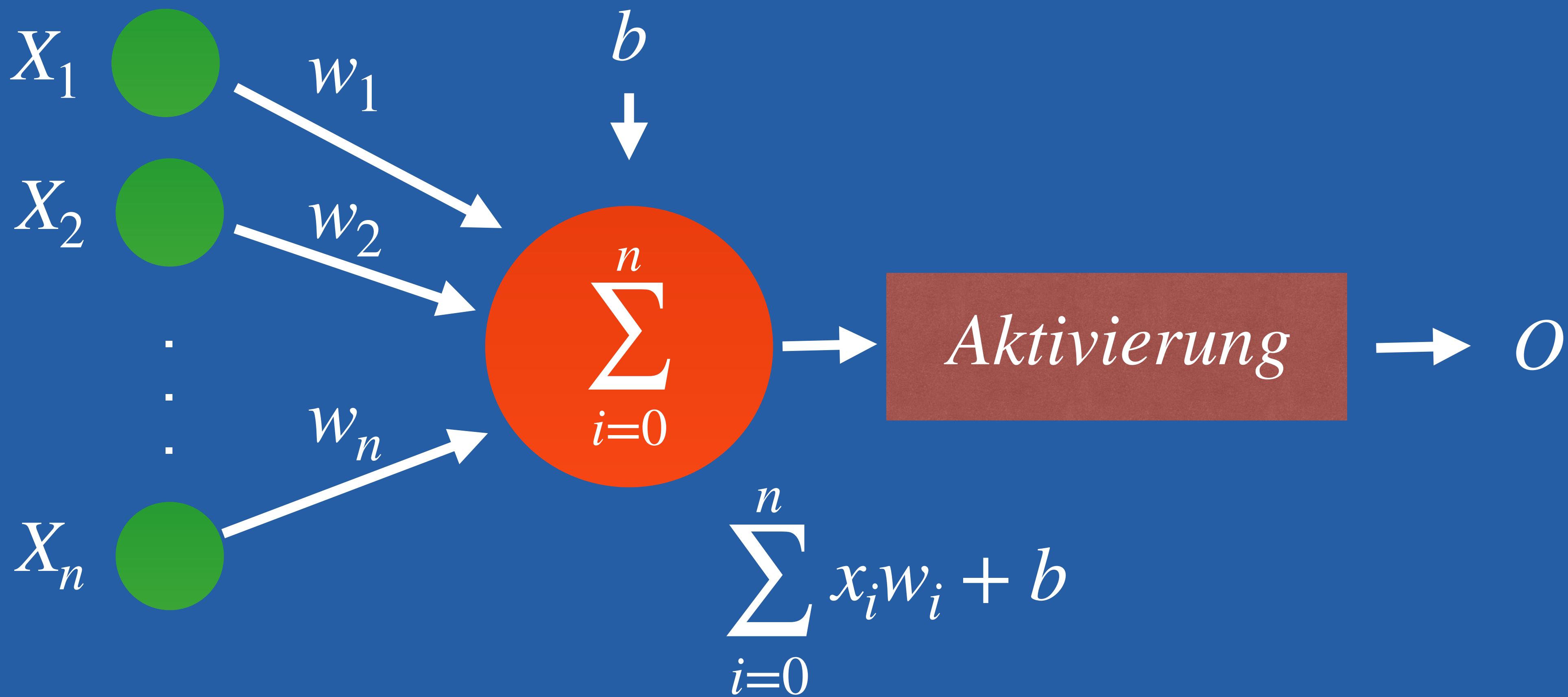


# Sinus Approximation

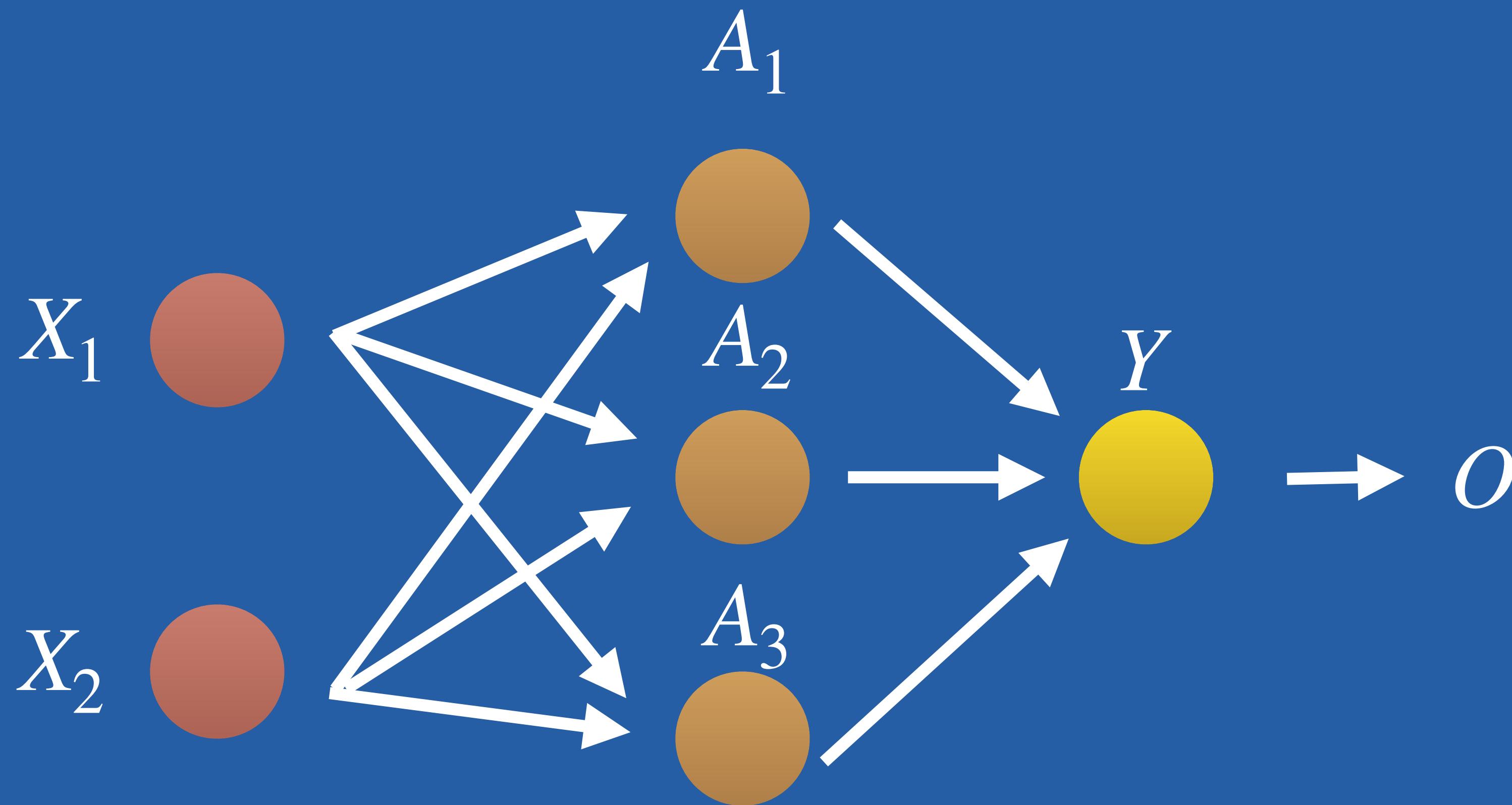
- Quellcode auf Github
- Kommentierter Quellcode beim **Fermat's Library Projekt**

