Deep Learning mit Keras und TensorFlow

Motivation

- Einführung Deep Learning
- Ein Neuronales Netz trainieren (Keras und TensorFlow)
- Selbststudium ermöglichen

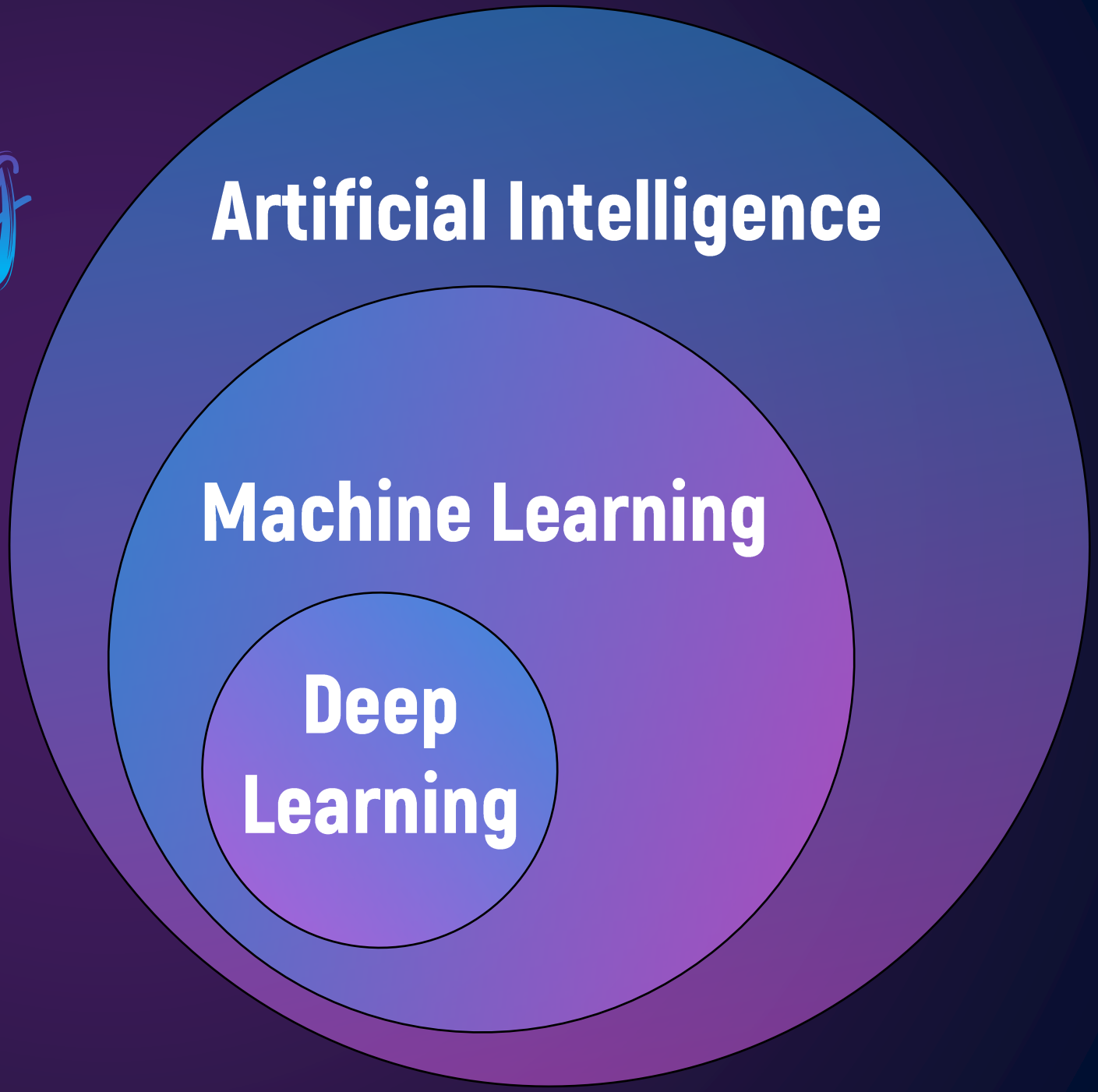
Inhalt

- Deep Learning
- Neuronales Netze
- Keras und TensorFlow (Python)
- Google Colab (Jupyter Notebook)

Deep Learning

Was ist

Deep Learning



Deep Learning

Warum brauchen wir

Deep Learning

https://hackernoon.com/

Deep Learning

Typische Deep Learning Probleme

https://www.zooroyal.de

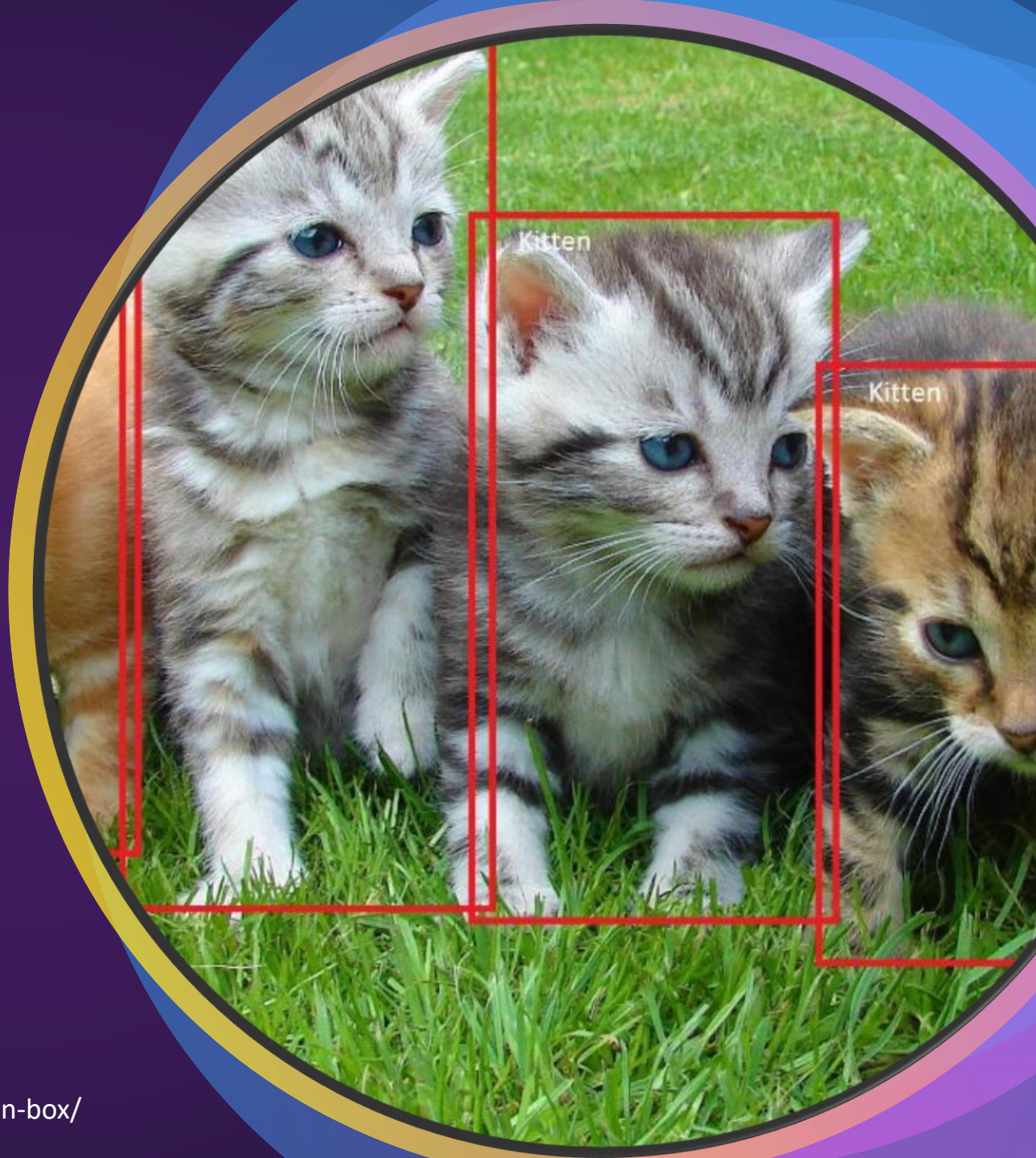
https://praxistipps.focus.de

Classification

Deep Learning

Typische Deep Learning Probleme

Object Detection



https://www.microsoft.com/developerblog/2017/04/10/end-end-object-detection-box/

Deep Learning

Typische

Deep Learning

Probleme

Natural Language Processing

Hey Alexa...

Deep Learning

Typische

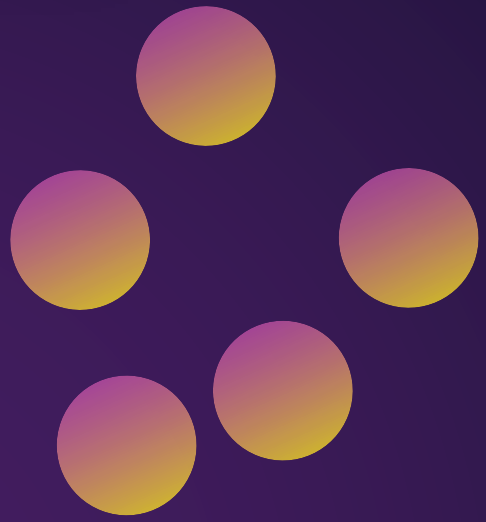
Deep Learning

Probleme

Regression

https://medium.com/mlreview/a-simple-deep-learning-model-for-stock-price-prediction-using-tensorflow-30505541d877

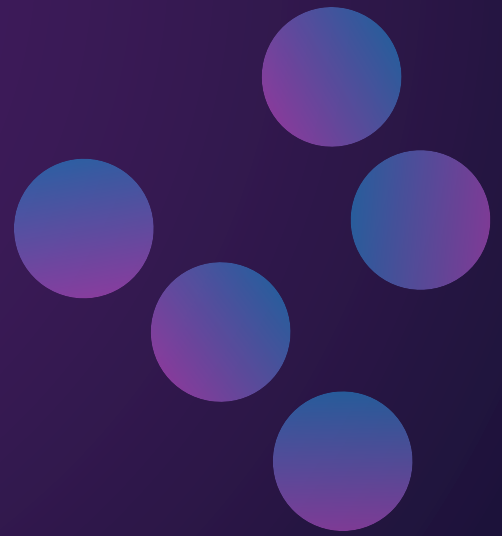
Deep Learning



Typische

Deep Learning

Probleme



Clustering

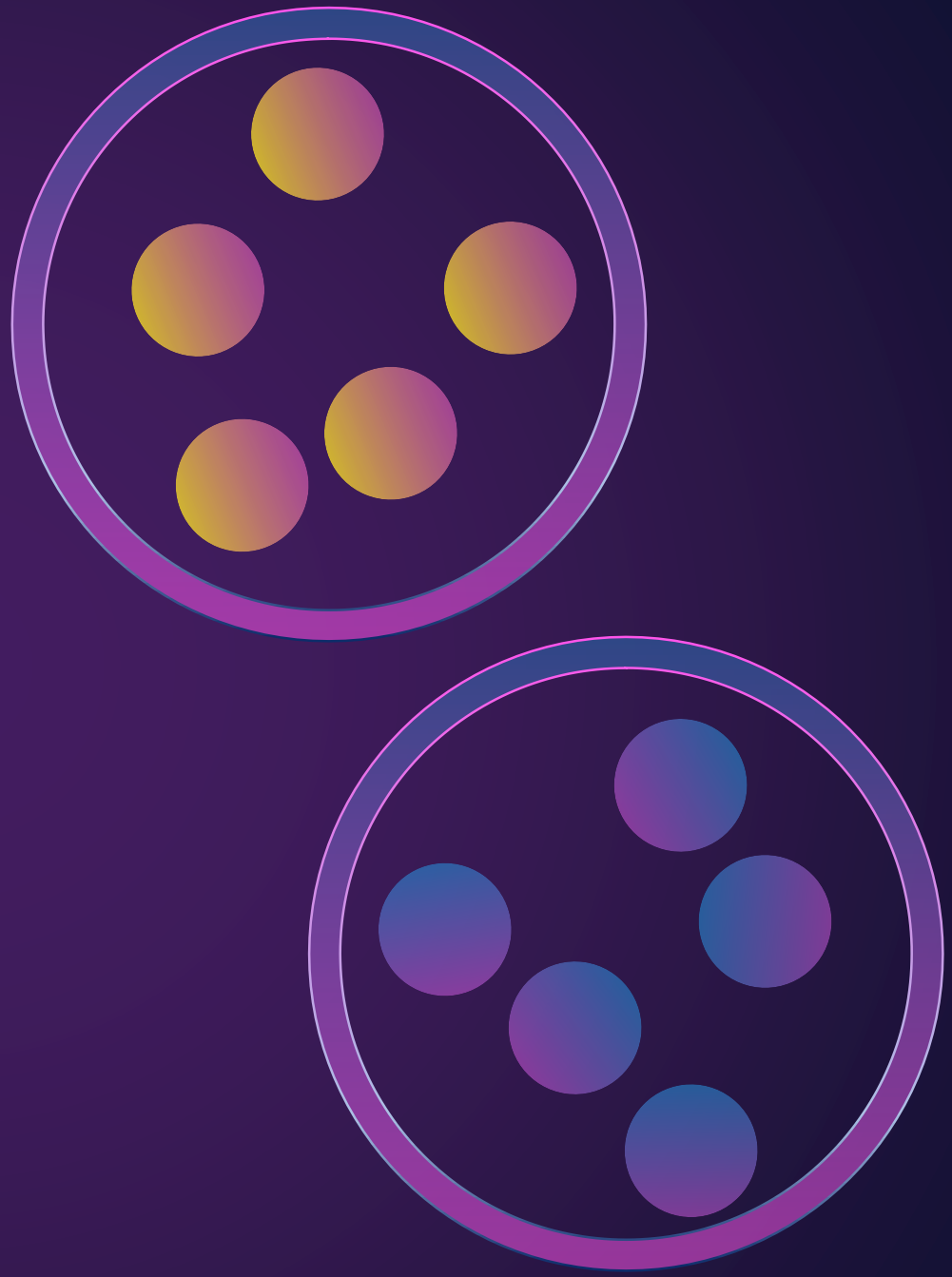
Deep Learning

Typische

Deep Learning

Probleme

Clustering



Deep Learning

Typische Deep Learning Probleme

https://www.heikovaneckert.de

https://www.fotocommunity.de

Games

Deep Learning

Wie funktioniert

Deep Learning

Deep Learning

Problem

https://theleadershipnetwork.com/

Wie funktioniert

Deep Learning

Deep Learning

Problem

https://theleadershipnetwork.com/

Daten

http://farefwd.com/

Wie funktioniert

Deep Learning

Deep Learning

Wie funktioniert

Deep Learning

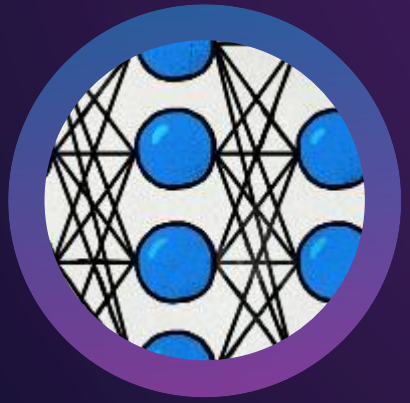
Problem

https://theleadershipnetwork.com/

Daten

http://farefwd.com/

Model



https://medium.com/@datamonsters/

Deep Learning

Wie funktioniert

Deep Learning

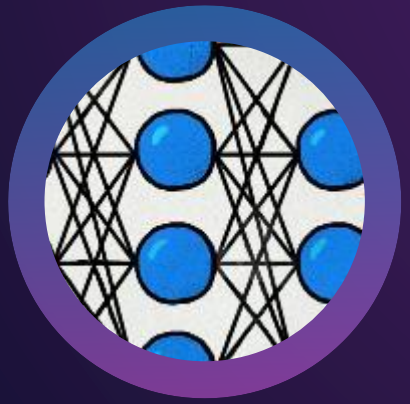
Problem

https://theleadershipnetwork.com/

Daten

http://farefwd.com/

Model



https://medium.com/@datamonsters/

Algorithmus

https://oupeltglobalblog.com/2018/03/27/

Deep Learning

Wie funktioniert

Deep Learning

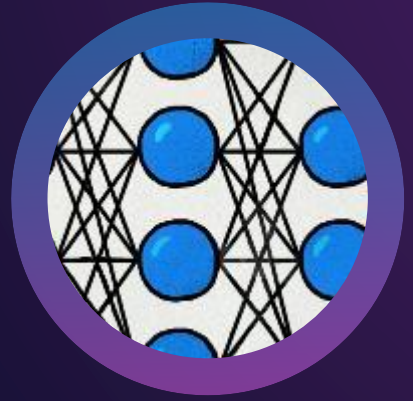
Problem

https://theleadershipnetwork.com/

Daten

http://farefwd.com/

Model

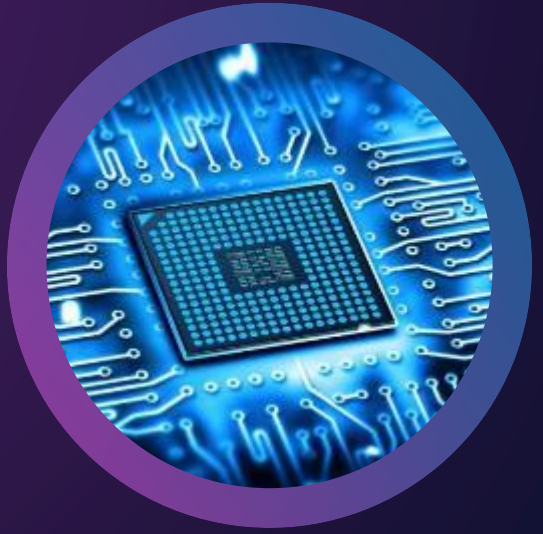


https://medium.com/@datamonsters/

Algorithmus

https://oupeltglobalblog.com/2018/03/27/

Rechenzeit und Leistung



https://www.pc-magazin.de/

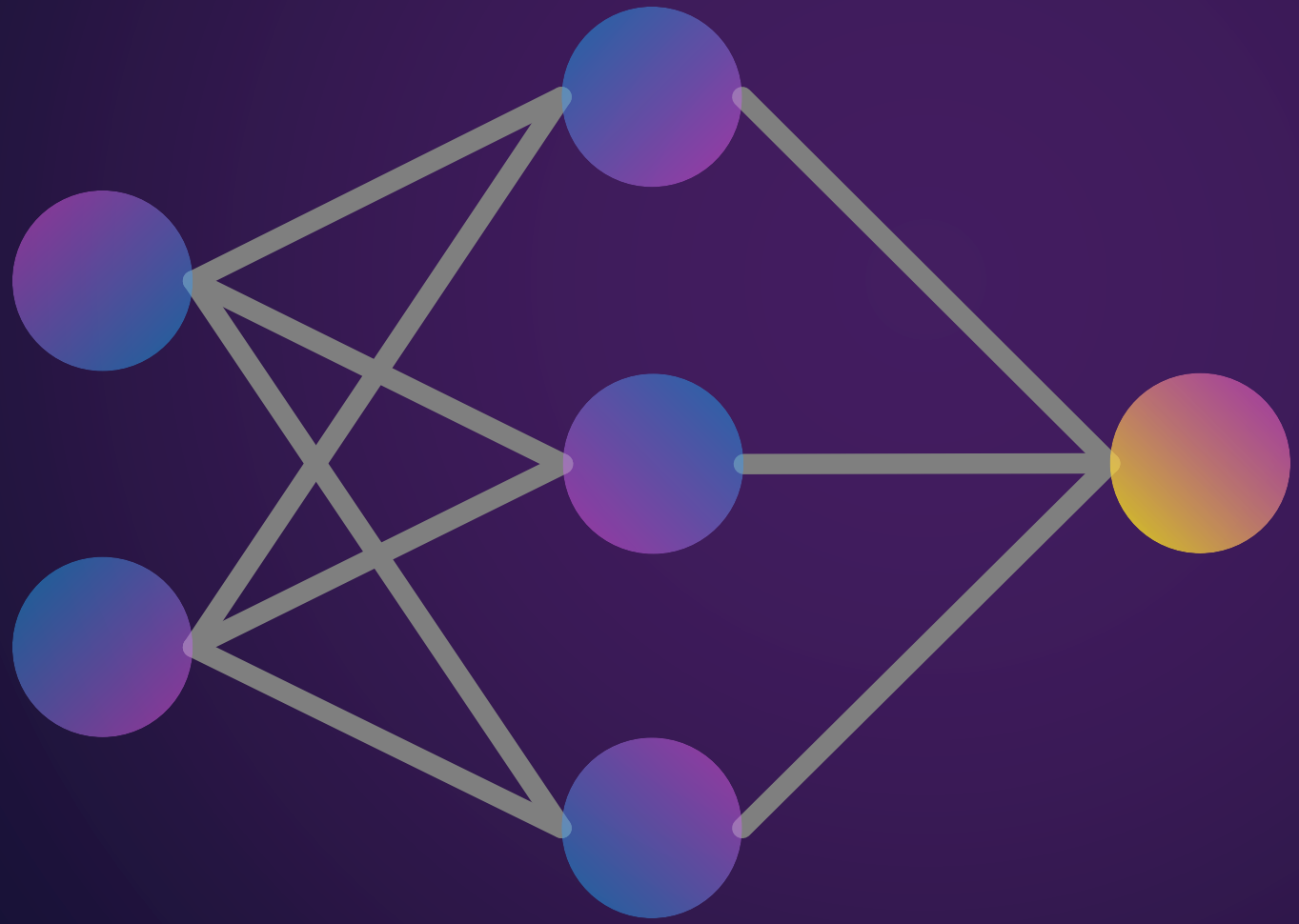
Neuronale Netze

Und ihre Geschichte

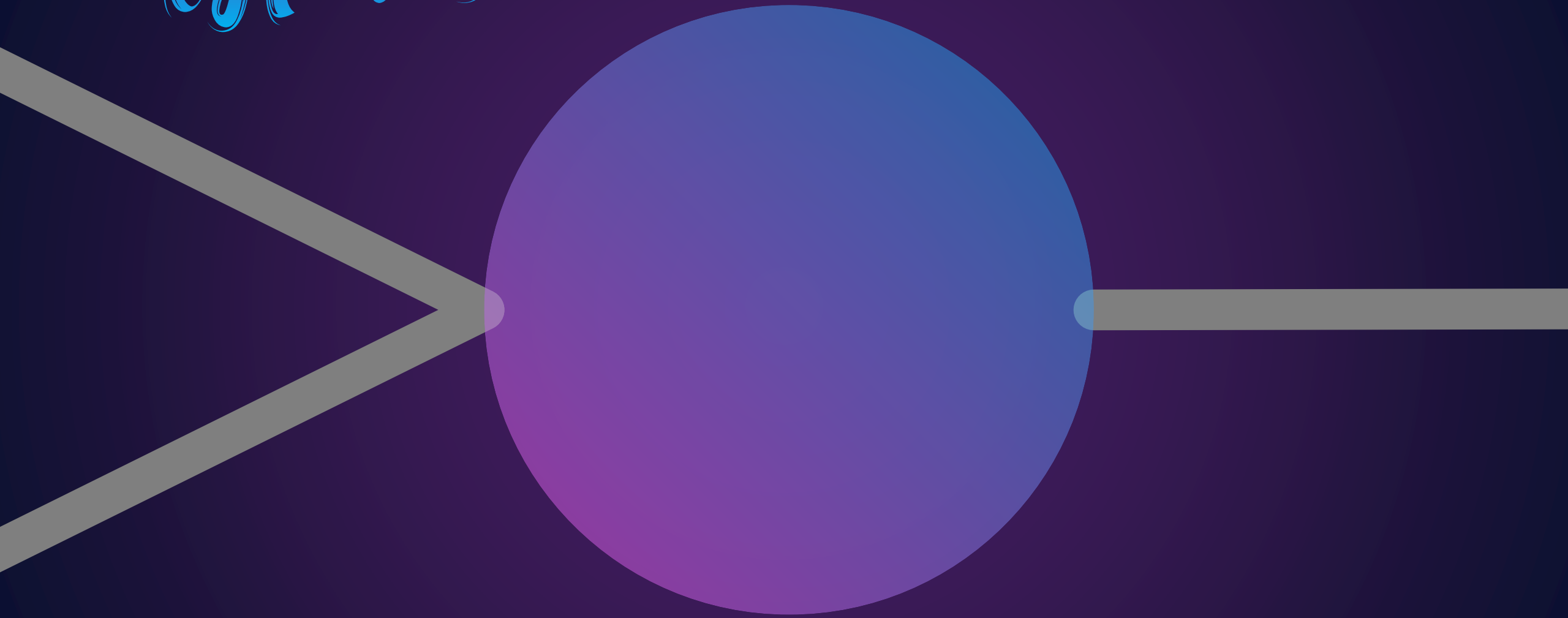
- Approximation belibiger Funktion
- 1943: Mathematisches Modell
- 1986: Stabiler Lernalgorithmus
- ‘loosely modeled after the human brain‘
- Vielzahl an Architecturen