# Neuronales Netz



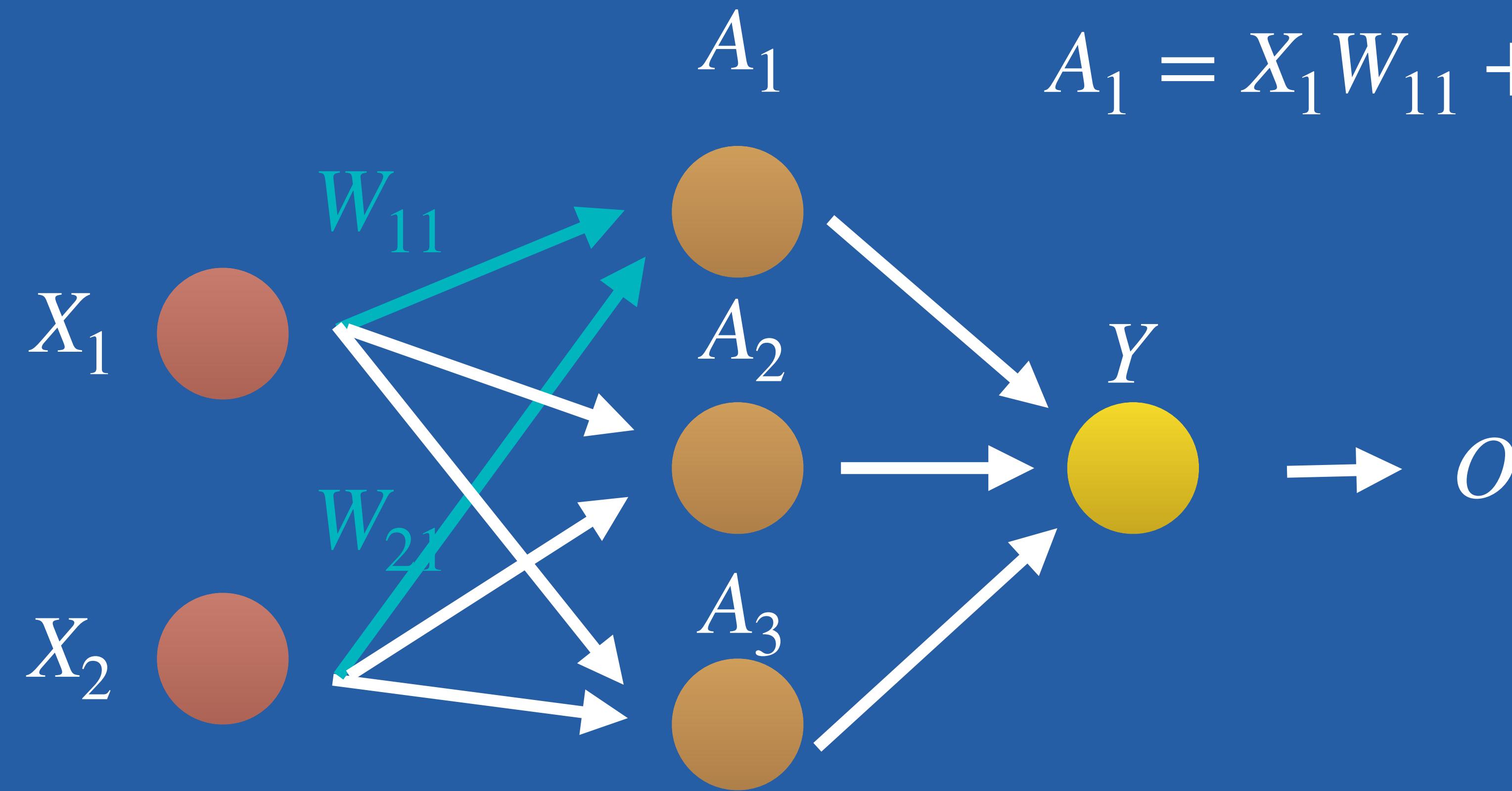


# Künstliches Neuron



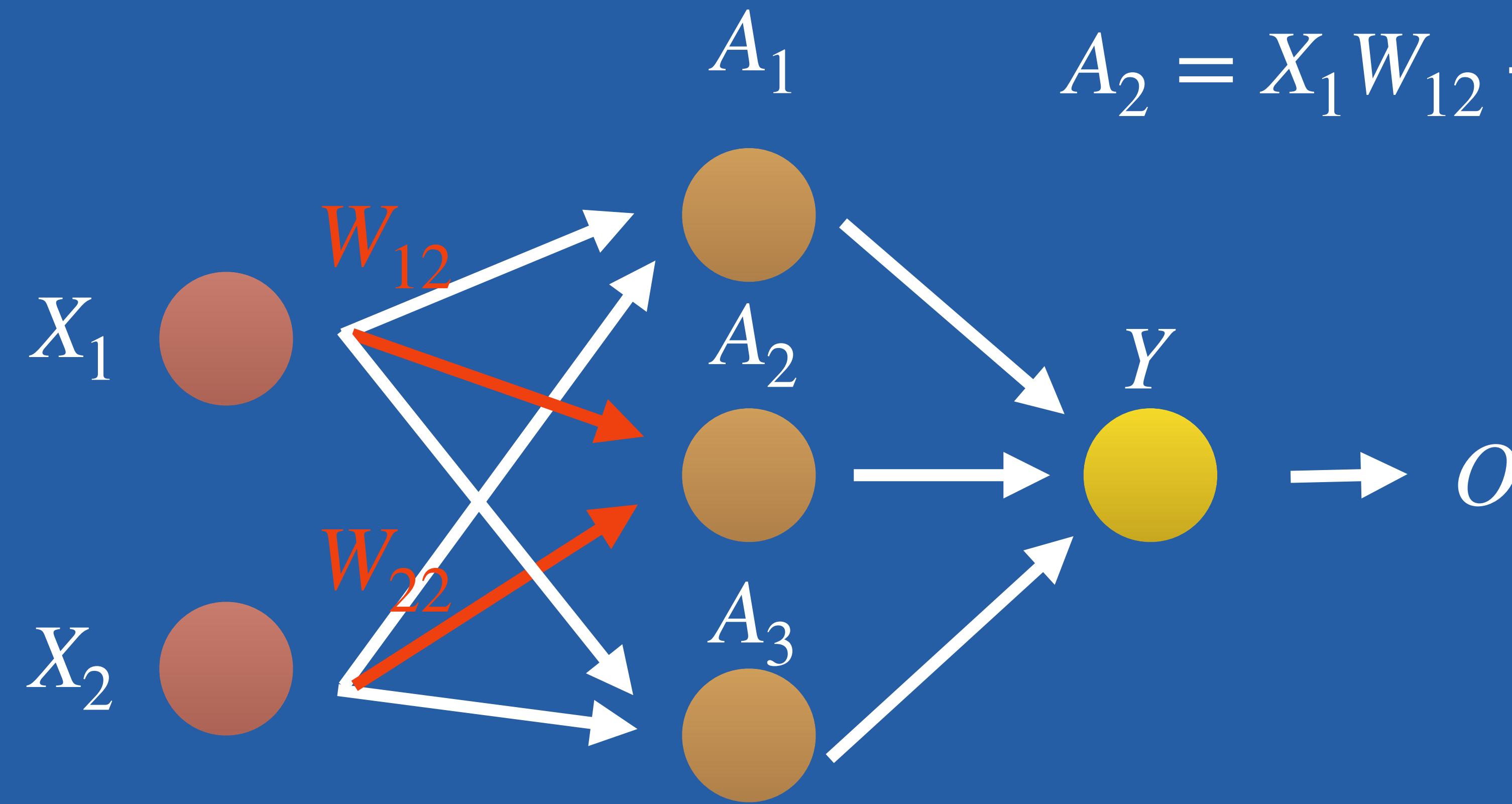
# Tiefes neuronales Netz

$$A_1 = X_1 W_{11} + X_2 W_{21}$$



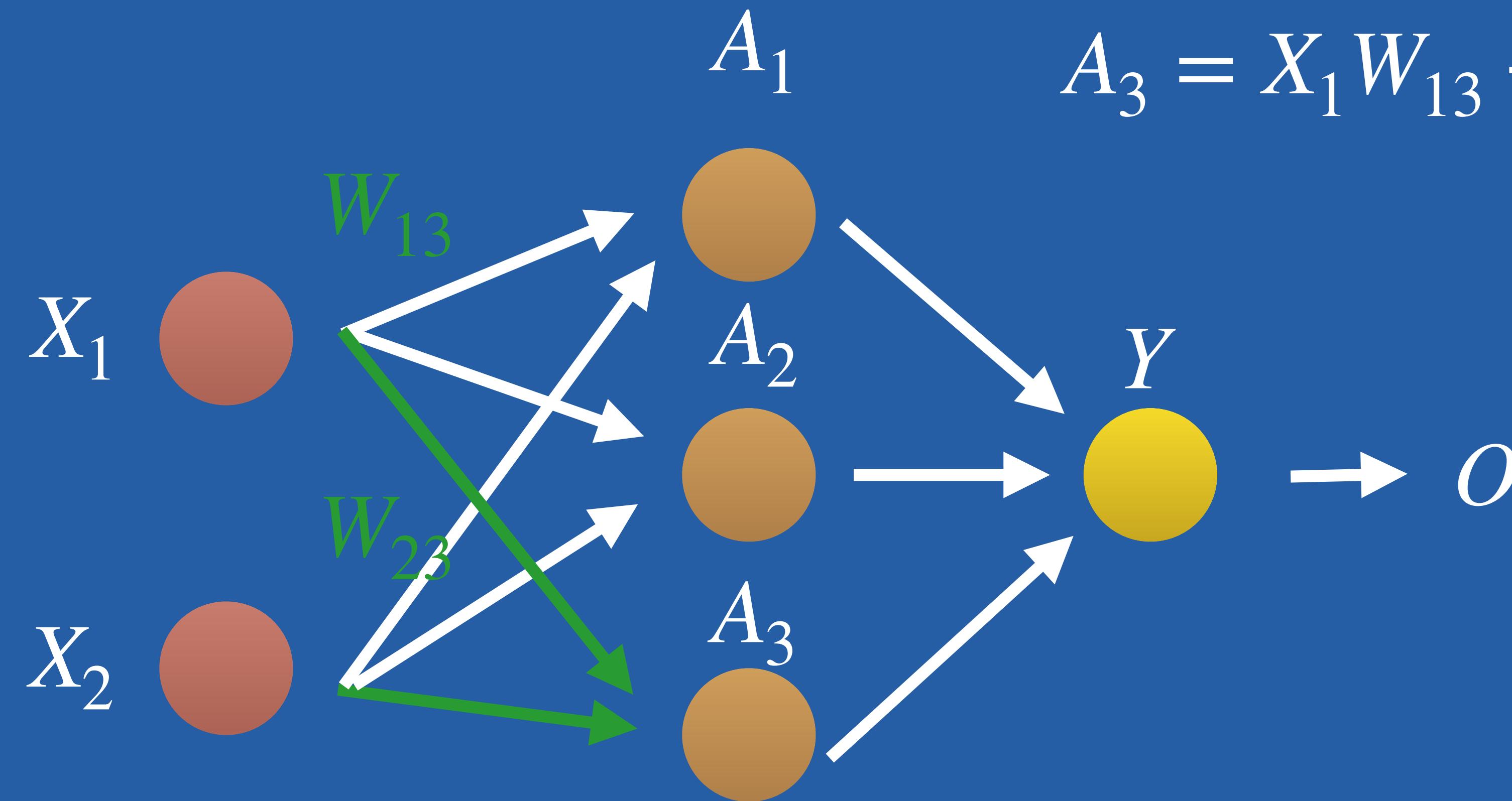
# Forward Propagation

$$A_2 = X_1 W_{12} + X_2 W_{22}$$



# Forward Propagation

$$A_3 = X_1 W_{13} + X_2 W_{23}$$



# Forward Propagation

$$A_{11} = X_1 W_{11} + X_2 W_{21}$$

$$A_{12} = X_1 W_{12} + X_2 W_{22}$$

$$A_{13} = X_1 W_{13} + X_2 W_{23}$$

$$W = \begin{bmatrix} W_{11} & W_{12} & W_{13} \\ W_{21} & W_{22} & W_{23} \end{bmatrix}$$

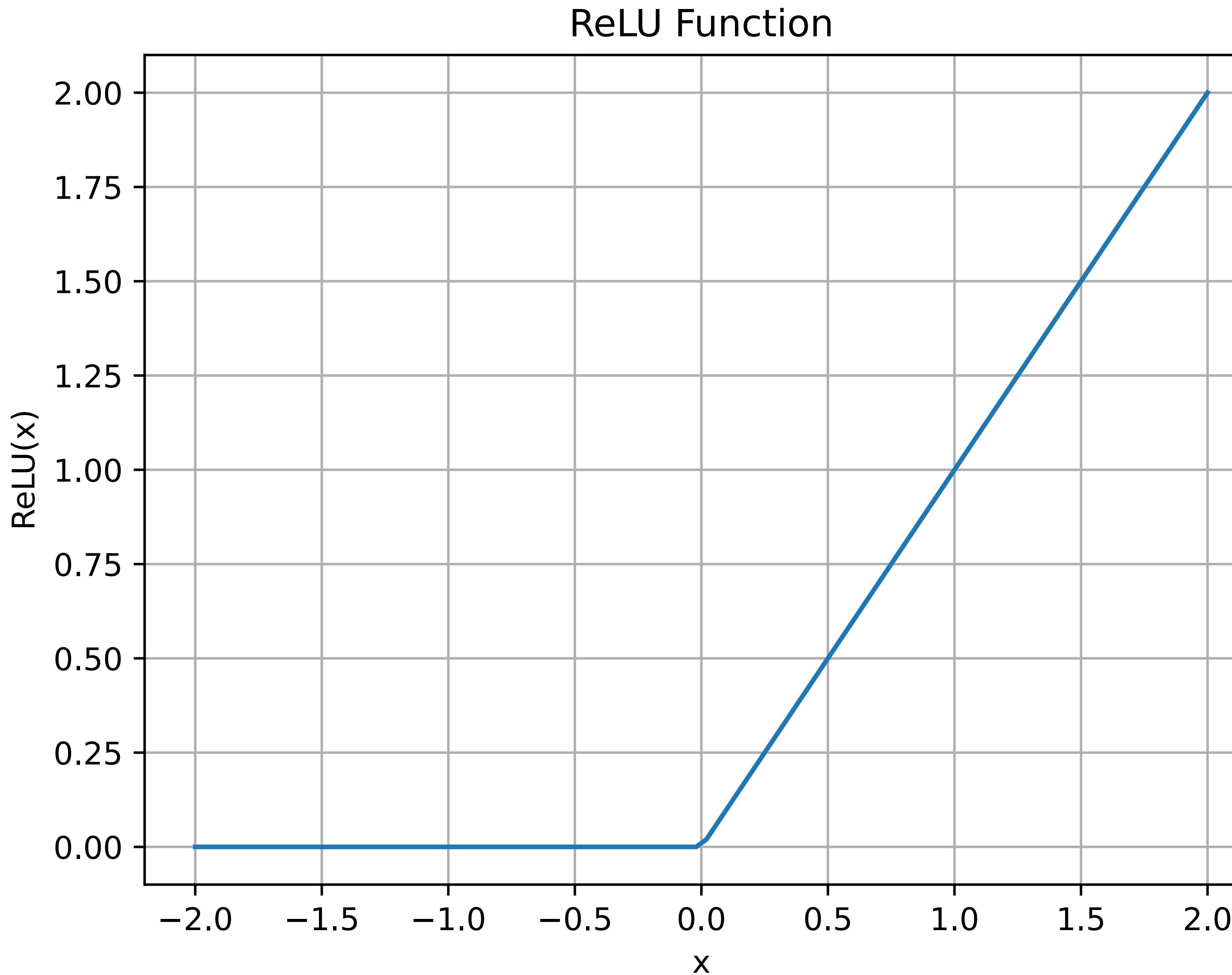
$$X = \begin{bmatrix} X_1 & X_2 \end{bmatrix} \quad A = \begin{bmatrix} A_{11} & A_{12} & A_{13} \end{bmatrix}$$

**Skalarprodukt = Matrizen  
Multiplikation**

$$A = X \cdot W$$

# Aktivierungsfunktion - ReLU(Gleichrichter)

$$f(x) = \max(0, x)$$



# PARADIGMA DES MASCHINELLEN LERNENS



# Vorhersagen

```
● ● ●

class SingleLayerPerceptron(inputSize: Int) {
    var weights: FloatArray = FloatArray(inputSize) { 0.0f } // Initialize weights to 0
    var bias: Float = 0.0f

    fun relu(x: Float): Float = if (x > 0) x else 0f

    fun forward(inputs: FloatArray): Float {
        var sum = 0.0f
        for (i in inputs.indices) {
            sum += inputs[i] * weights[i]
        }
        sum += bias
        return relu(sum)
    }
}
```