Neuronale Netze

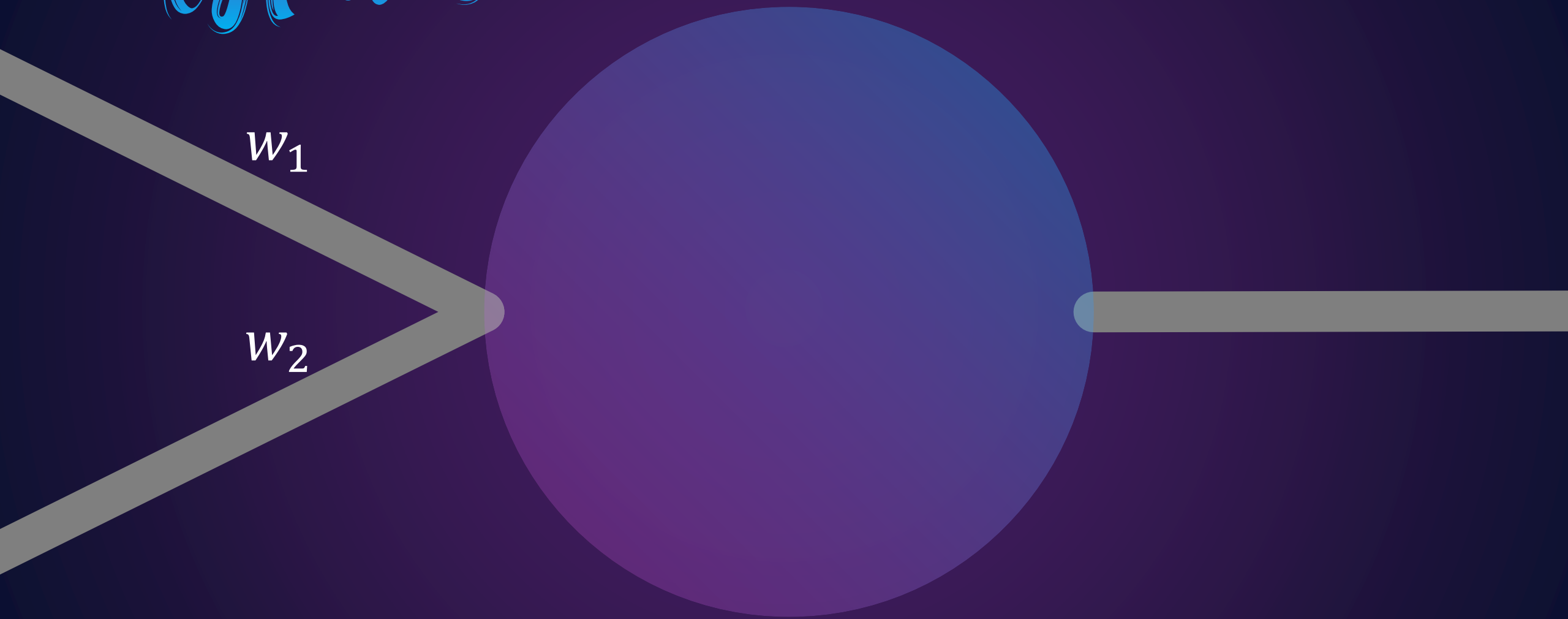
Neuronale Netze



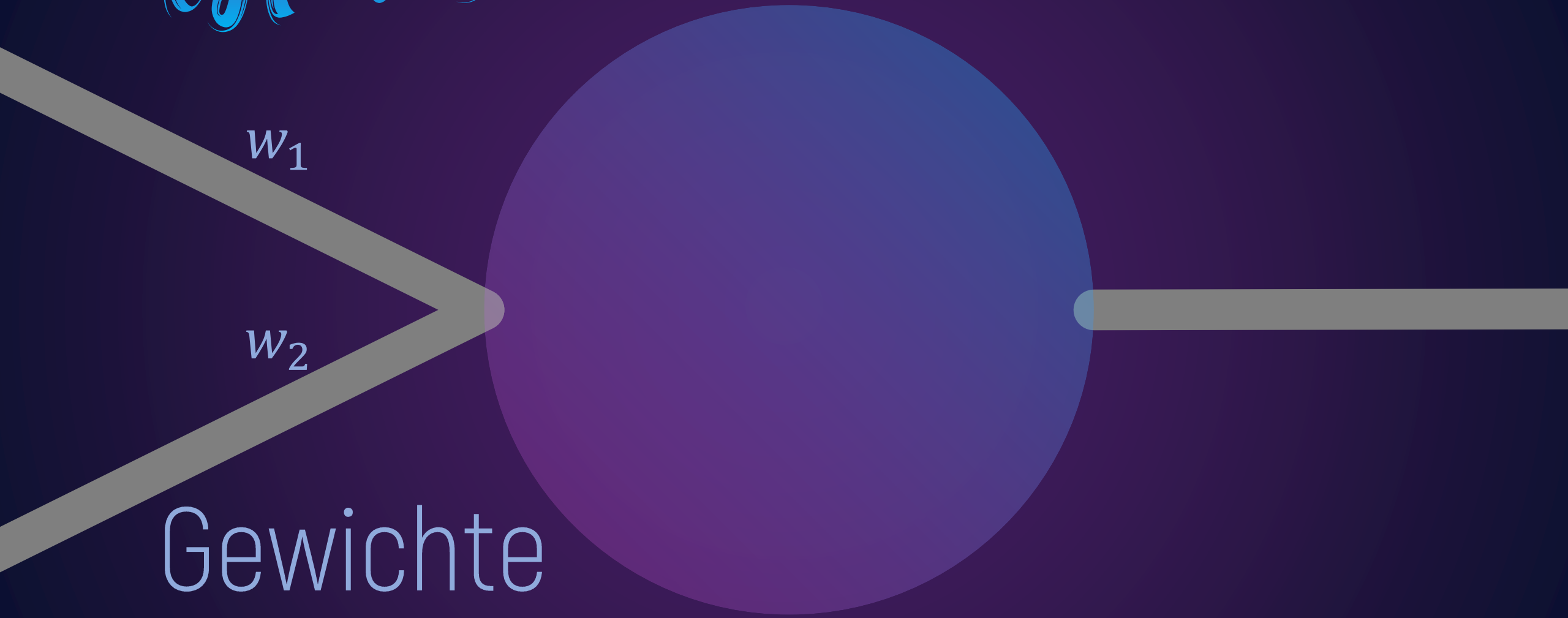
Neuron



Neuron



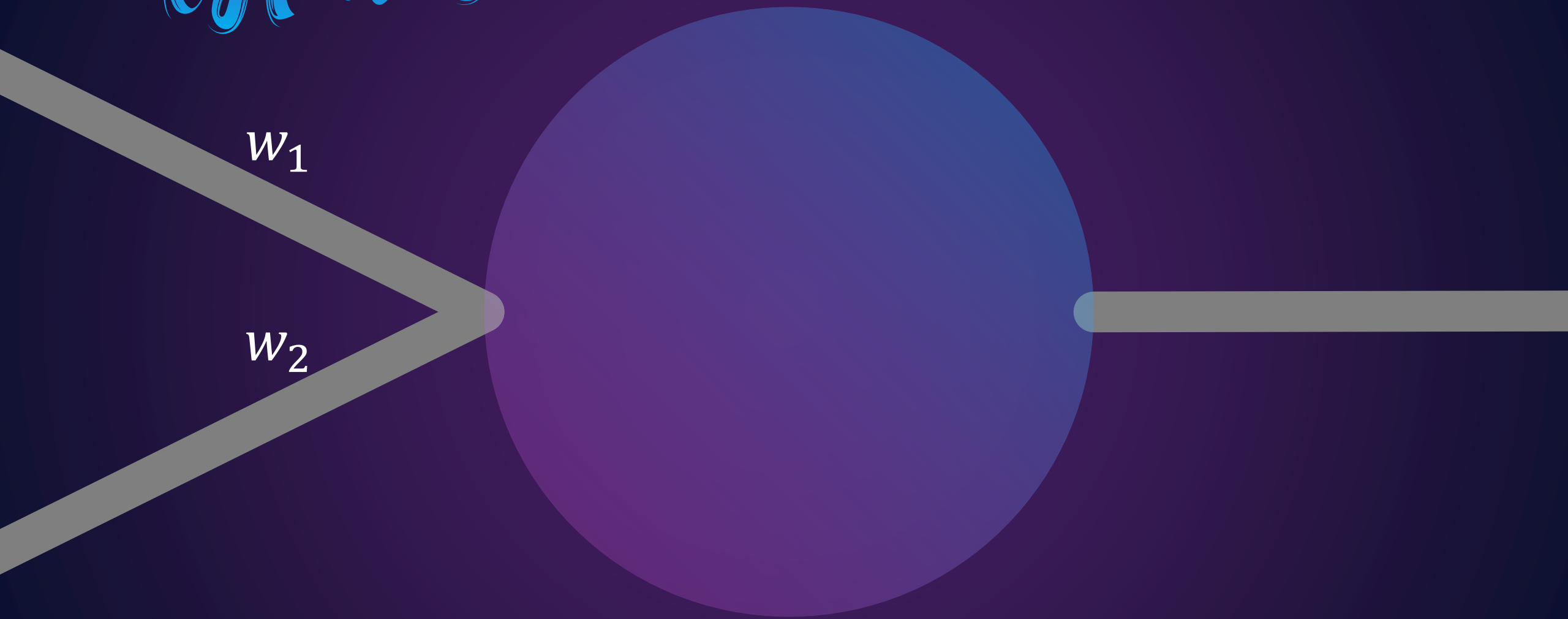
Neuron



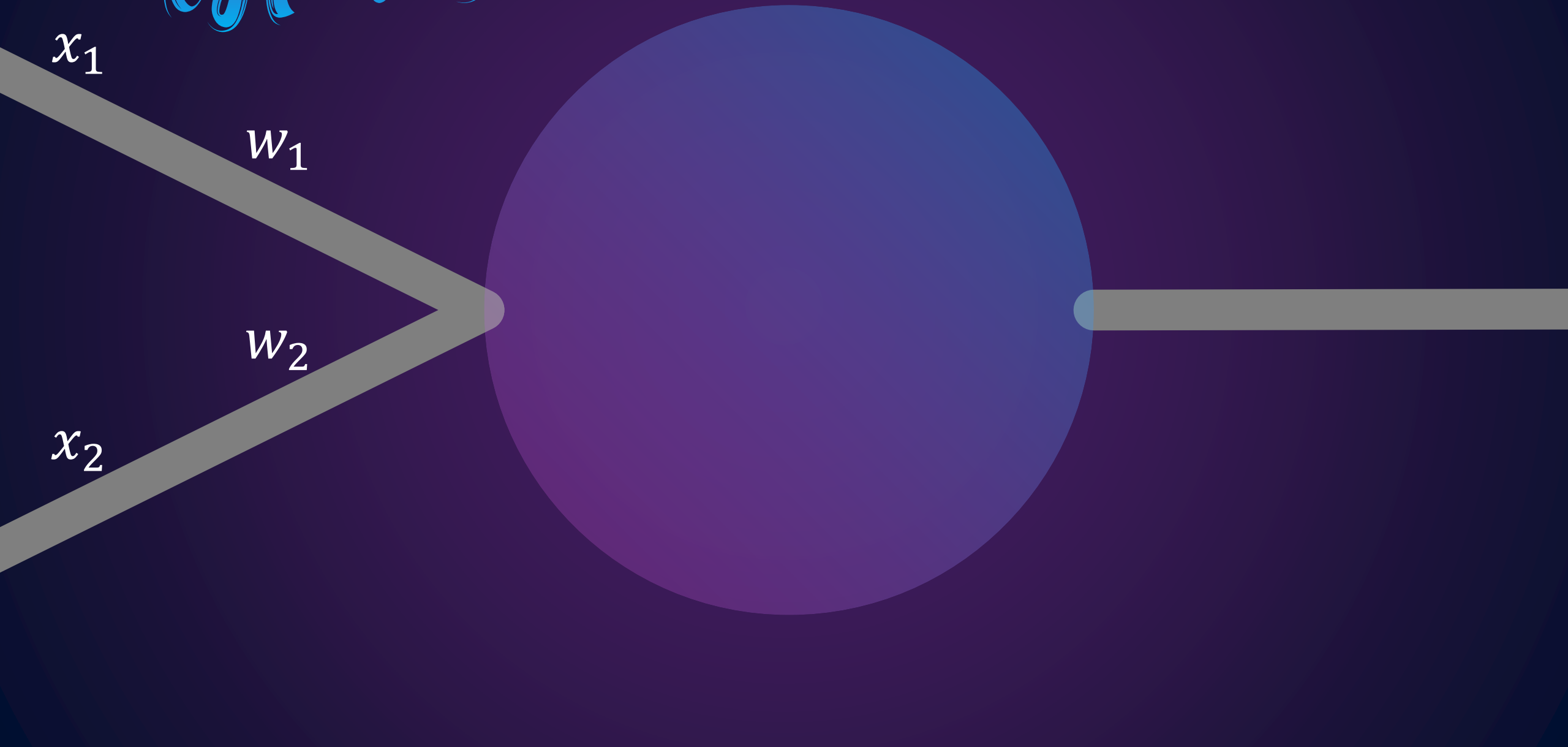
Gewichte

weights

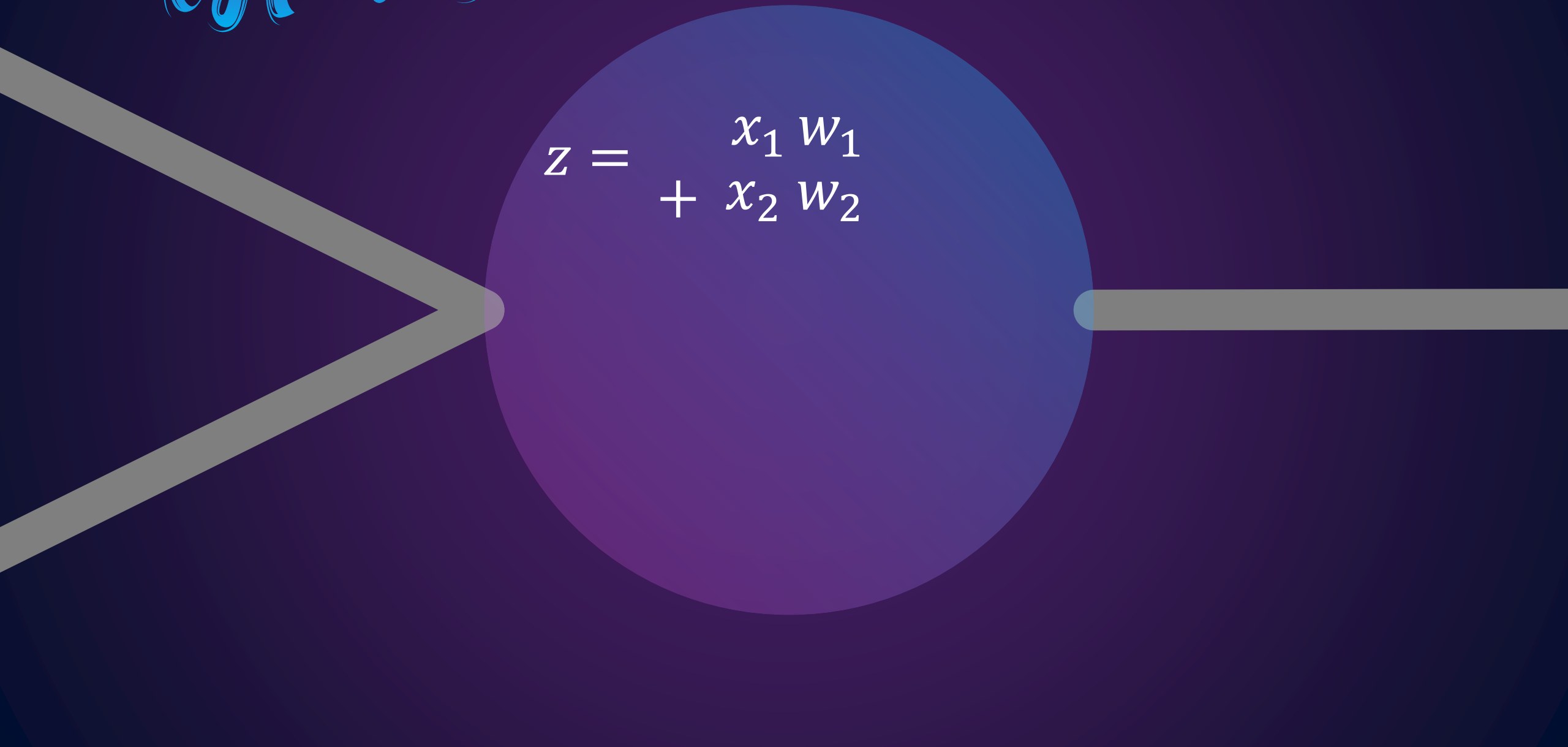
Neuron



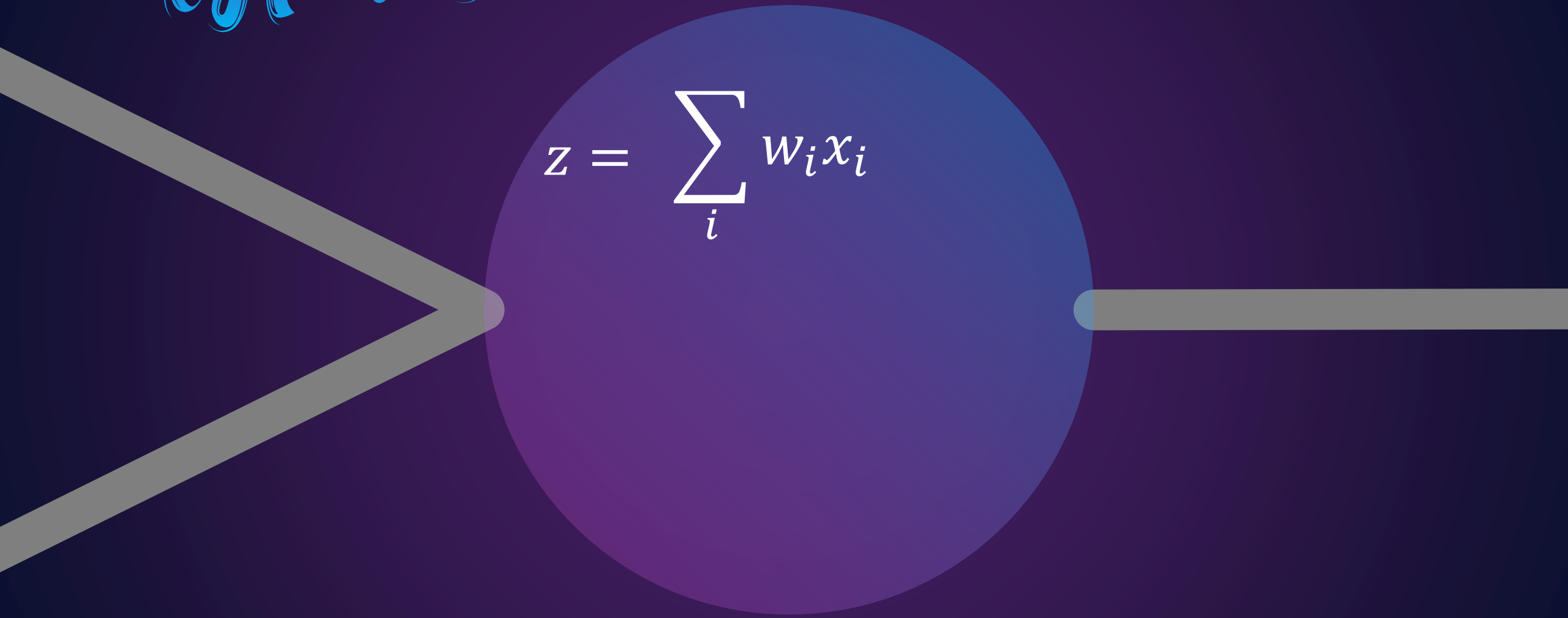
Neuron



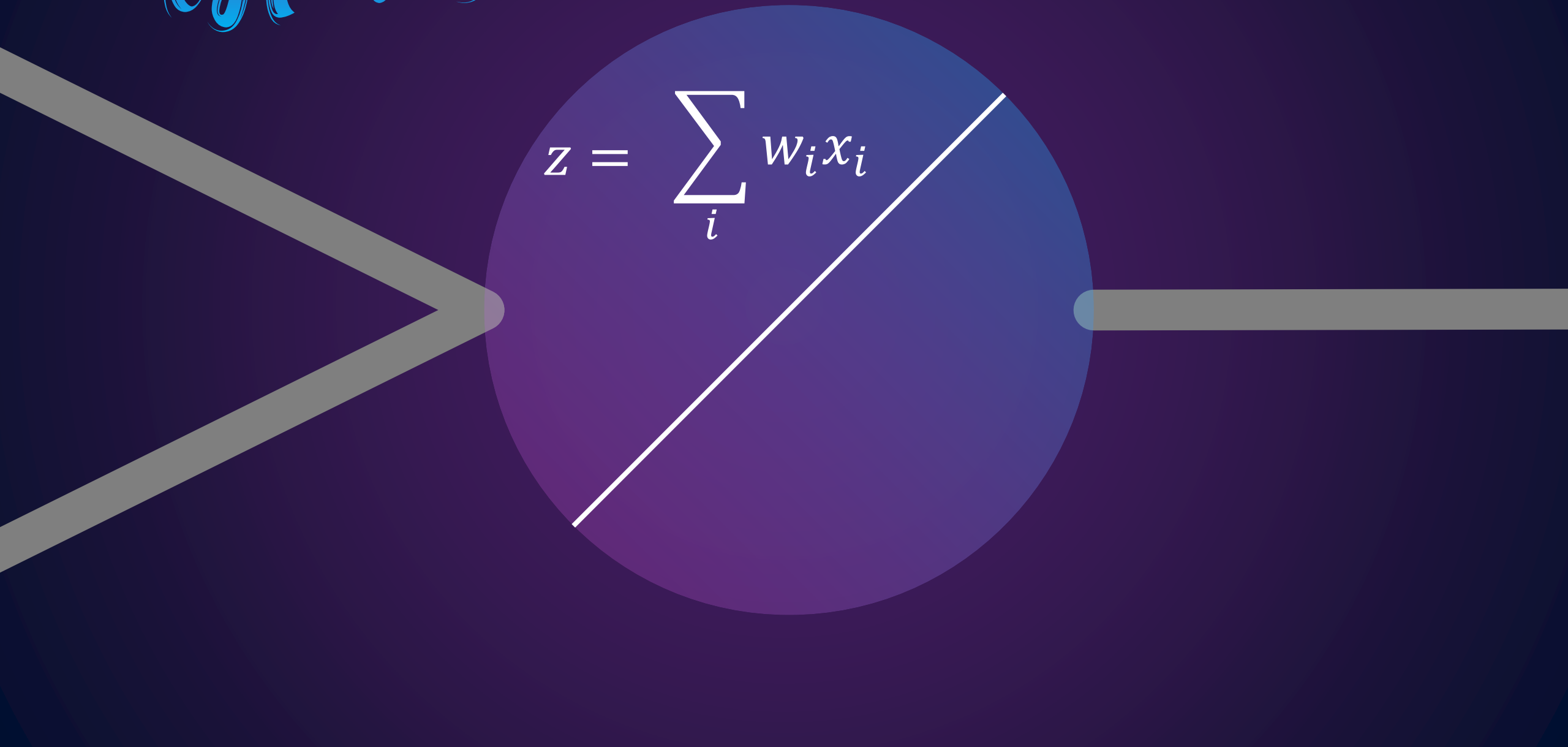
Neuron



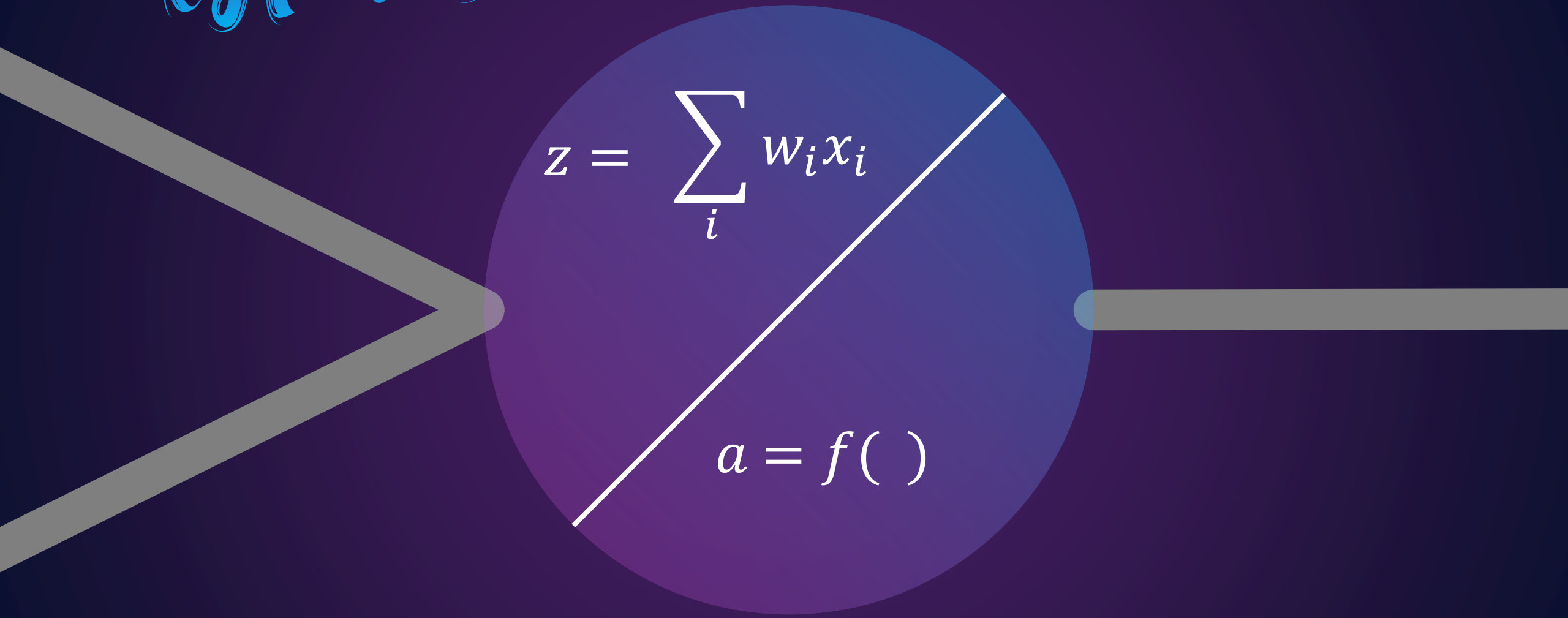
Neuron



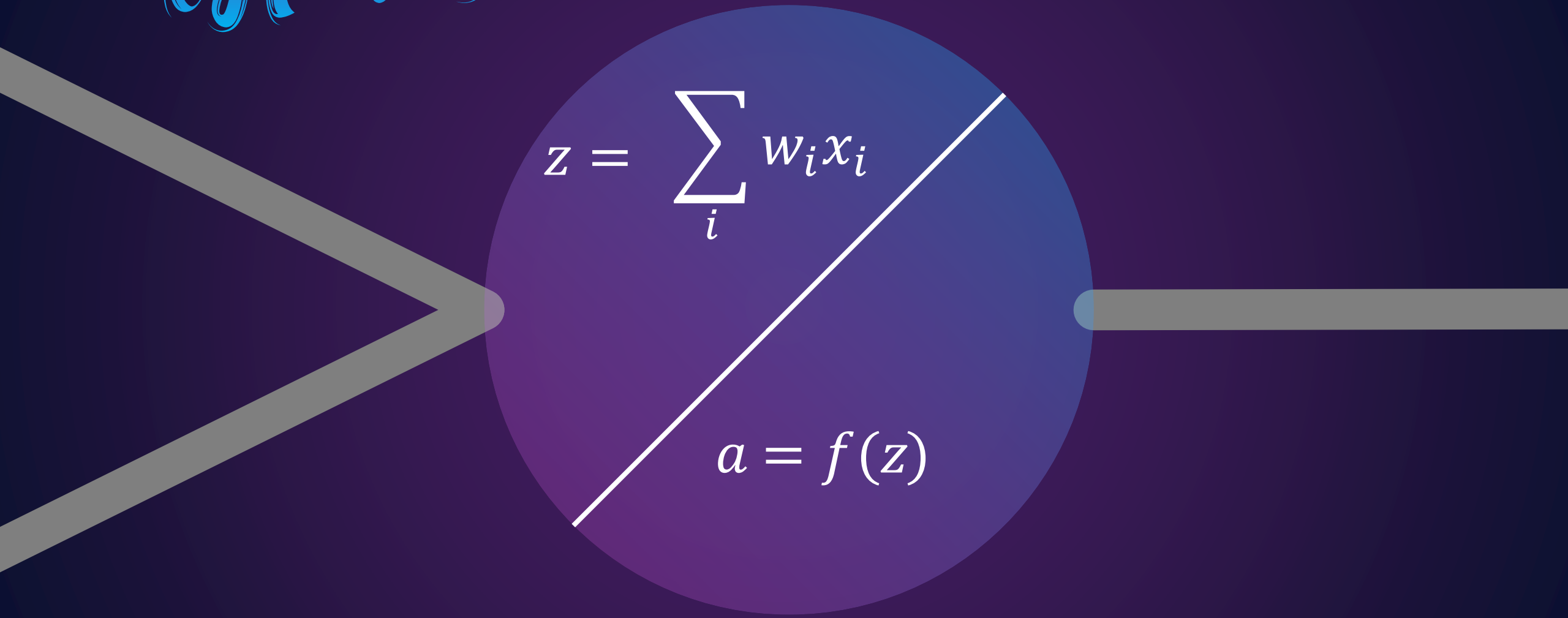
Neuron



Neuron

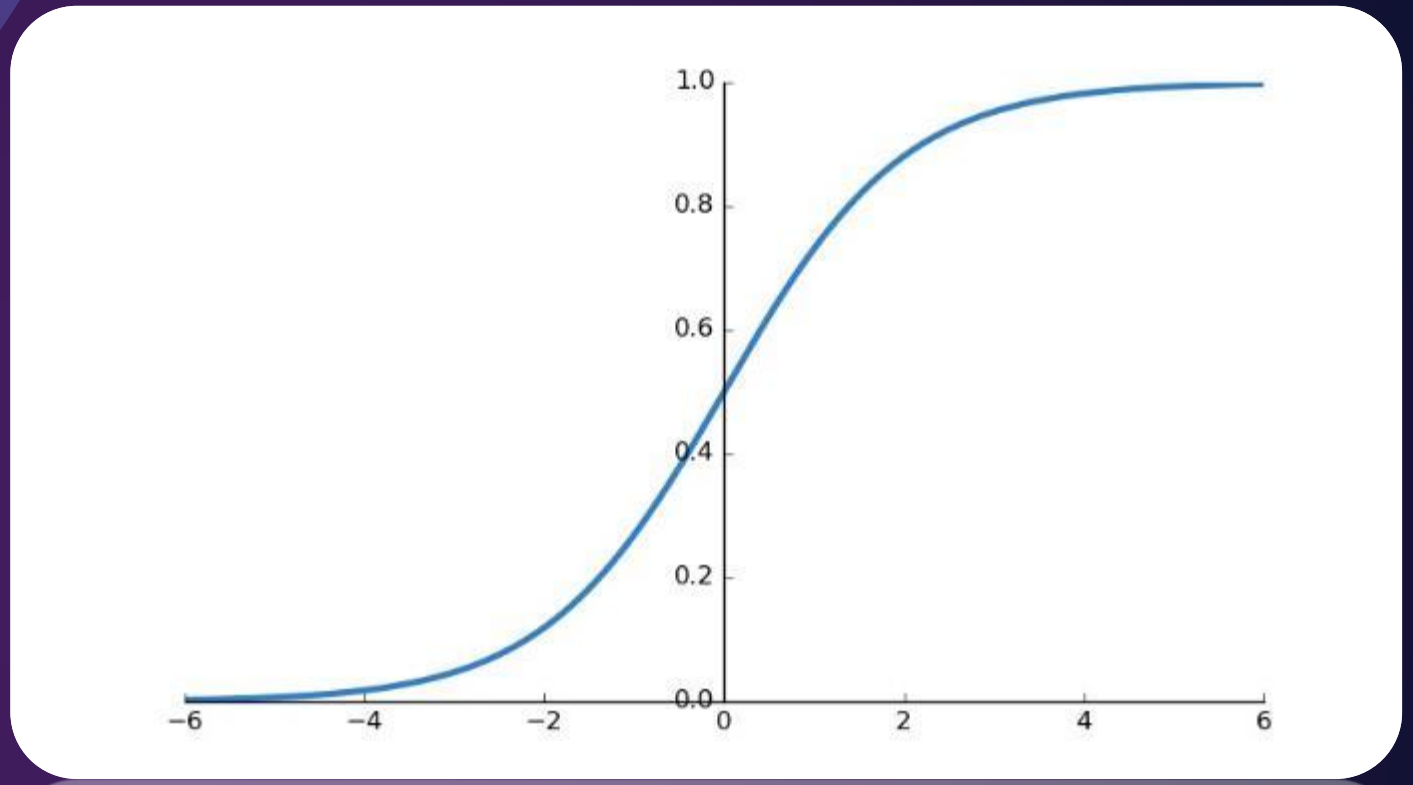


Neuron



$$a = f(z)$$

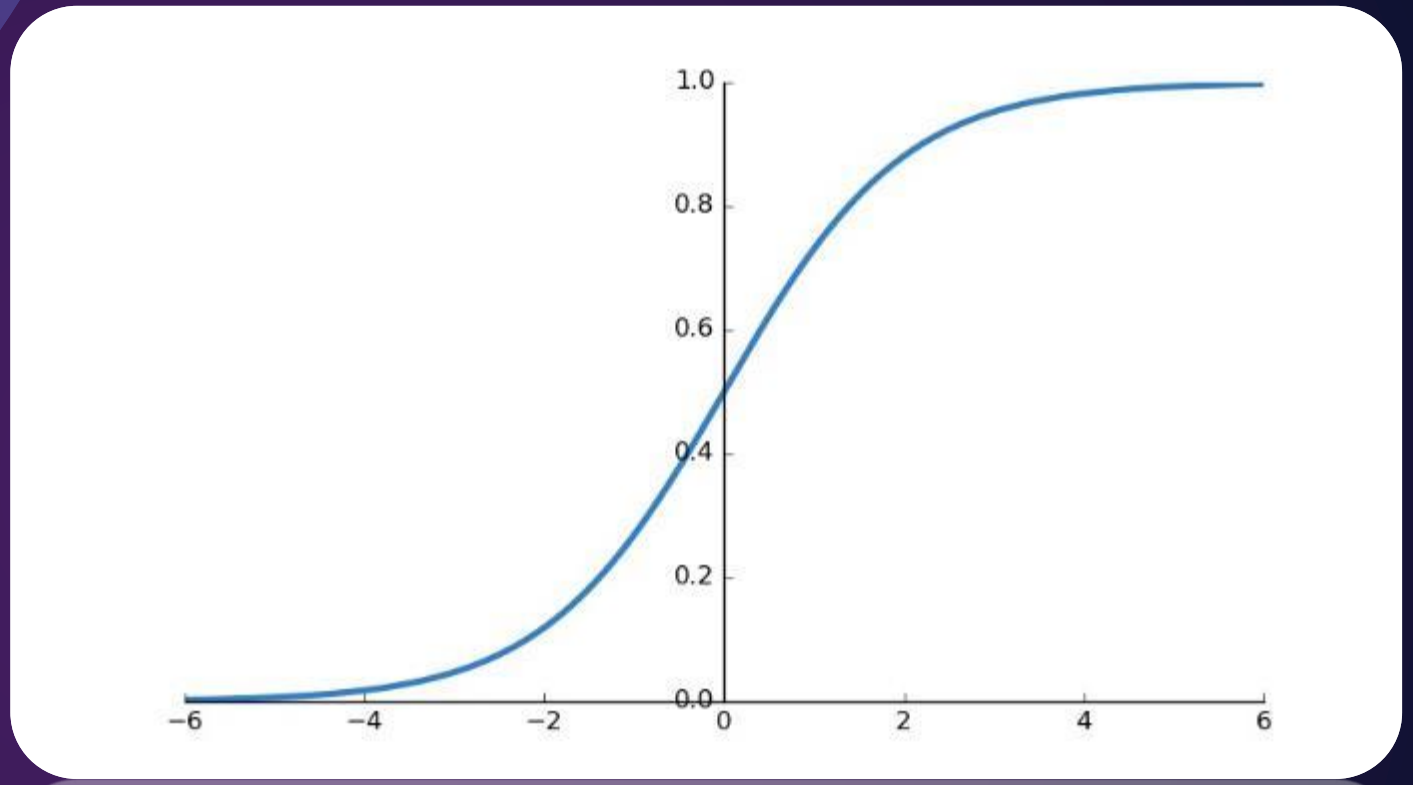
Aktivierungsfunktion



http://ronny.rest/blog/post_2017_08_10_sigmoid/

$$a = f(z)$$

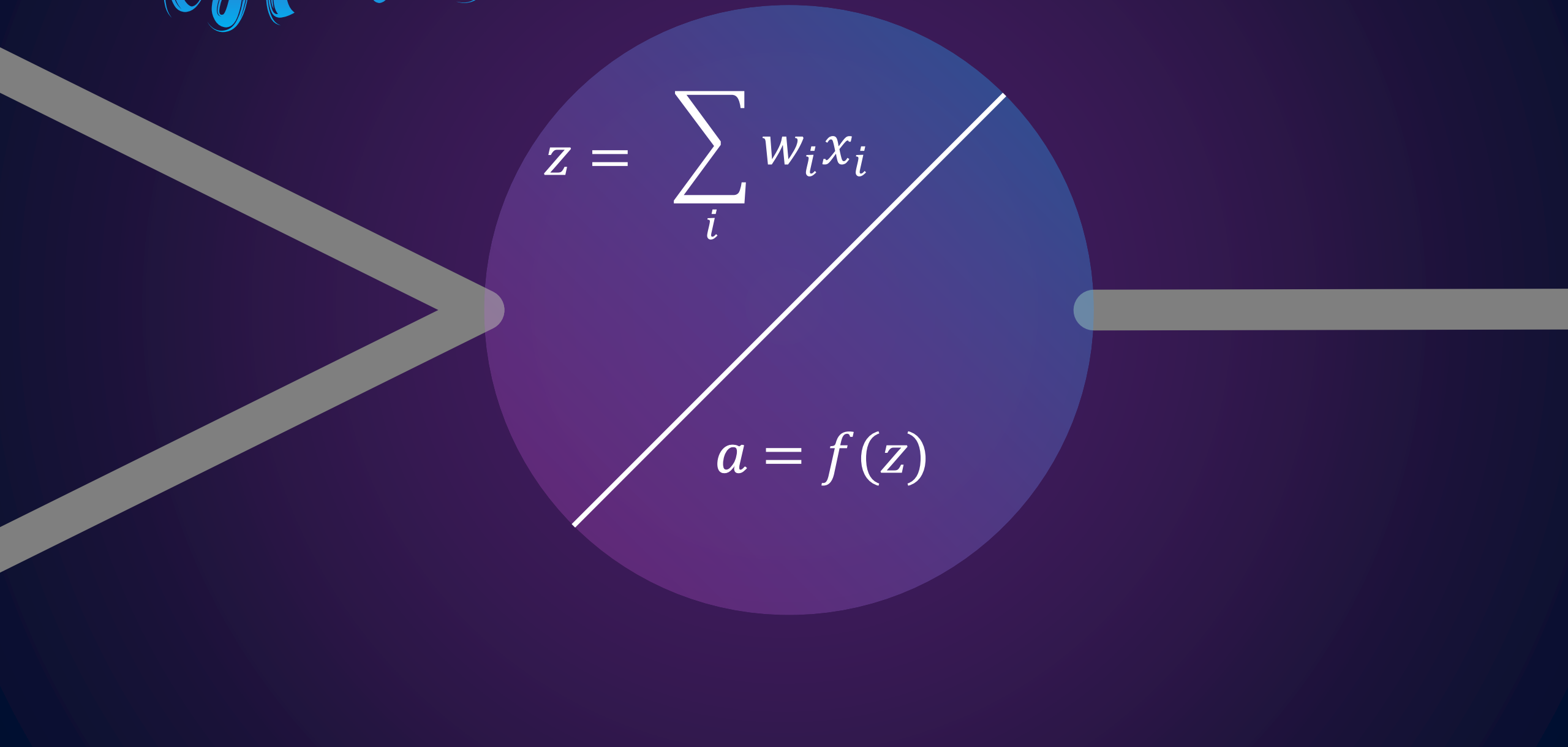
Aktivierungsfunktion



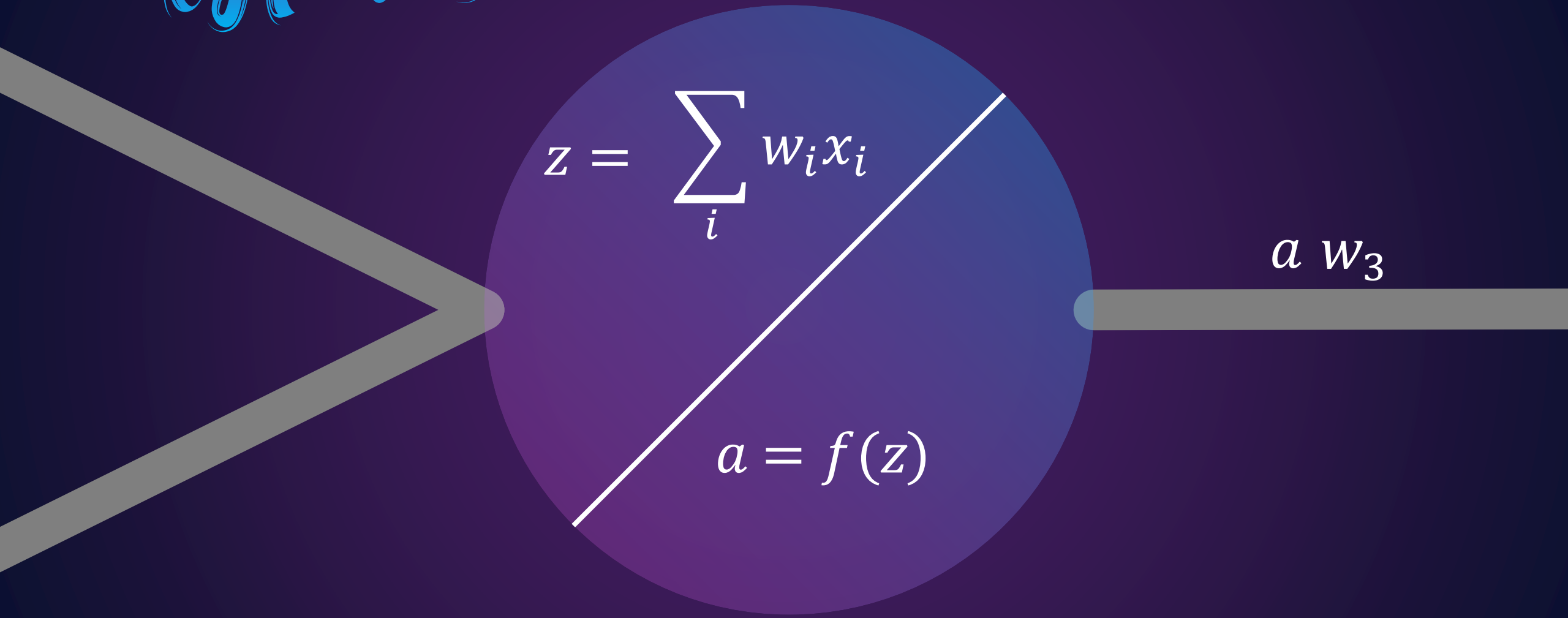
http://ronny.rest/blog/post_2017_08_10_sigmoid/