# Genaugkeit messen

Fehlerfunktion - **Mittlere quadratische Abweichung**

$$\frac{1}{N} \sum_{i=0}^N y_i + \hat{y}_i$$

# **Optimieren mit Fehlerrückführung**

**Gradientenabstiegsverfahren zum Fehler minimieren**

# Programmiersprachen



**Wir müssen über C sprechen**

**Python**

# Python 101

- Einrücken statt Klammern
- Die #1 Sprache für Datenwissenschaft
- ML Frameworks und Bibliotheken Ökosystem
- Werkzeuge

# Python 101

- Interaktive Kommandozeileapplikation - REPL
- Dependency management (**pip** und **pivenv**)
- Jupyter notebooks
- Unterstützung von mehreren Entwicklung Umgebungen

# Frameworks für maschinelles Lernen



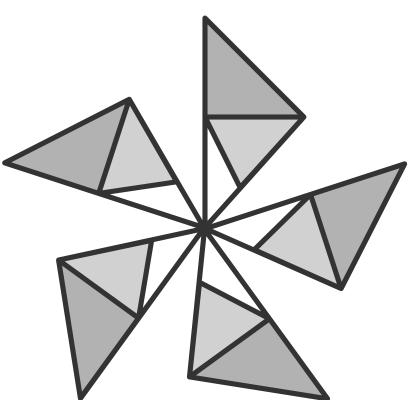
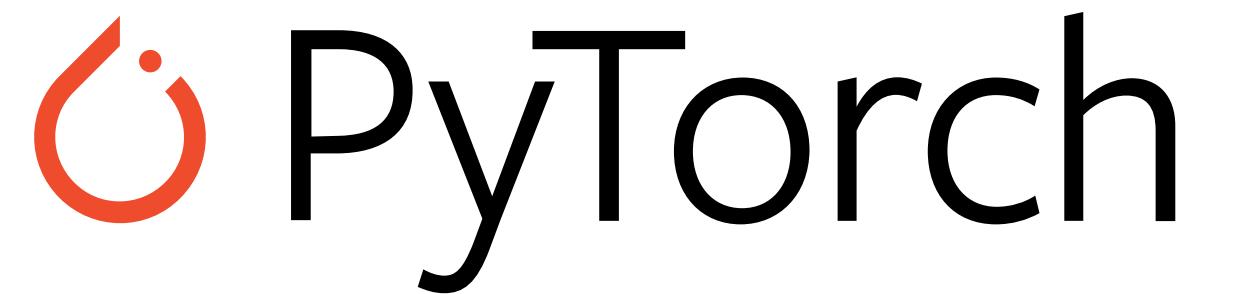
# ML Frameworks

## Mobile devices

- Tensorflow
- Pytorch
- ONNX Runtime



TensorFlow



ONNX  
RUNTIME

# Tensorflow

- Tensorflow, Tensorflow Lite, Tensorflow Micro
- Offizial Ports
- Community Ports

# PyTorch

- Pythonische Integration
- Computer Vision
- Natural Language Processing
- Starkes Fokus auf eine Automatische Differenzierung mit Autograd

# ONNX

- Offizielles Linux Foundation Projekt
- Microsoft als ein starker Unterstützer
- Unterstützung für Core ML