$$f(x) = \frac{1}{1 + e^{-x}}$$

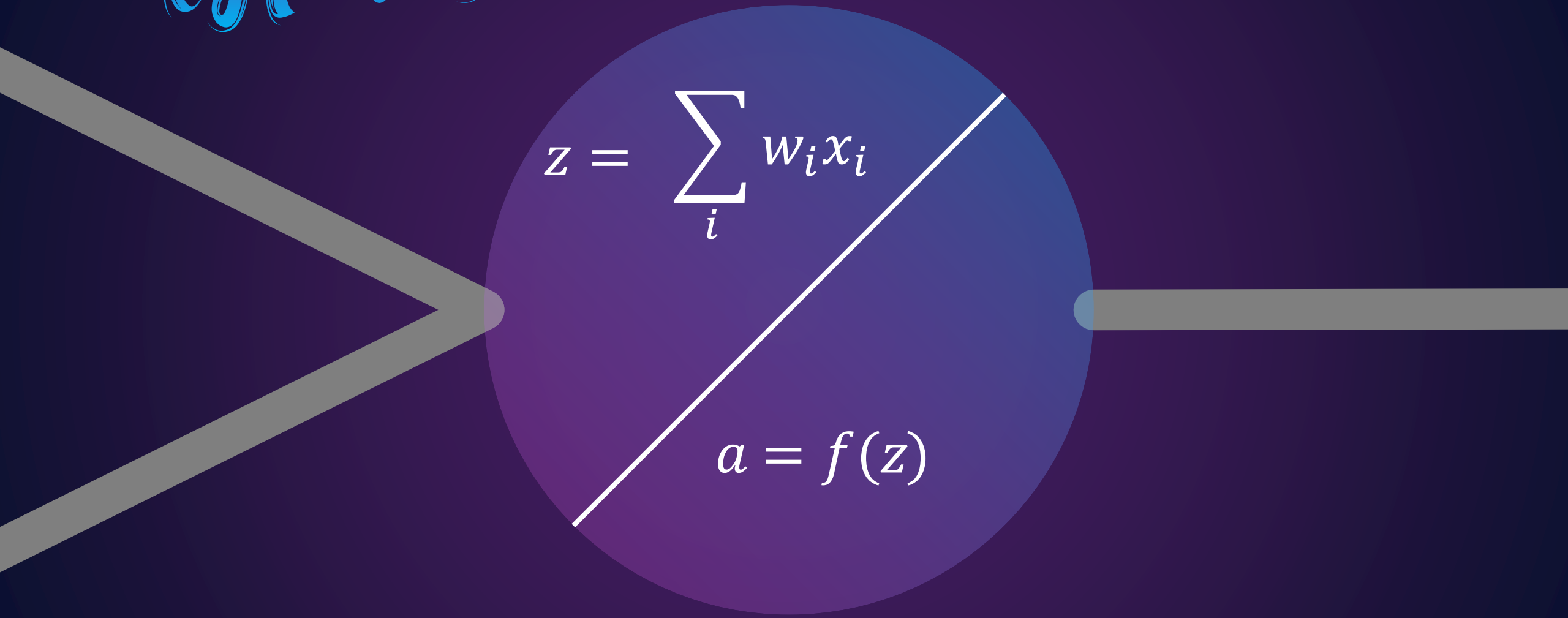
Neuron

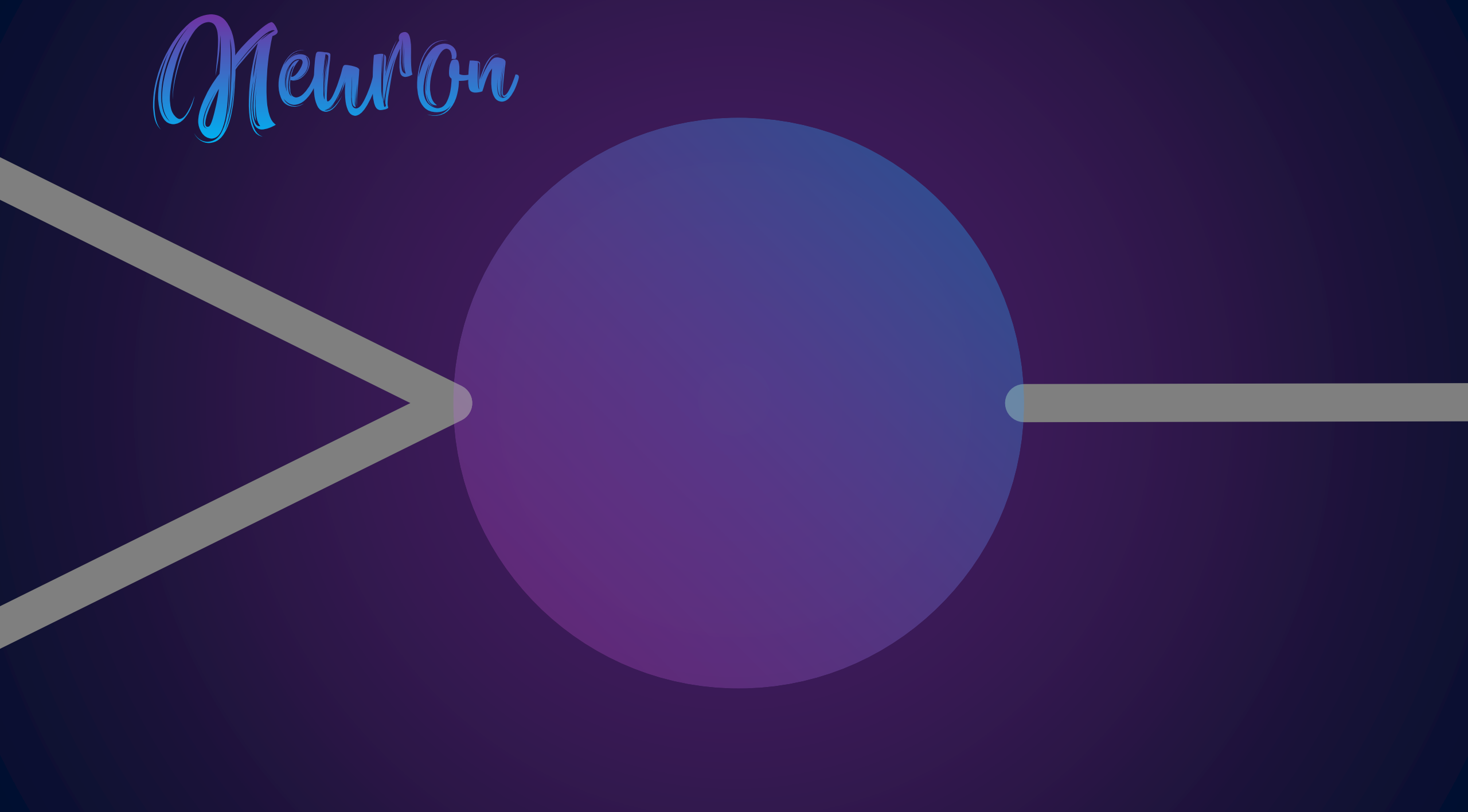


Neuron

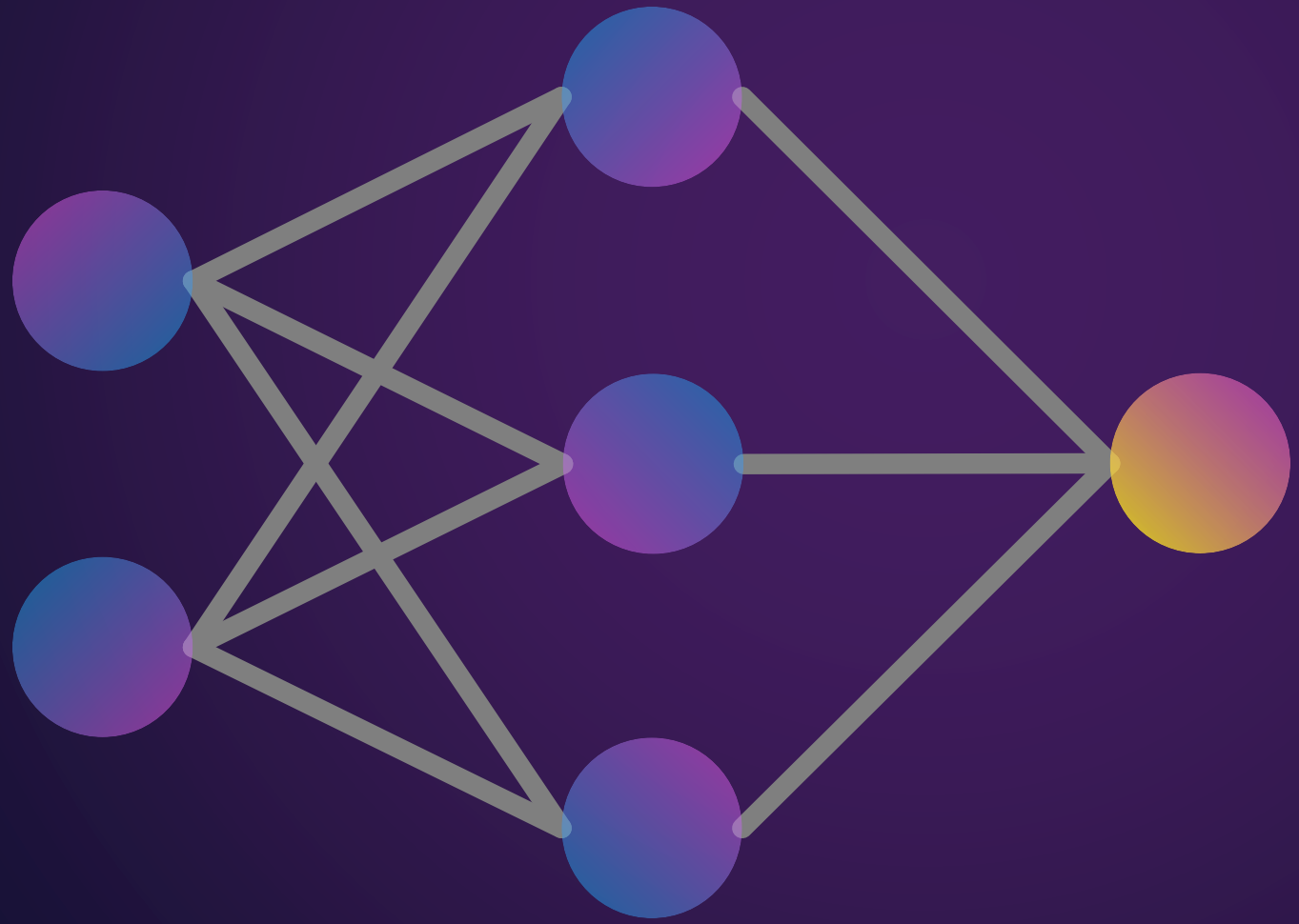


Neuron



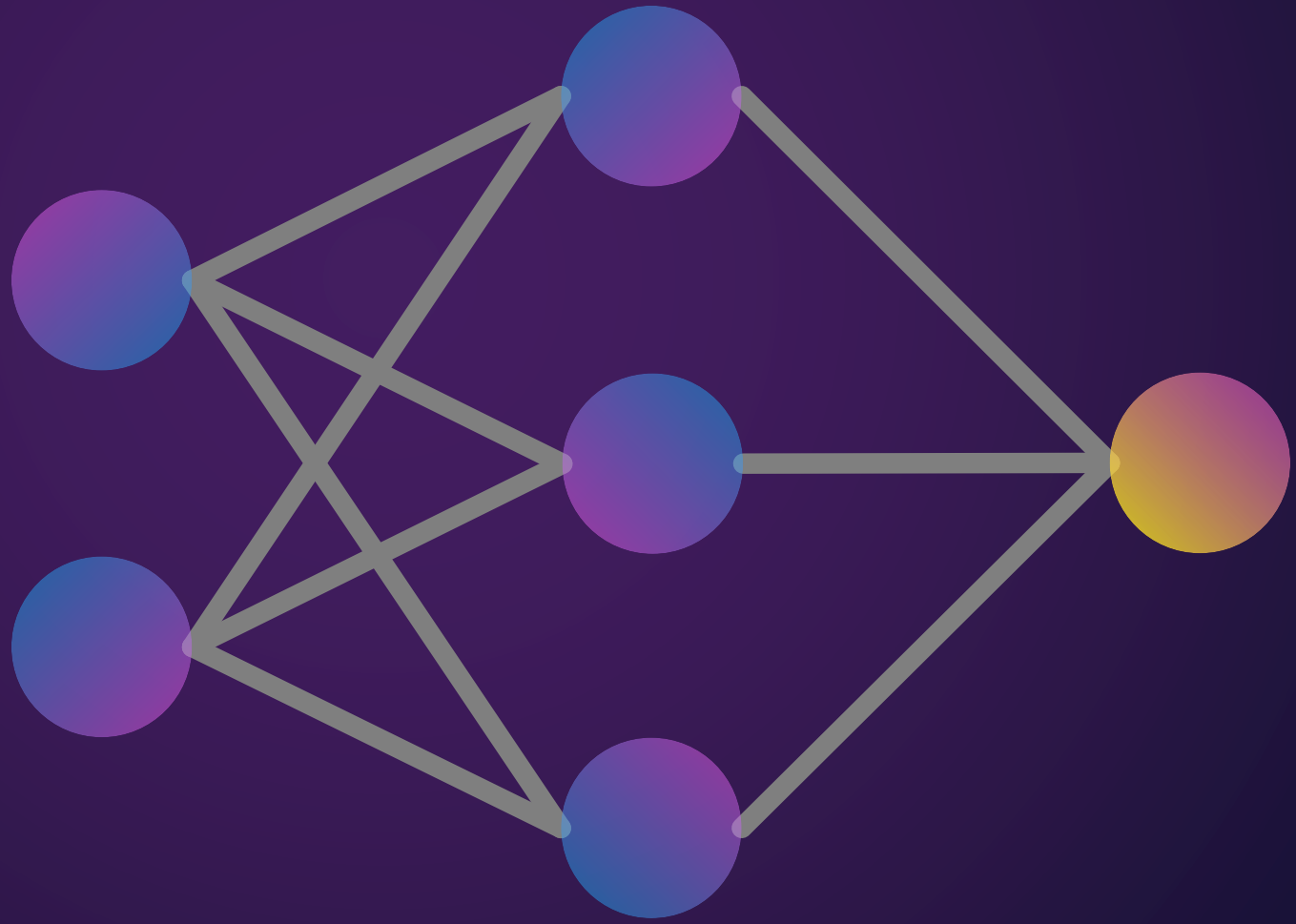

Neuron

Neuronale Netze



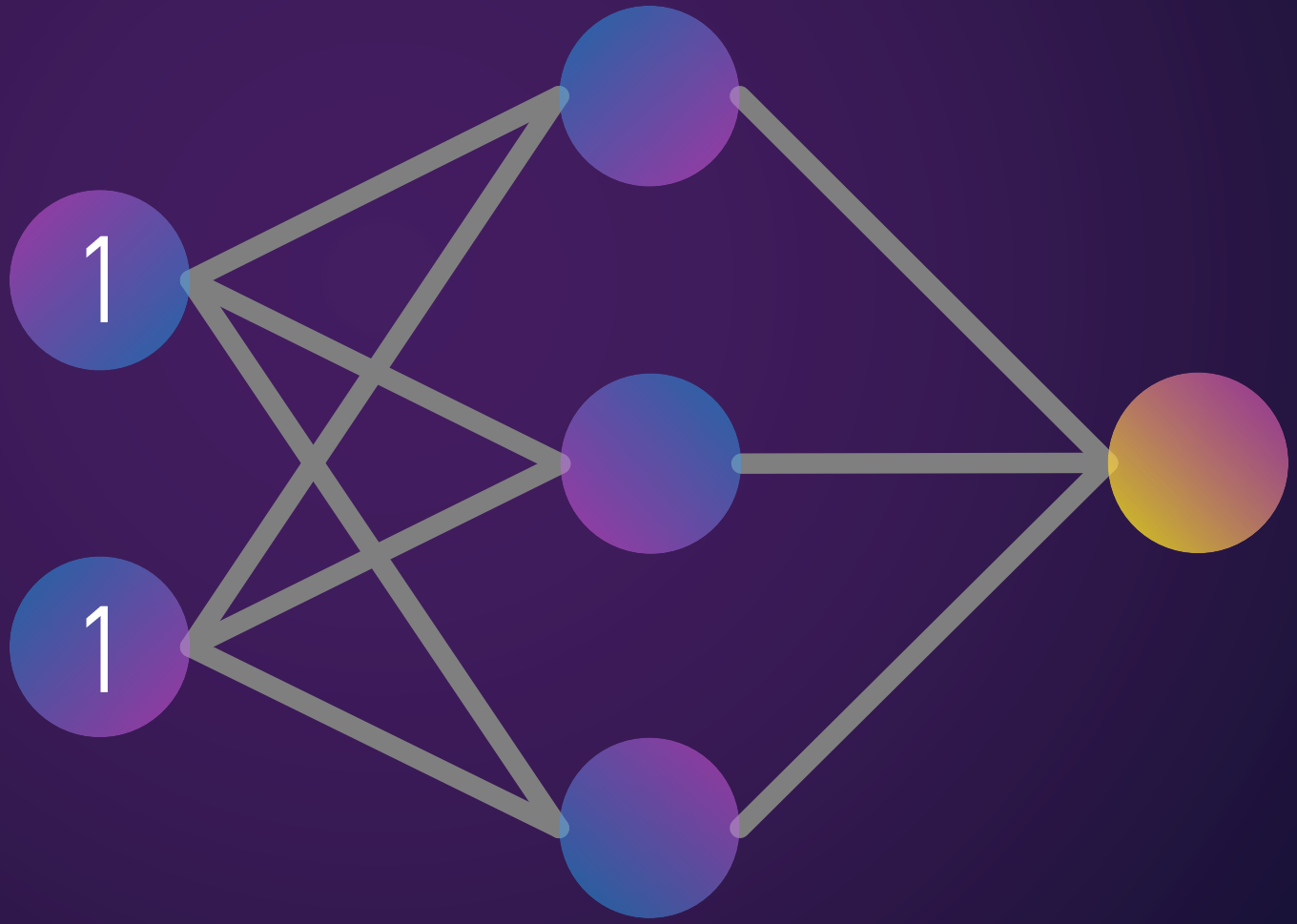
Lernalgorithmus

Supervised
Learning



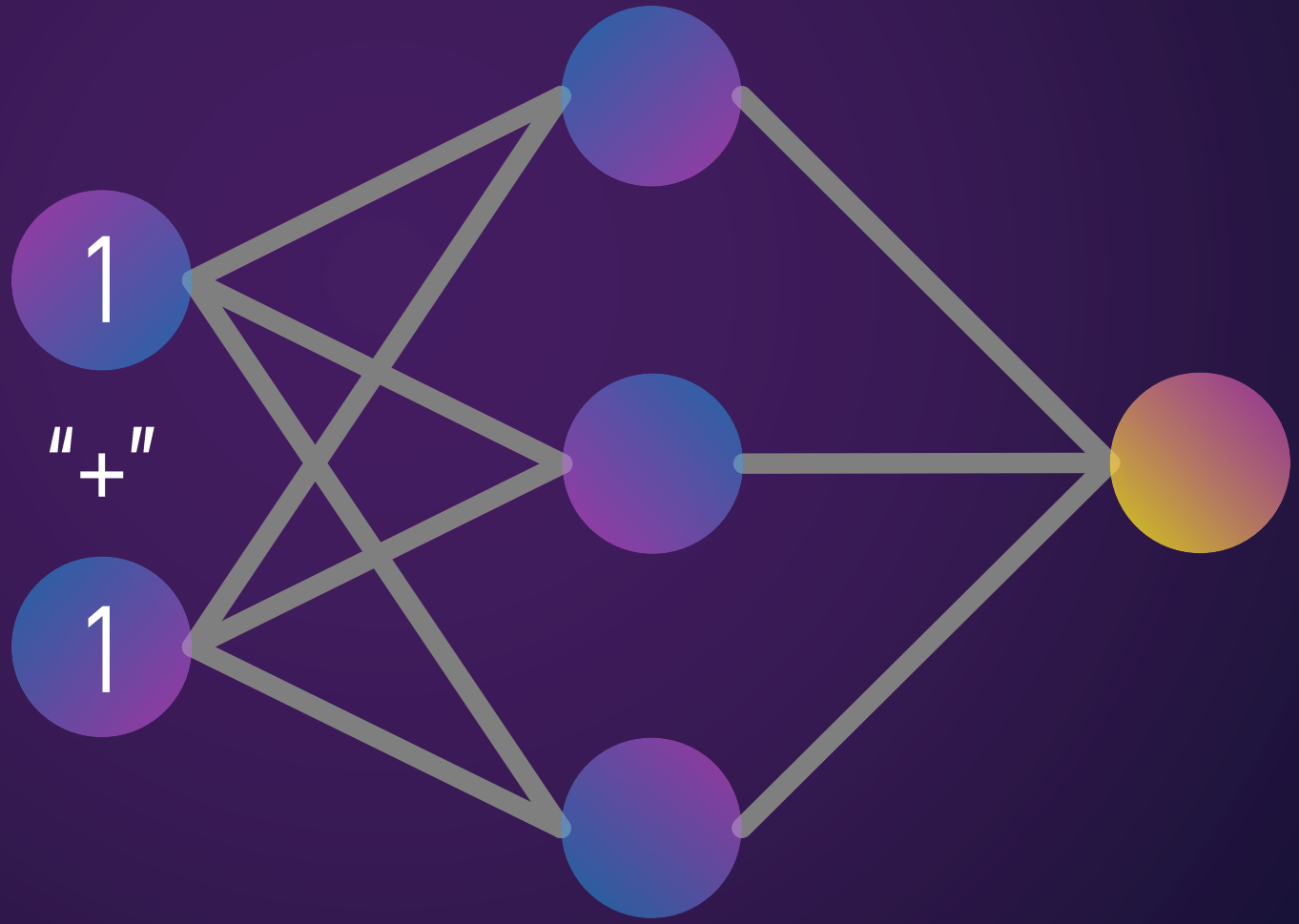
Lernalgorithmus

Supervised
Learning



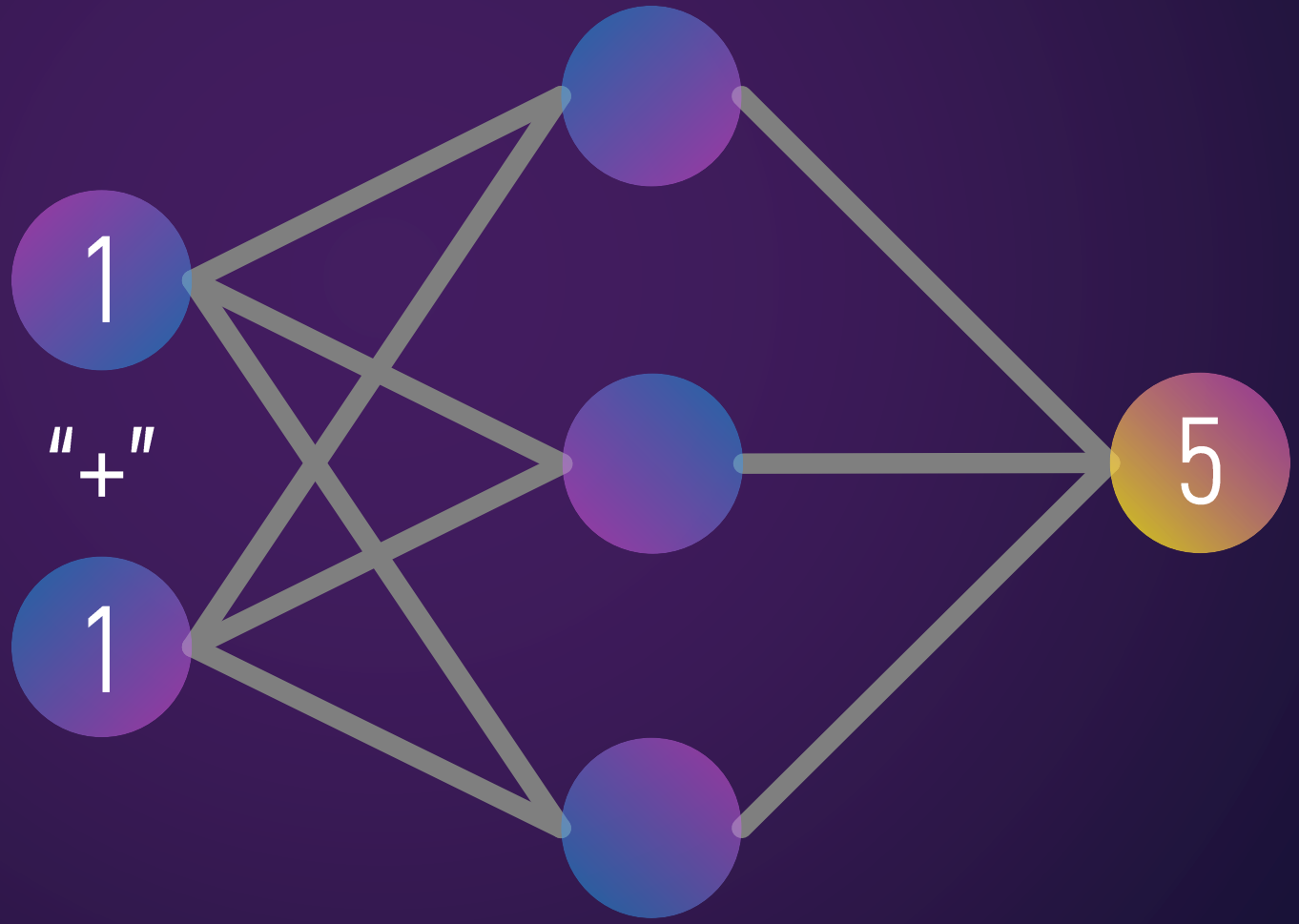
Lernalgorithmus

Supervised
Learning



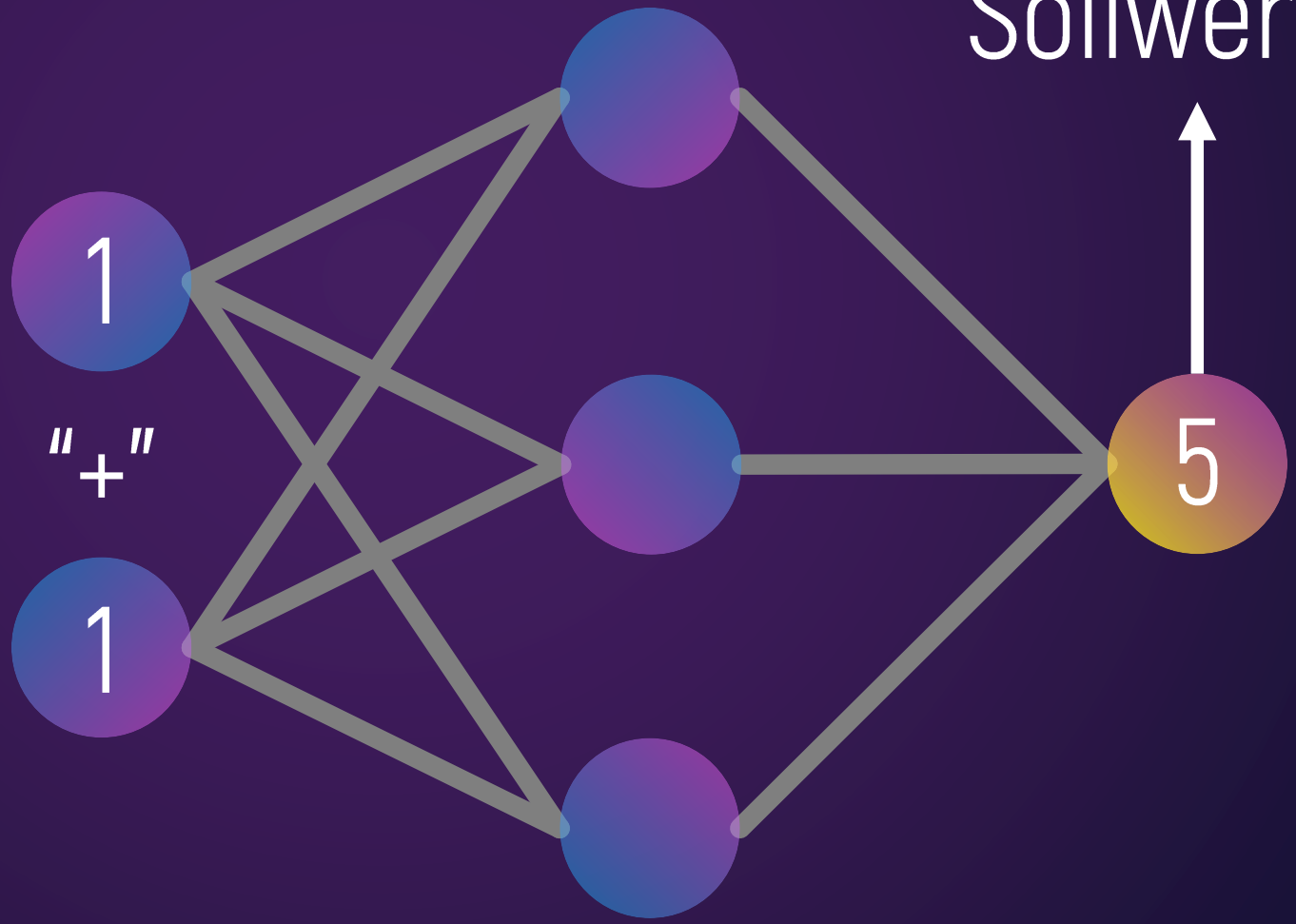
Lernalgorithmus

Supervised
Learning



Lernalgorithmus

Supervised
Learning



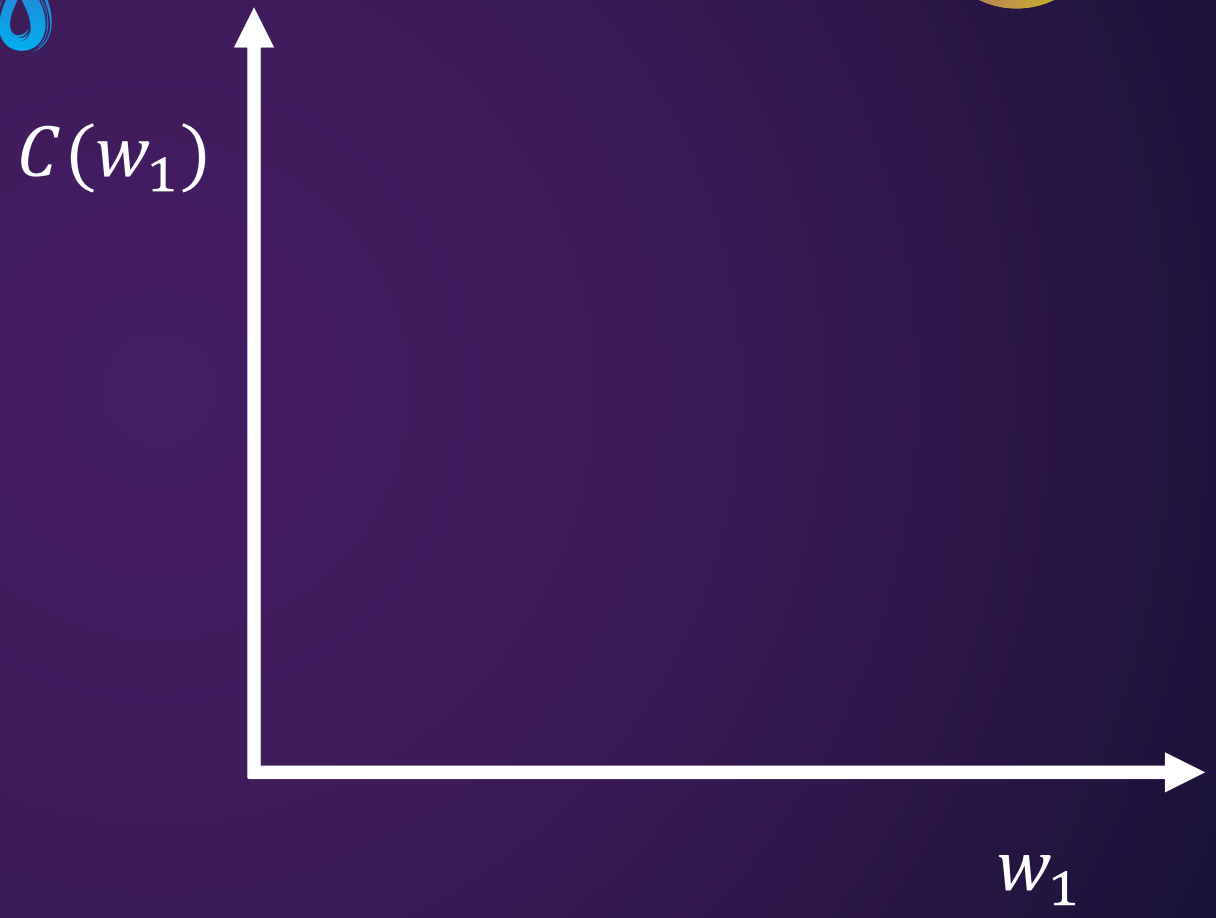
Lernalgorithmus

Gradient Descent

Lernalgorithmus

Gradient

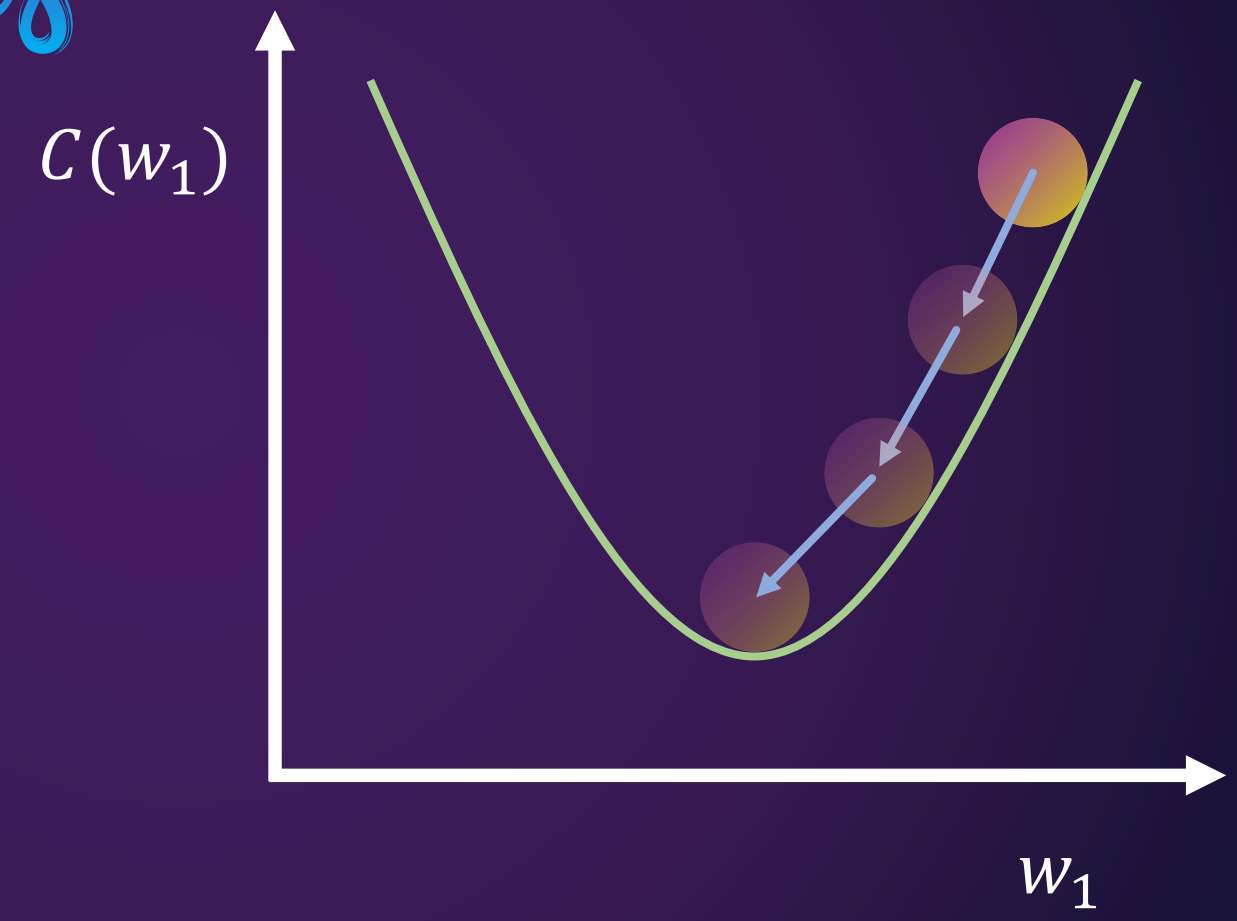
Descent



Lernalgorithmus

Gradient

Descent



Daten


Training
Test

Lernalgorithmus

Hyper-

Parameter

Lernalgorithmus

Hyper-

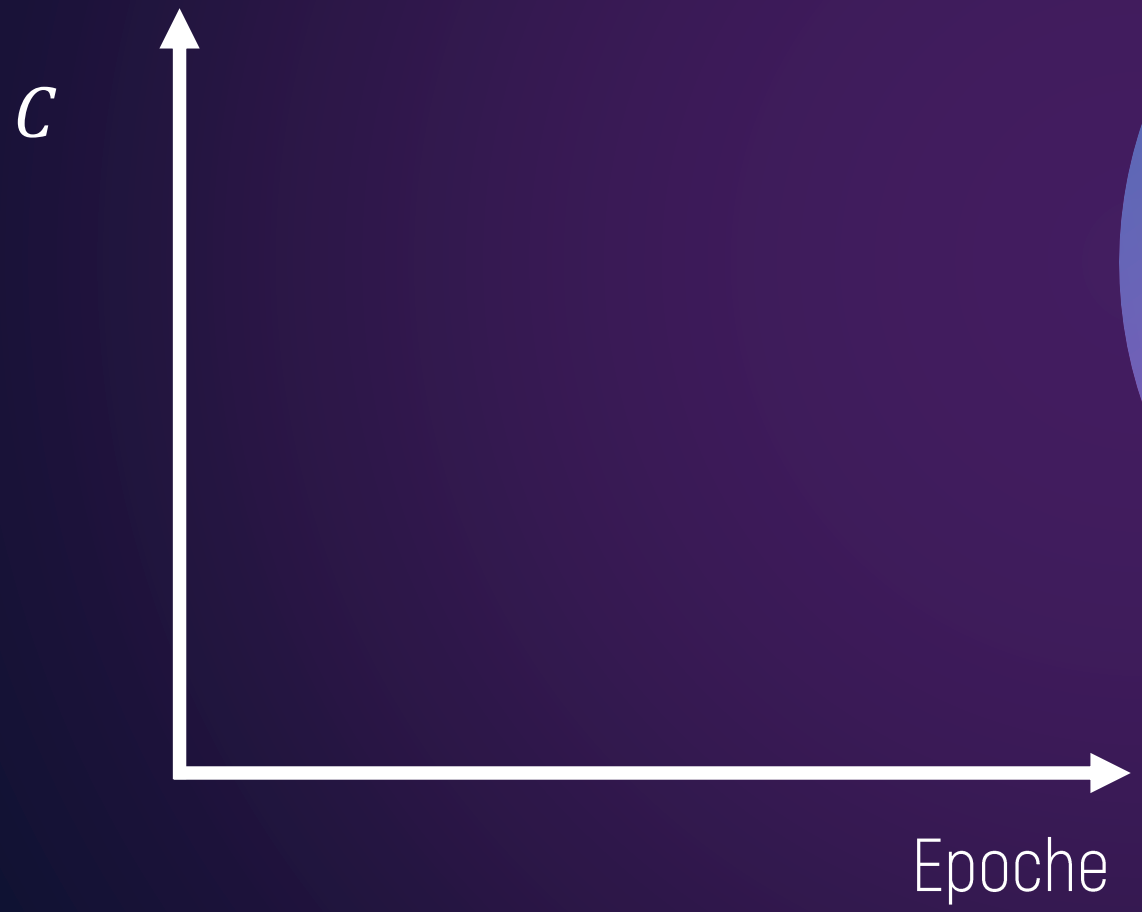
Parameter

Lernalgorithmus

Hyper-
Parameter

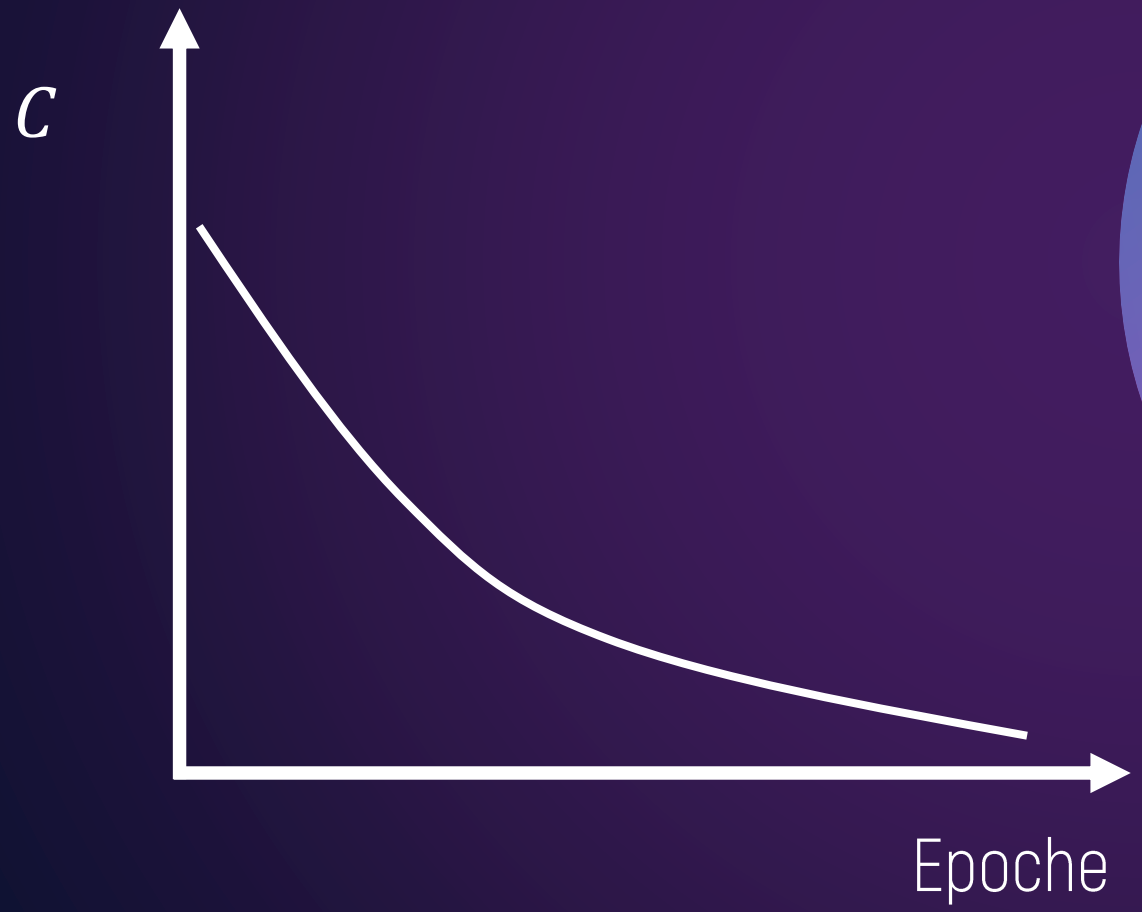
Epochen

Lernalgorithmus



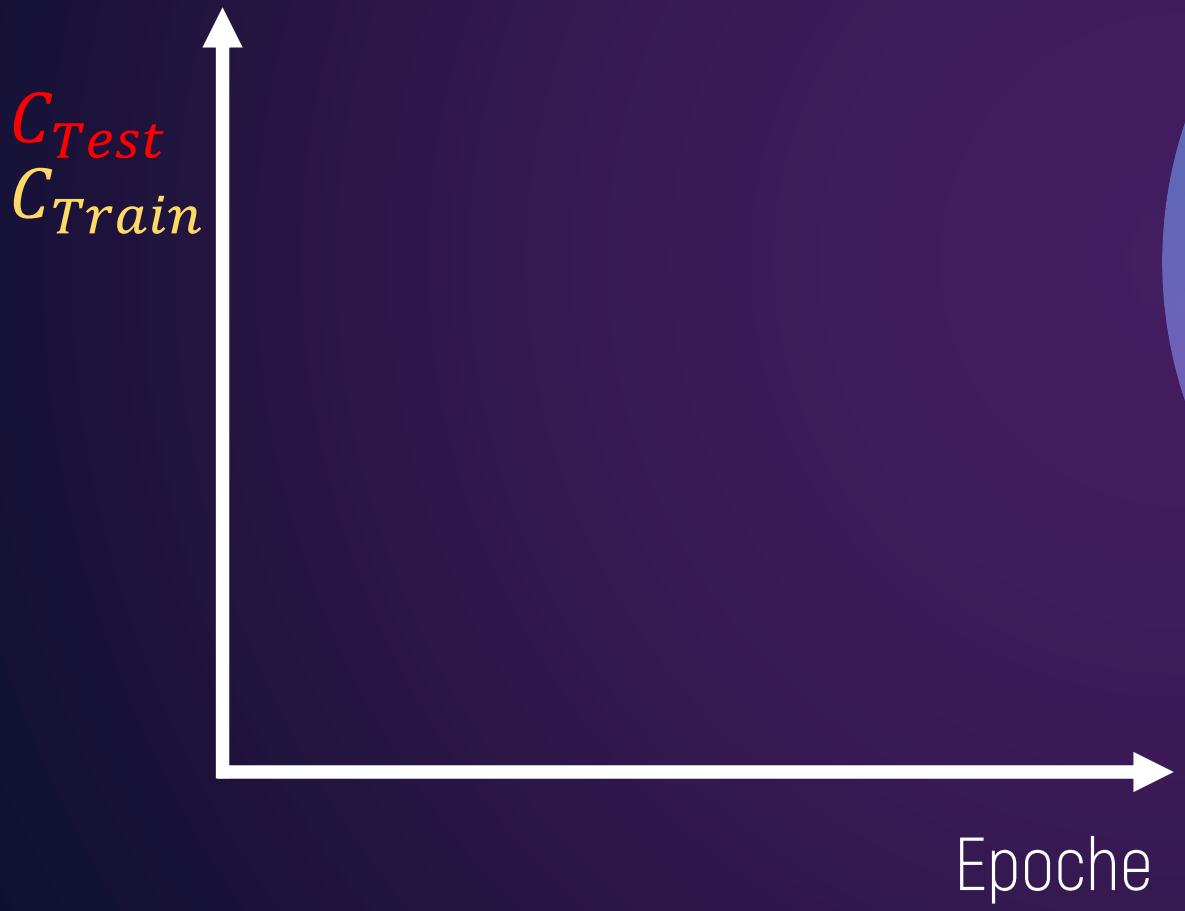
Epochen

Lernalgorithmus



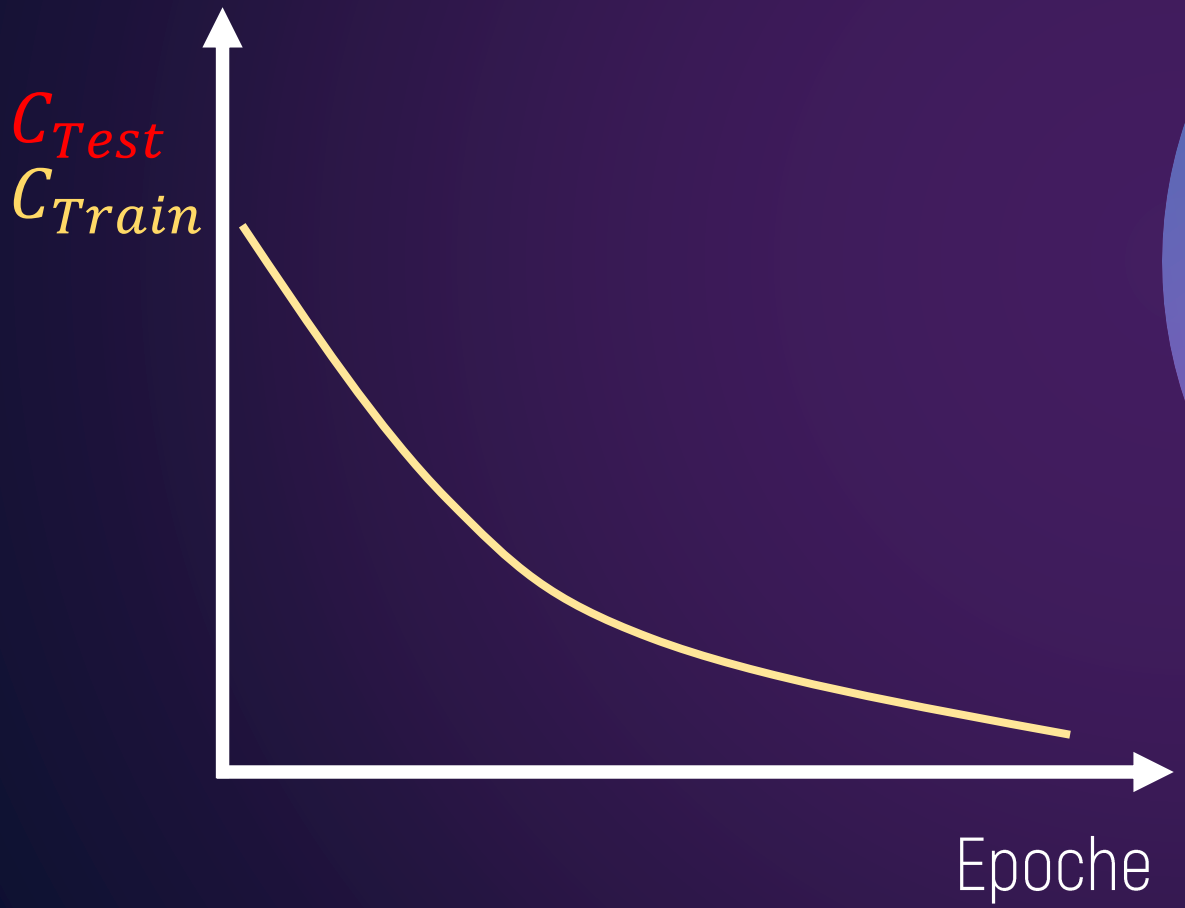
Epochen

Lernalgorithmus



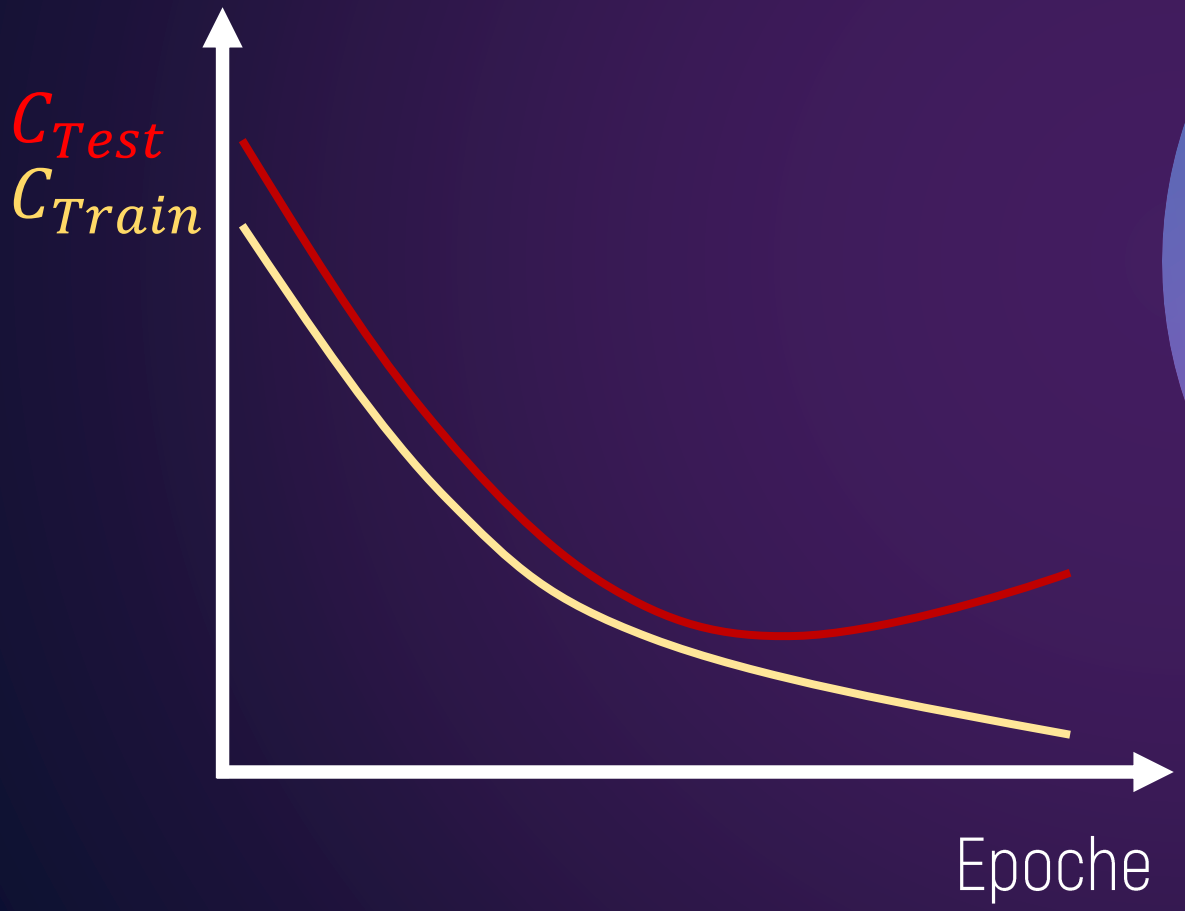
Epochen

Lernalgorithmus



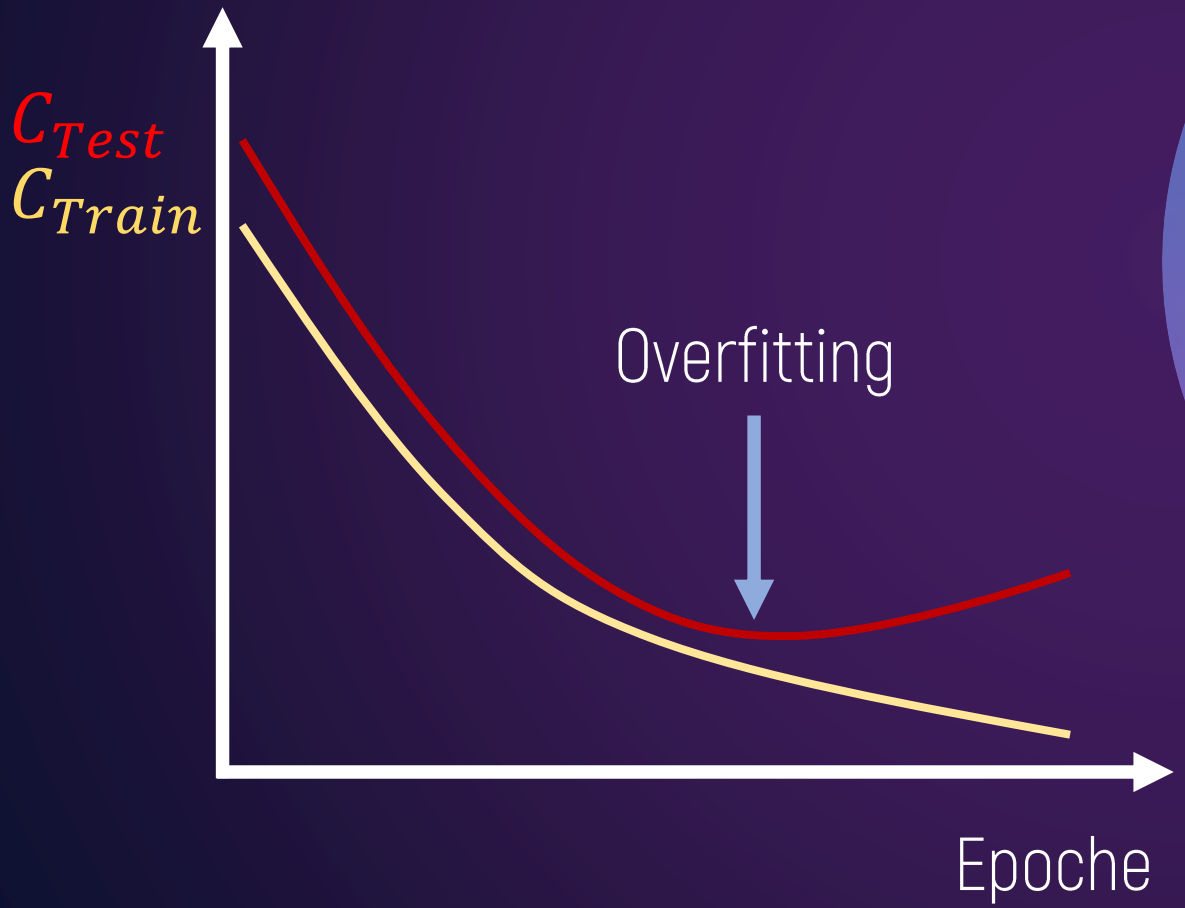
Epochen

Lernalgorithmus



Epochen

Lernalgorithmus



Epochen

Lernalgorithmus

Hyper-
Parameter

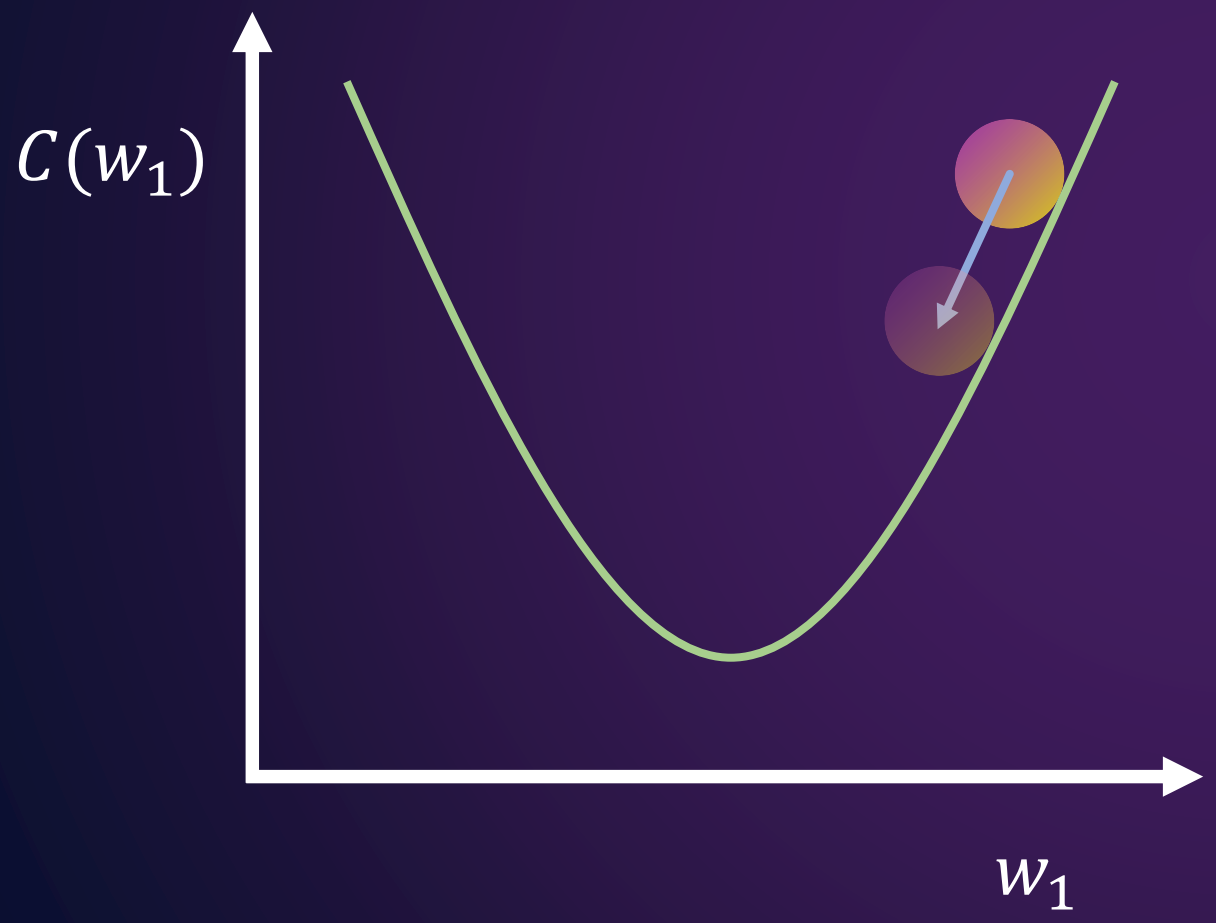
Epochen

Lernalgorithmus

Hyper-
Parameter

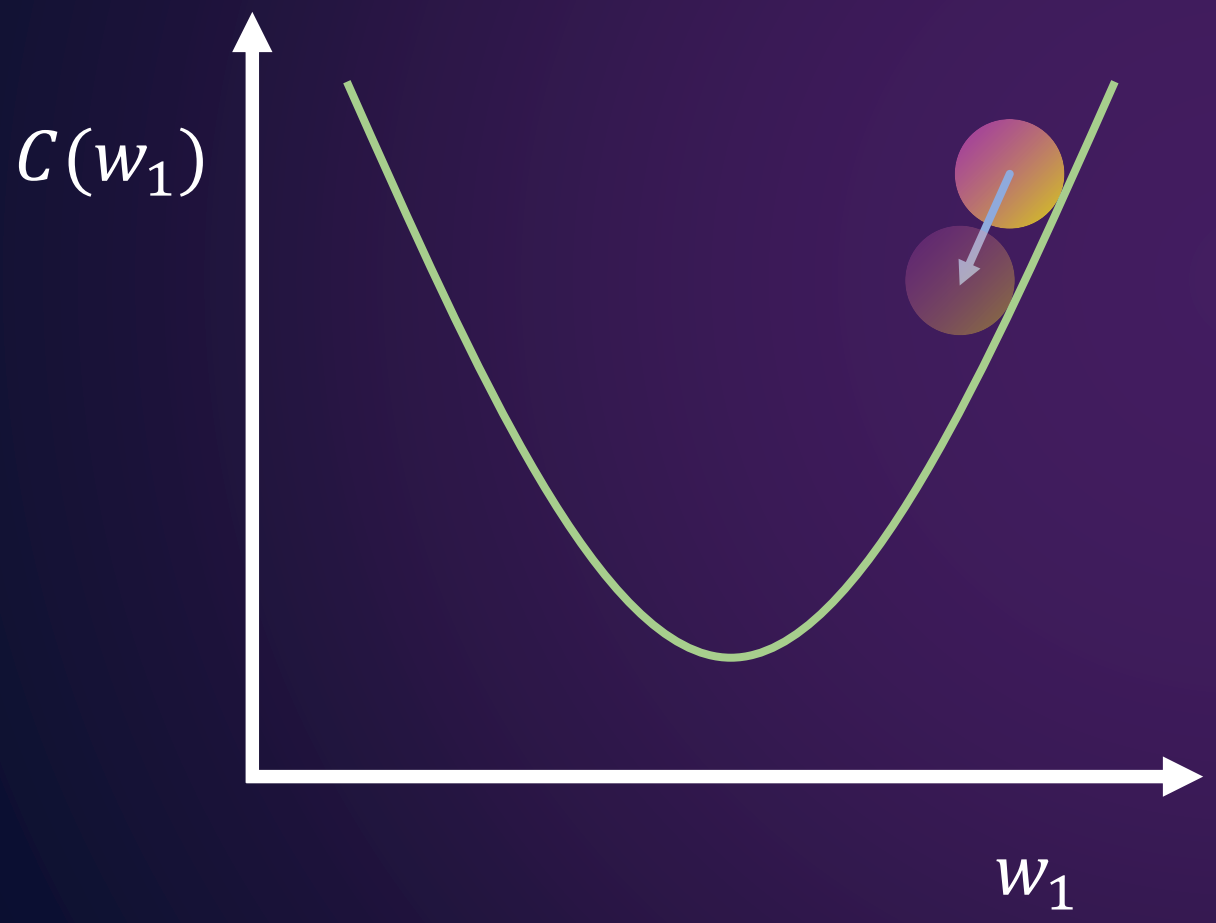
Lernrate

Lernalgorithmus



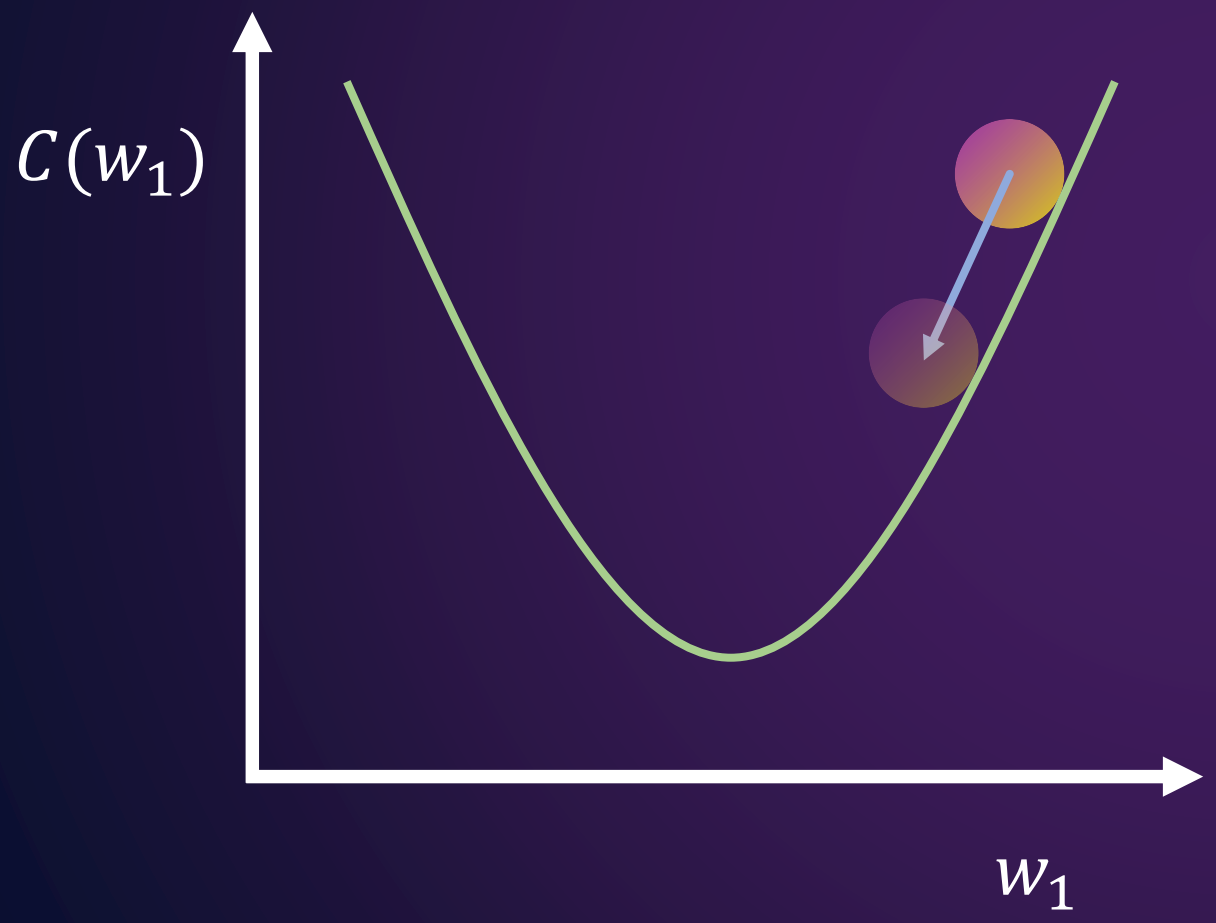
Lernrate

Lernalgorithmus



Lernrate

Lernalgorithmus



Lernrate

Lernalgorithmus

Hyper-
Parameter

Lernrate

Lernalgorithmus

Hyper-
Parameter

Batch Size

Lernalgorithmus

Batch Size

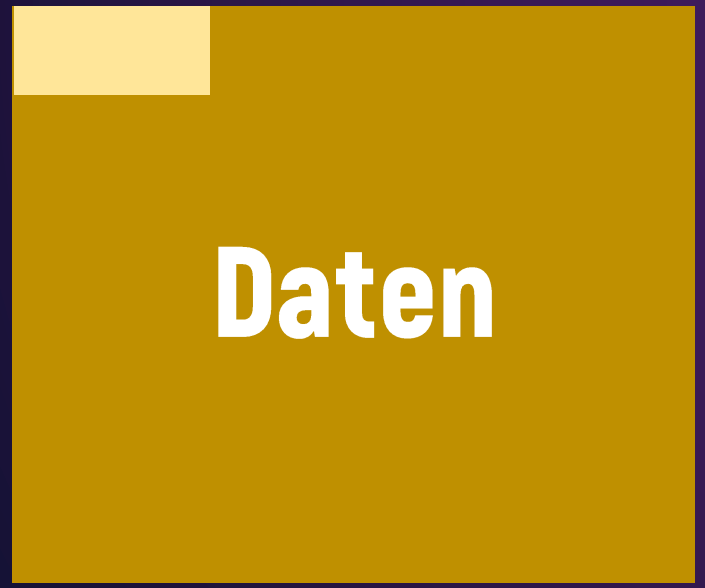
Lernalgorithmus

Daten

Batch Size

Lernalgorithmus

Batch



Batch Size

Lernalgorithmus

Hyper-
Parameter

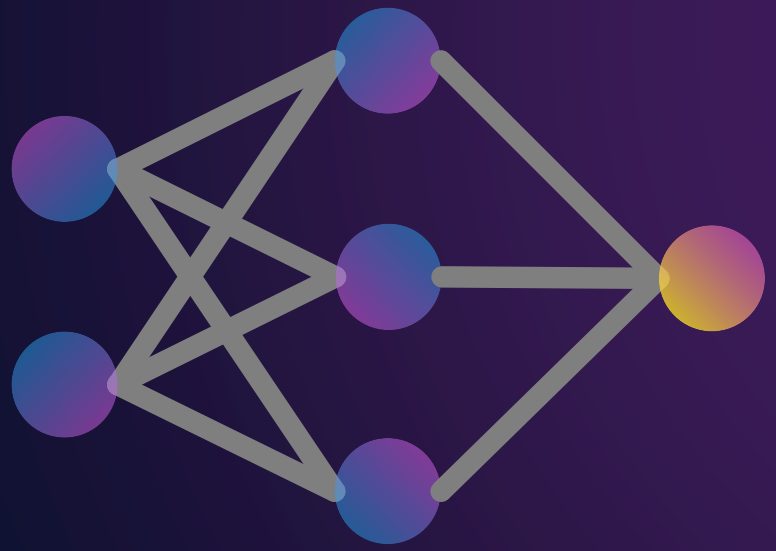
Batch Size

Lernalgorithmus

Hyper-
Parameter

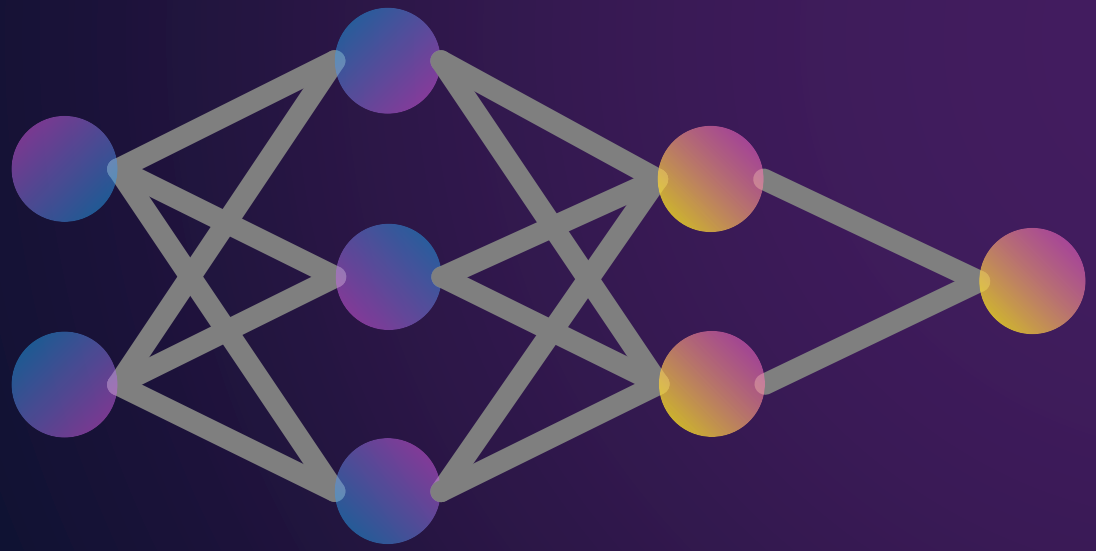
Anzahl der Schichten

Lernalgorithmus



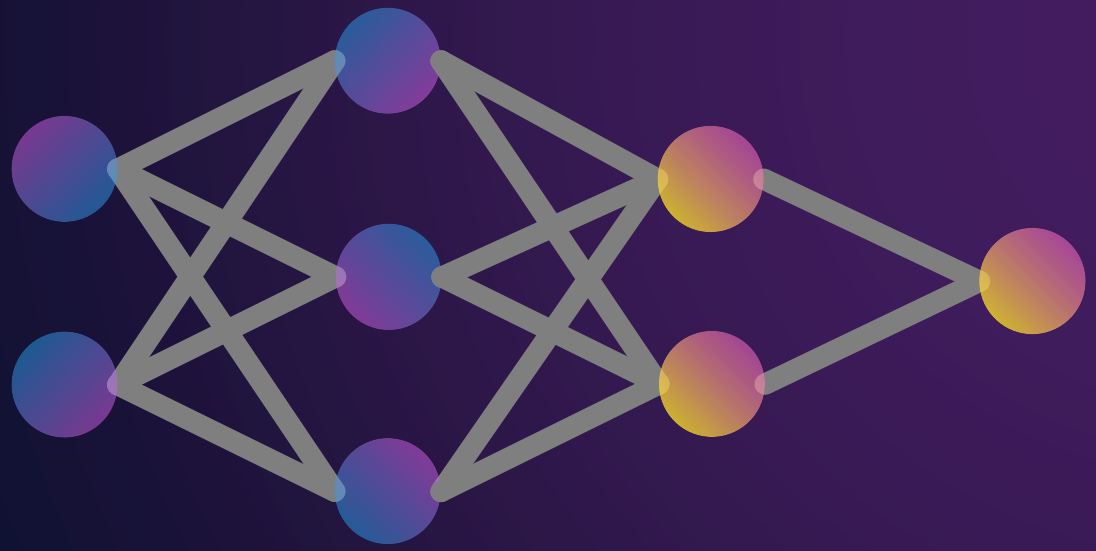
Anzahl der Schichten

Lernalgorithmus



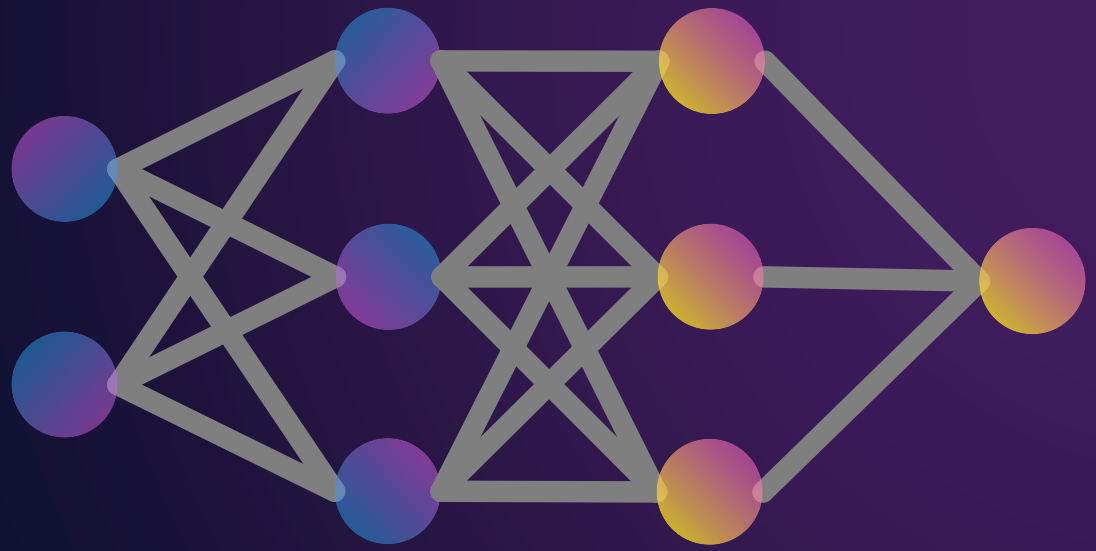
Anzahl der Schichten

Lernalgorithmus



Anzahl der Neuronen

Lernalgorithmus



Anzahl der Neuronen

Lernalgorithmus

Hyper-
Parameter

Anzahl der Neuronen

Lernalgorithmus

Hyper-

Parameter


Tensor
Flow


Tensor


Flow

Was ist ein
Tensor

Was ist ein Tensor

Eine Zahl

$$s = a$$

Was ist ein Tensor

Ein Vektor

$$\vec{v} = \begin{bmatrix} a \\ b \\ c \end{bmatrix}$$

Was ist ein Tensor

Eine Matrix