# ML Frameworks

## Mobile devices

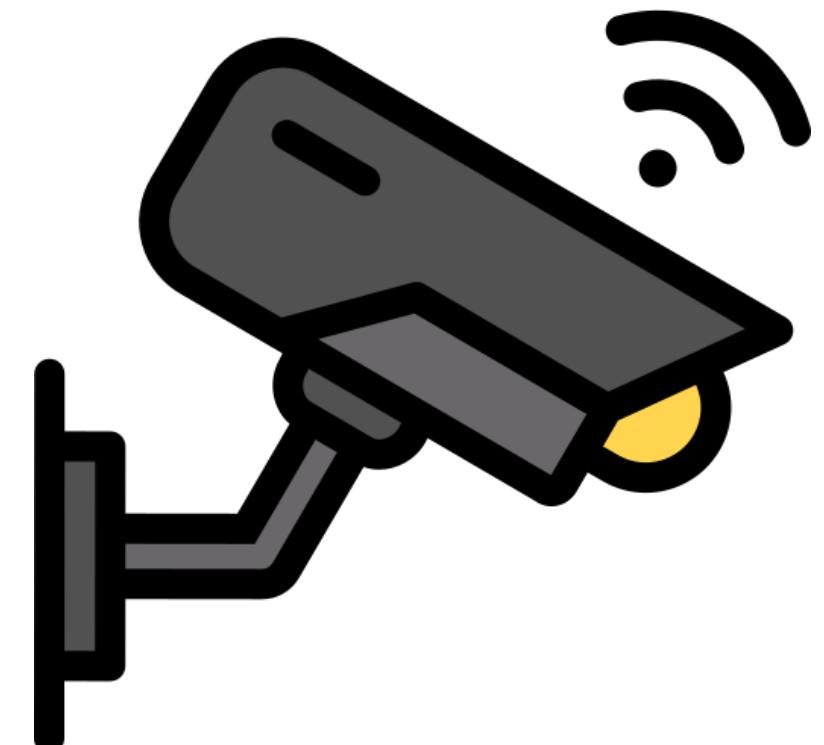
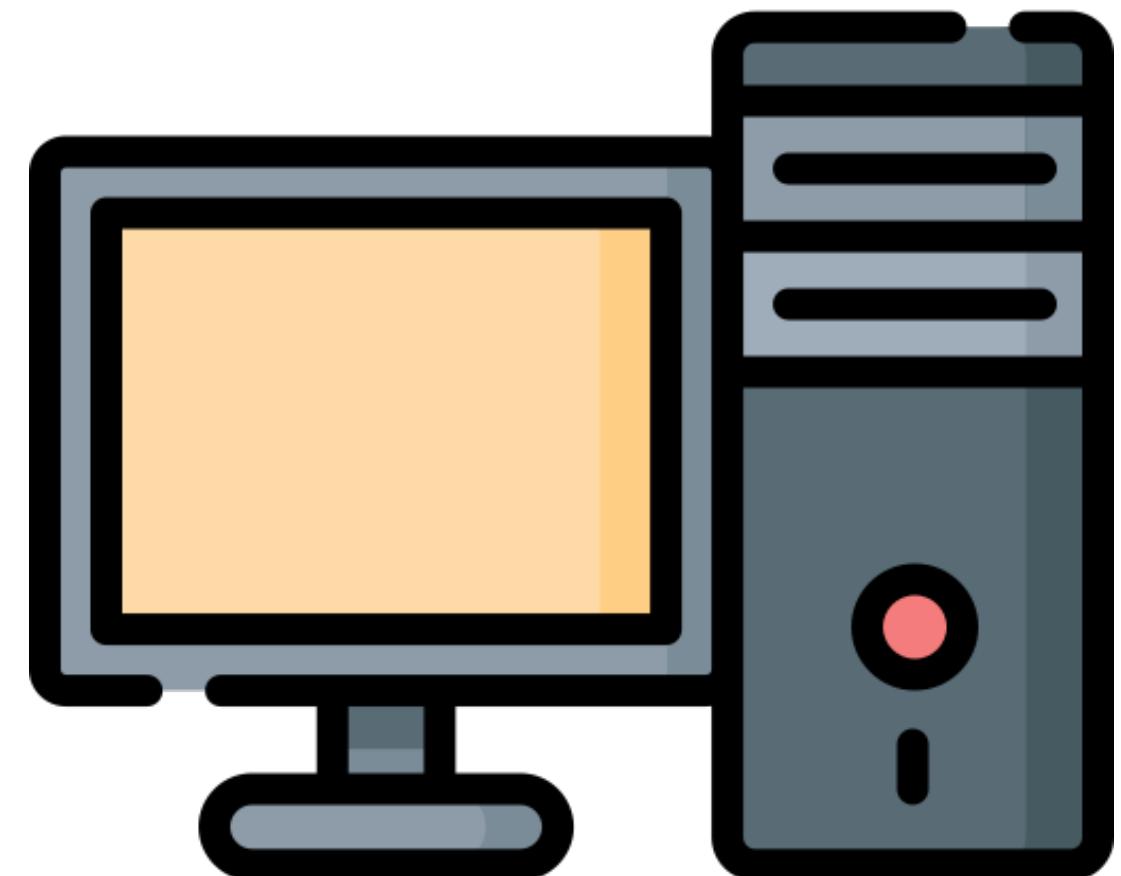
- Tensorflow Lite
- Pytorch Mobile
- Core ML