$$M = \begin{bmatrix} a & b & c \\ d & e & f \\ g & h & i \end{bmatrix}$$

Was ist ein

Tensor

N-Dimensional

$$T = \begin{bmatrix} a & b & c \\ d & e & f \\ g & h & i \end{bmatrix}$$


Tensor
Flow


Tensor
Flow

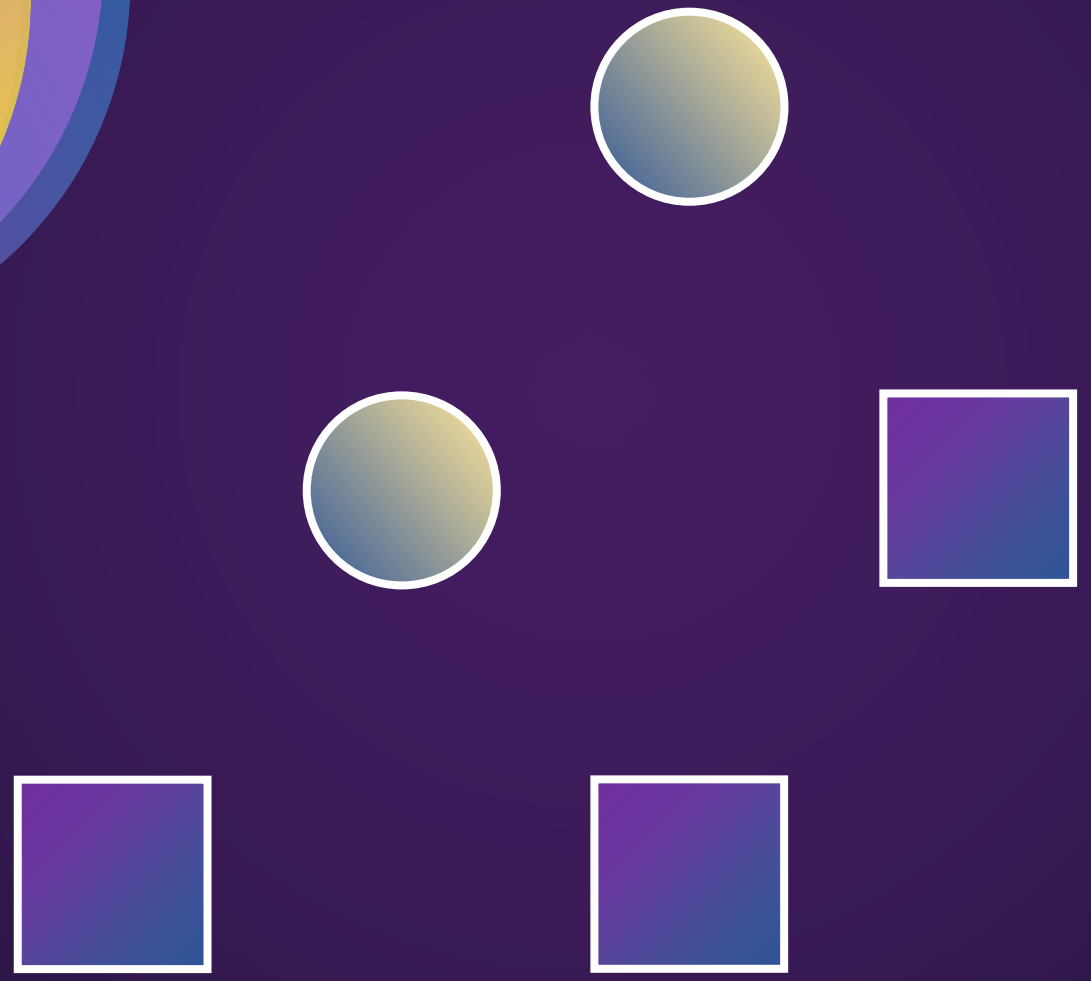
Wie fließen Tensor

Wie fließen
Tensor

Tensor

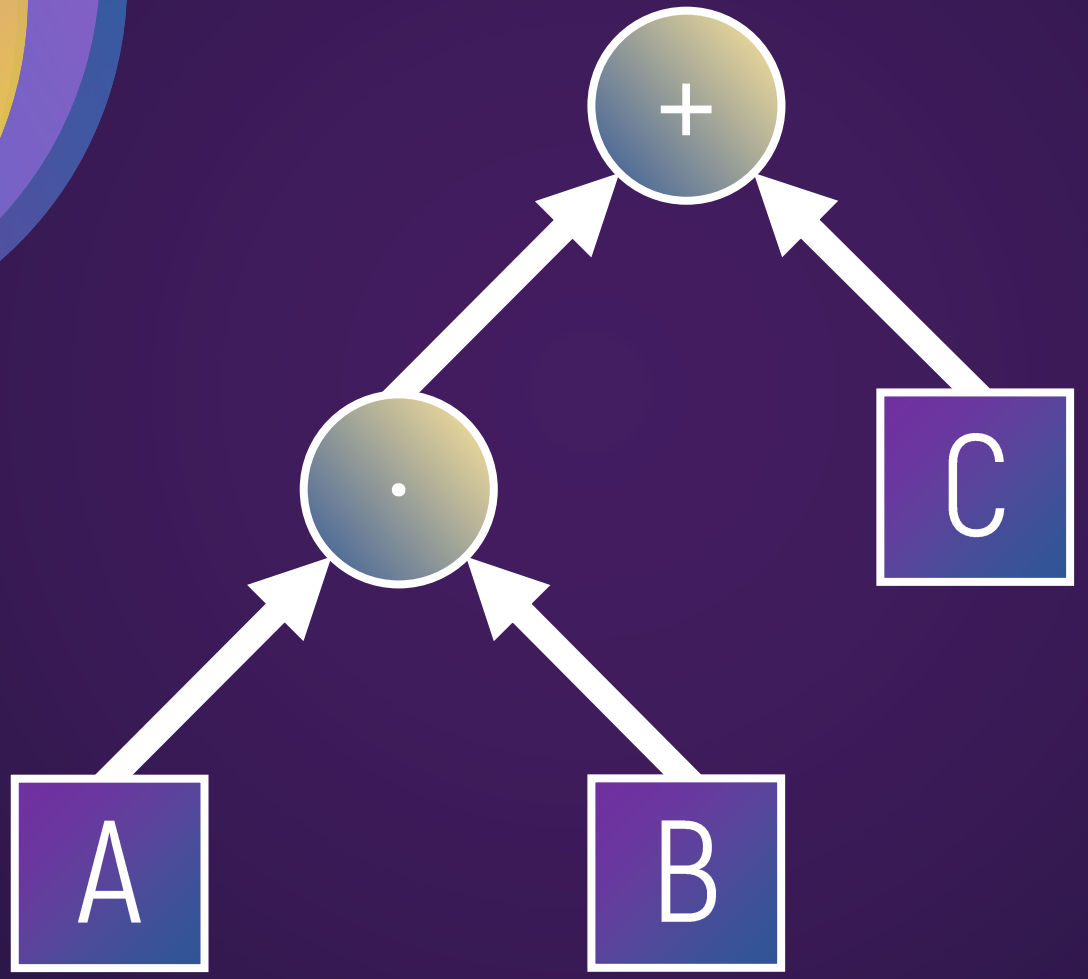
Operation

Wie fließen Tensor



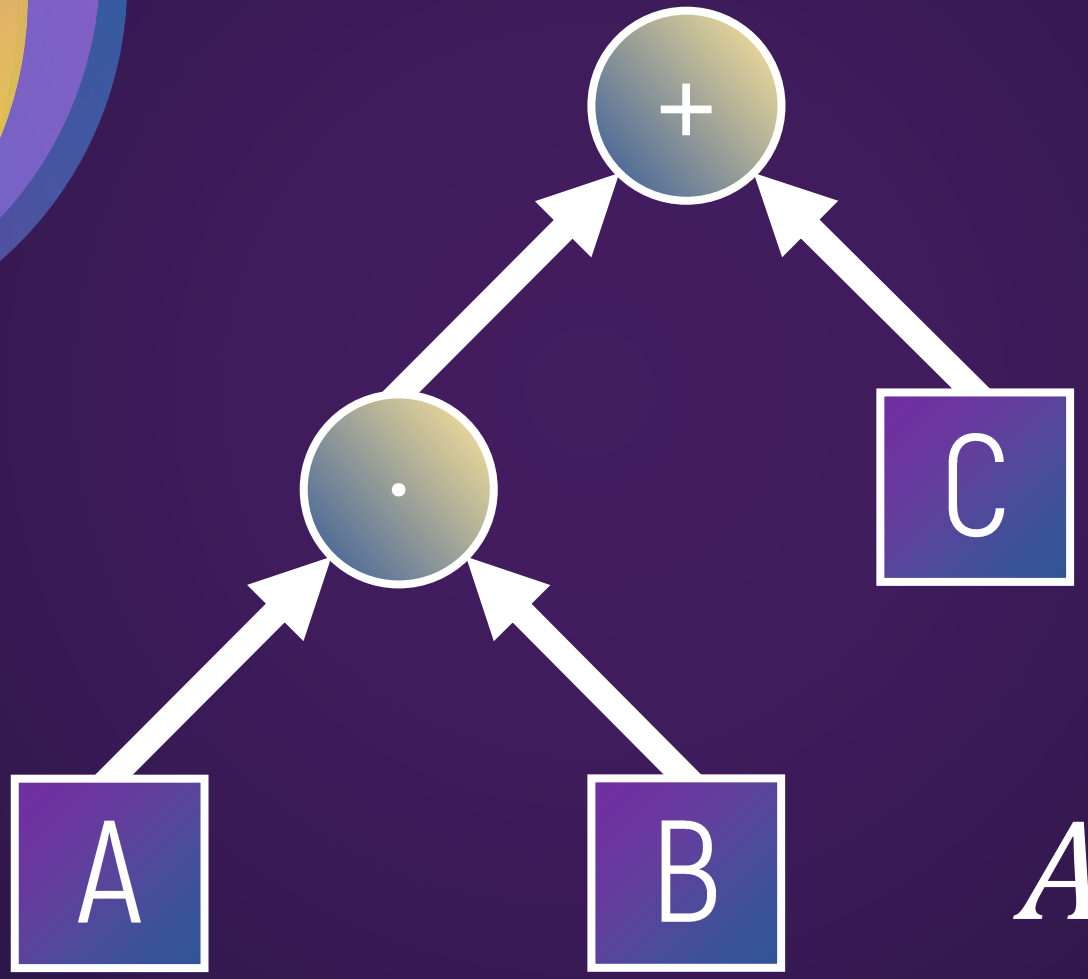
Flow

Wie fließen Tensor



Flow

Wie fließen Tensor



$$A \cdot B + C$$

Was

Tensoren

Wie

Operationen

Was

Tensoren

Wie

Operationen

Keras


TensOr
Flow


Keras

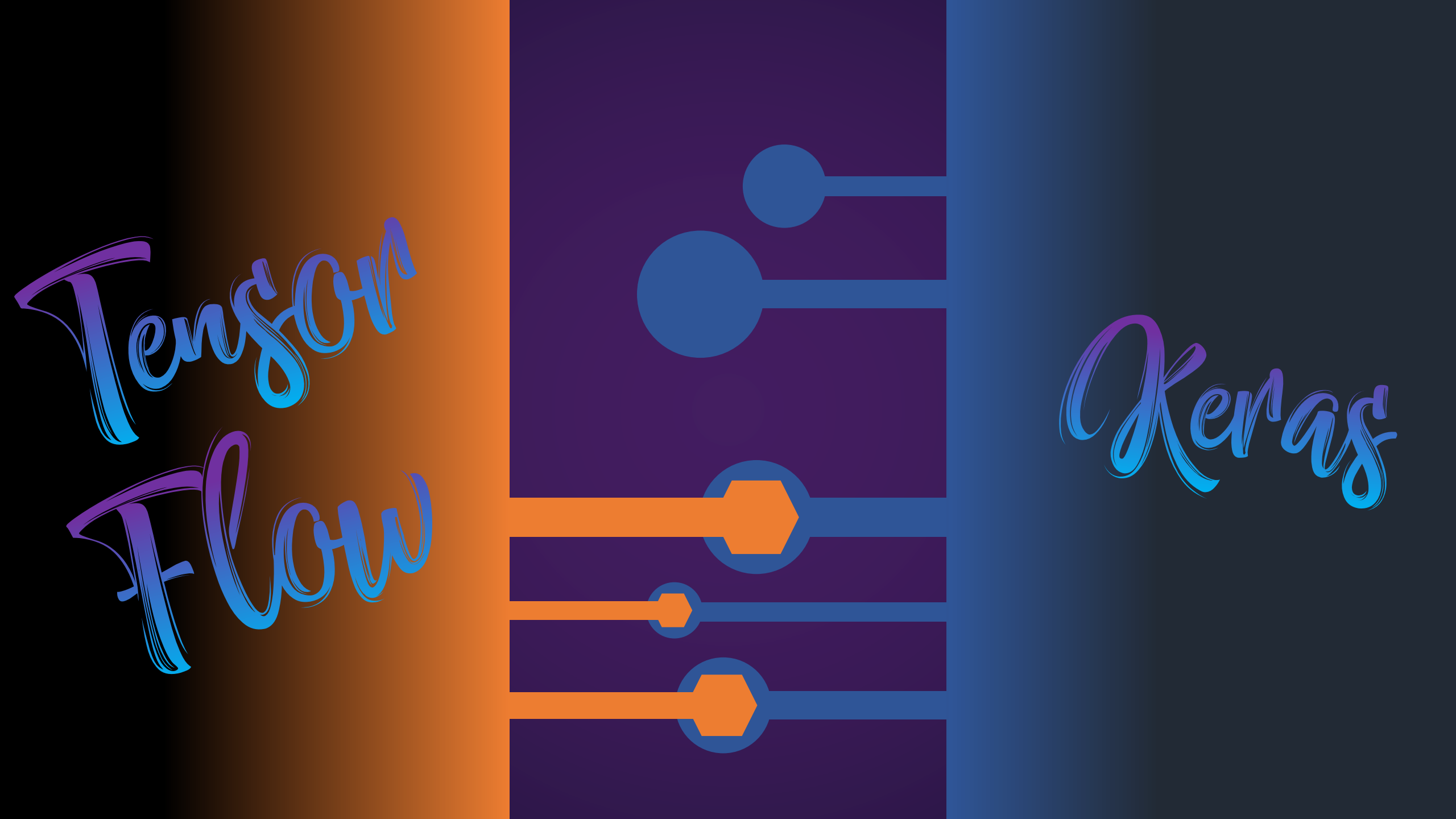
TensorFlow

Keras


TensorFlow
Keras

TensorFlow

Keras


TensorFlow
Keras

Versus

+ Schneller
+ Flexibler

+ Leserlicher
+ Abstrakter

In der Sprache

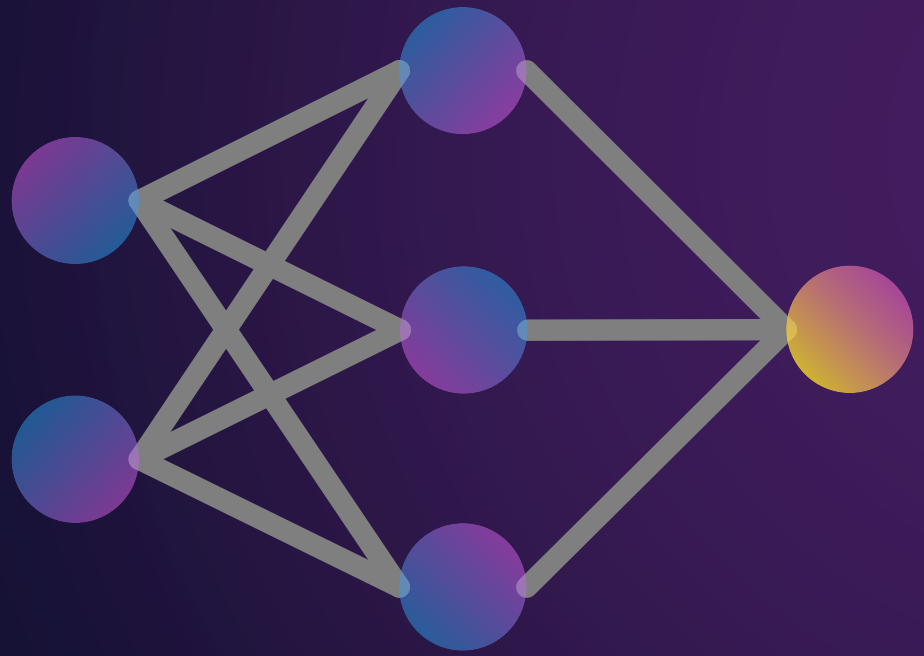
Python

- Neuronales Netz
- Quellcode der das Netz erzeugt und trainiert
- Vergleich von Keras und TensorFlow Code

Modell

Keras

Modell



Keras

Modell

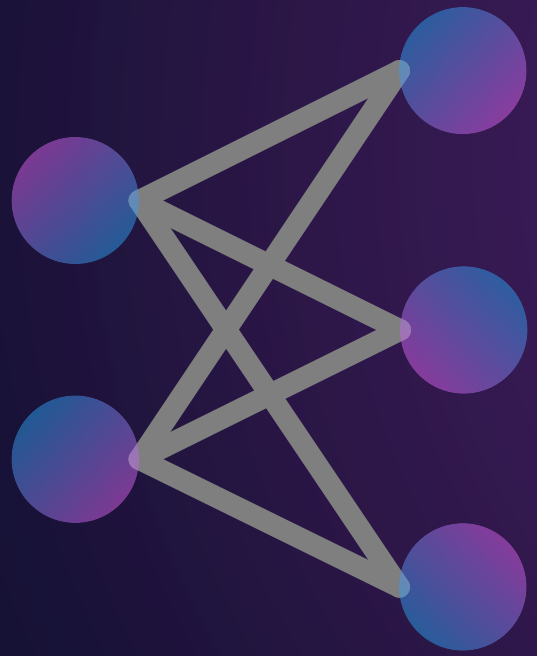
Keras

Modell

Keras

```
model = Sequential()
```

Modell

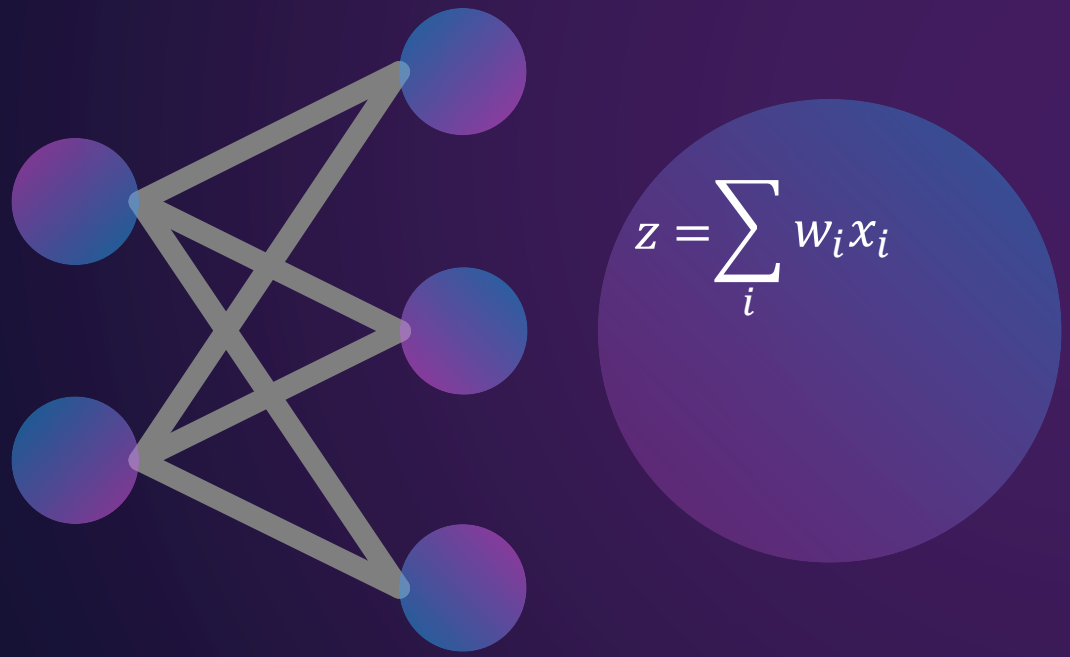


Keras

```
model = Sequential()

model.add(Dense(3, input_dim=2))
```

Modell

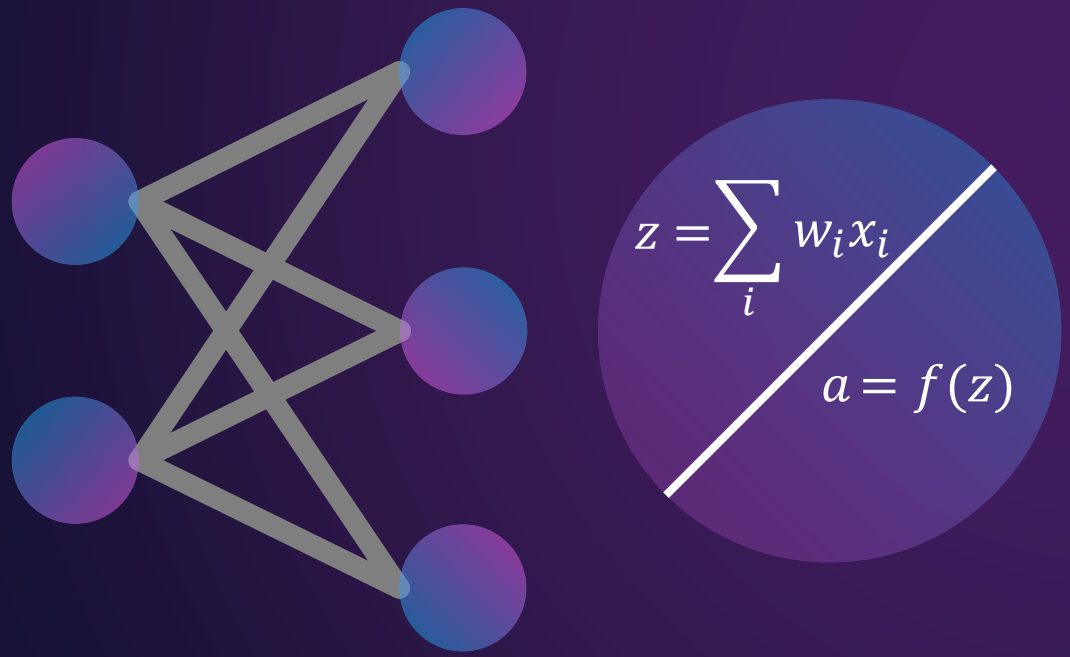


Keras

```
model = Sequential()

model.add(Dense(3, input_dim=2))
```

Modell

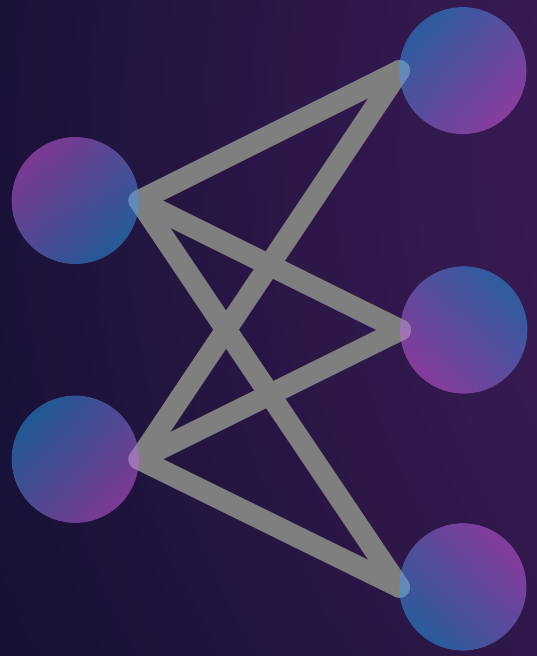


Keras

```
model = Sequential()

model.add(Dense(3, input_dim=2))
model.add(Activation('sigmoid'))
```

Modell

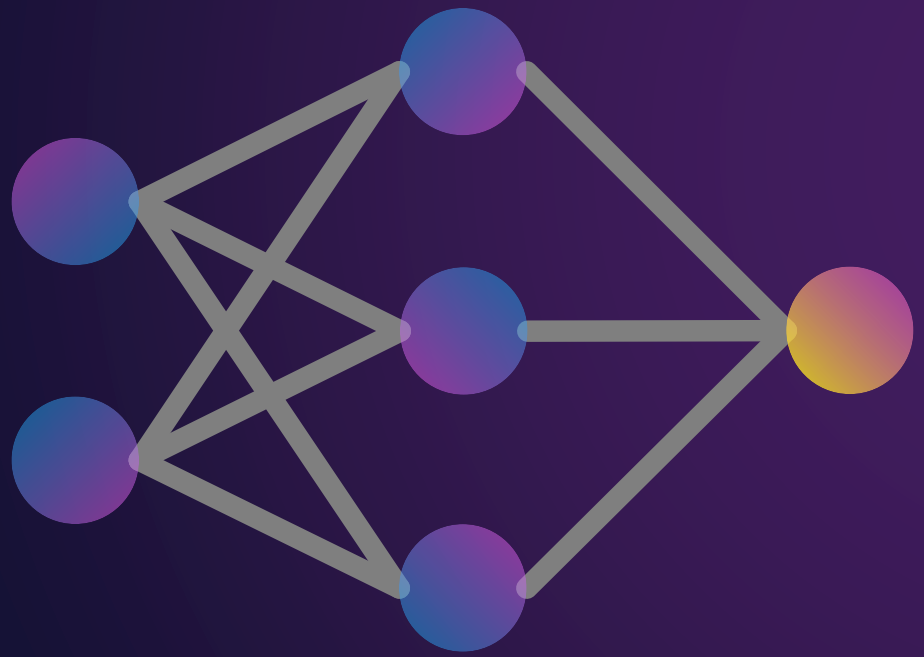


Keras

```
model = Sequential()

model.add(Dense(3, input_dim=2))
model.add(Activation('sigmoid'))
```

Modell



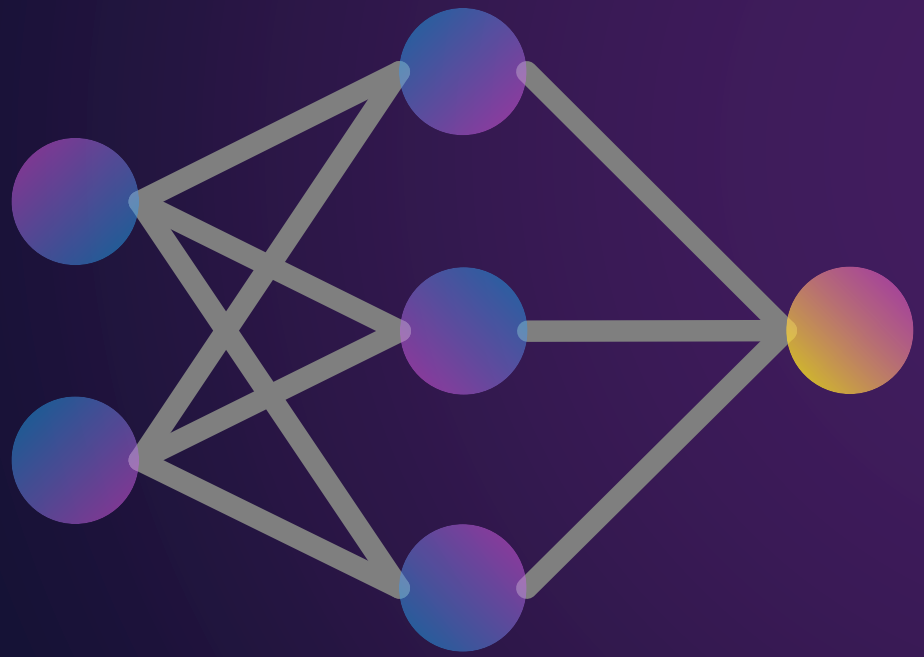
Keras

```
model = Sequential()

model.add(Dense(3, input_dim=2))
model.add(Activation('sigmoid'))

model.add(Dense(1))
```

Modell



Keras

```
model = Sequential()

model.add(Dense(3, input_dim=2))
model.add(Activation('sigmoid'))

model.add(Dense(1))
model.add(Activation('sigmoid'))
```

Keras

```
model = Sequential()

model.add(Dense(3, input_dim=2))
model.add(Activation('sigmoid'))

model.add(Dense(1))
model.add(Activation('sigmoid'))
```

TensorFlow

Keras

```
model = Sequential()

model.add(Dense(3, input_dim=2))
model.add(Activation('sigmoid'))

model.add(Dense(1))
model.add(Activation('sigmoid'))
```

TensorFlow

```
x = tf.placeholder(
        shape=[None, 2],
        dtype=tf.float32)
```

Keras

```
model = Sequential()

model.add(Dense(3, input_dim=2))
model.add(Activation('sigmoid'))

model.add(Dense(1))
model.add(Activation('sigmoid'))
```

TensorFlow

```
x = tf.placeholder(
        shape=[None, 2],
        dtype=tf.float32)