# ML Frameworks

## Edge computing und IoT

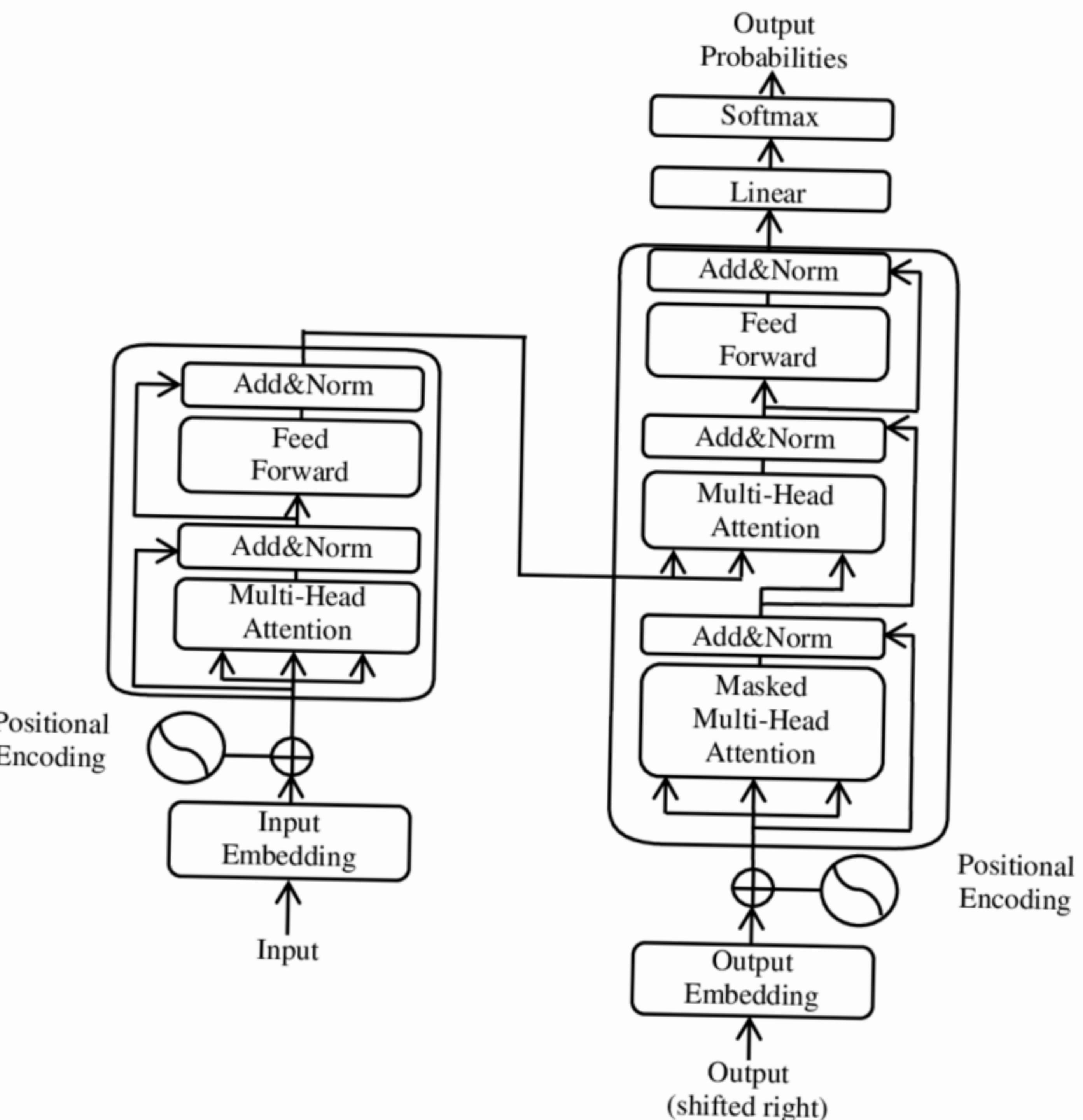
- TFLite Micro
- Edge Impulse
- Apache TVM, MicroTVM
- Cube.AI

# ML in Edge Computing und IoT



# Netztopologien

- **Convolutional Neural Network**
- **Rekurrente Neuronale Netze**
- **Transformers**



# Lösung mit ML -DEMO

- Appollo Navigation trainieren
- Modellinferenz - plain c
- Modellinferenz - TF



# Zusammenfassung

- ML sind Matrizen Multiplikationen
- ML mit Mobile, Edge Computing und IoT
- Wir lernen jetzt Python





# Fragen?

**Vielen Dank**

# Quellcode

- <https://github.com/michalharakal/ml4iot>



# Linksammlung

- <https://fermatslibrary.com/s/apollo-11-implementation-of-trigonometric-functions>
- <https://onlinelibrary.wiley.com/doi/abs/10.1111/j.1365-2818.1990.tb02967.x>
- <https://paperswithcode.com>
- <https://en.wikipedia.org/wiki/AlphaGo>
- [https://aiimpacts.org/scale-of-the-human-brain/#Number of neurons in the brain](https://aiimpacts.org/scale-of-the-human-brain/#Number_of_neurons_in_the_brain)
- <https://onlinelibrary.wiley.com/doi/abs/10.1111/j.1365-2818.1990.tb02967.x>
- [https://storage.googleapis.com/deepmind-media/AlphaCode/competition level code generation with alphacode.pdf](https://storage.googleapis.com/deepmind-media/AlphaCode/competition_level_code_generation_with_alpha_code.pdf)
- [https://en.wikipedia.org/wiki/Apollo\\_Guidance\\_Computer](https://en.wikipedia.org/wiki/Apollo_Guidance_Computer)
- <http://www.ibiblio.org/apollo>
- <https://github.com/michalharakal/flagsam>
- [https://www.tensorflow.org/lite/examples/pose estimation/overview](https://www.tensorflow.org/lite/examples/pose_estimation/overview)
- <https://github.com/breandan/kotlingrad>

# Bildernachweis

- (C) Adriana Harakalova, 2021
- Logos PyTorch, Tensorflow, ONNX Runtime
- Slide 44:
  - ,image: computer-desktop.png'. This slide has been designed using images from [flaticon.com](https://flaticon.com)
  - ,image: cctv.png'. This slide has been designed using images from [flaticon.com](https://flaticon.com)