l1 = tf.layers.dense(x, 3,
        activation=tf.nn.sigmoid)
```

Keras

```
model = Sequential()

model.add(Dense(3, input_dim=2))
model.add(Activation('sigmoid'))

model.add(Dense(1))
model.add(Activation('sigmoid'))
```

TensorFlow

```
x = tf.placeholder(
        shape=[None, 2],
        dtype=tf.float32)
l1 = tf.layers.dense(x, 3,
        activation=tf.nn.sigmoid)
l2 = tf.layers.dense(l1, 1,
        activation=tf.nn.sigmoid)
```

Keras

```
model = Sequential()

model.add(Dense(3, input_dim=2))
model.add(Activation('sigmoid'))

model.add(Dense(1))
model.add(Activation('sigmoid'))
```

TensorFlow

```
x = tf.placeholder(
        shape=[None, 2],
        dtype=tf.float32)
l1 = tf.layers.dense(x, 3,
        activation=tf.nn.sigmoid)
l2 = tf.layers.dense(l1, 1,
        activation=tf.nn.sigmoid)
```

Keras

```
model = Sequential()

model.add(Dense(3, input_dim=2))
model.add(Activation('sigmoid'))

model.add(Dense(1))
model.add(Activation('sigmoid'))
```

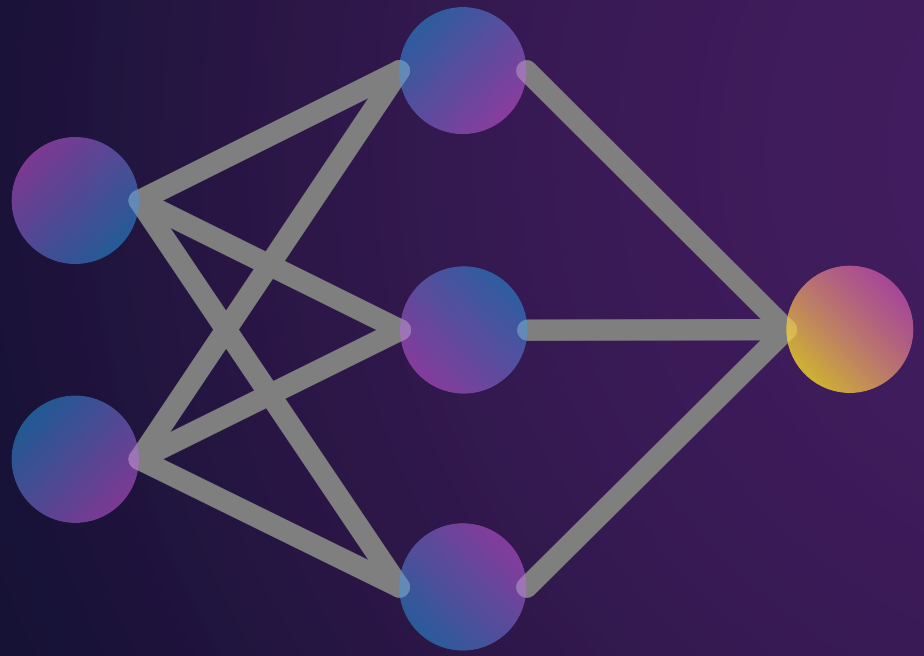
TensorFlow

```
x = tf.placeholder(
        shape=[None, 2],
        dtype=tf.float32)
l1 = tf.layers.dense(x, 3,
        activation=tf.nn.sigmoid)
l2 = tf.layers.dense(l1, 1,
        activation=tf.nn.sigmoid)
```

Keras

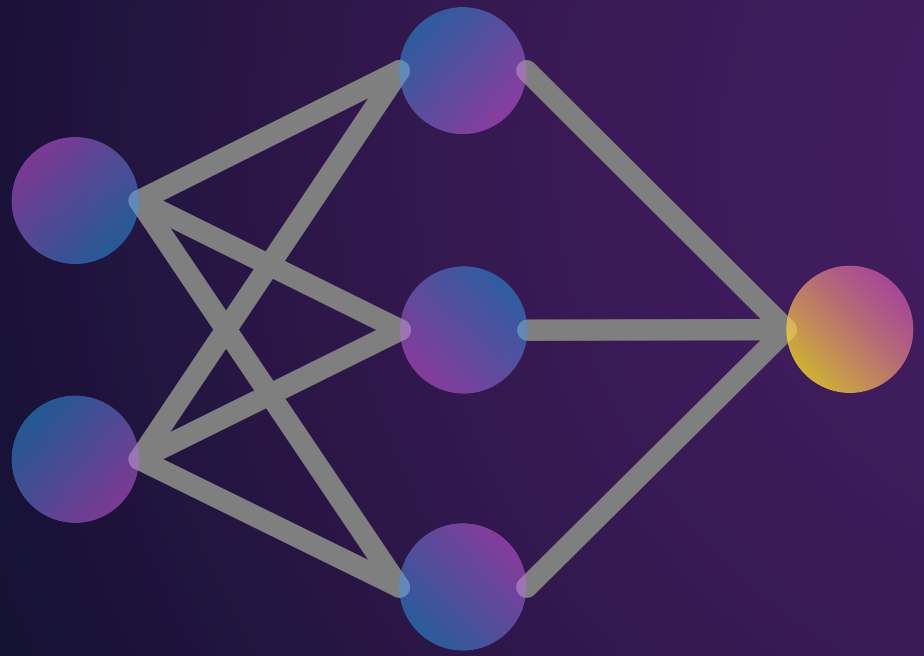
TensorFlow

Modell



Keras

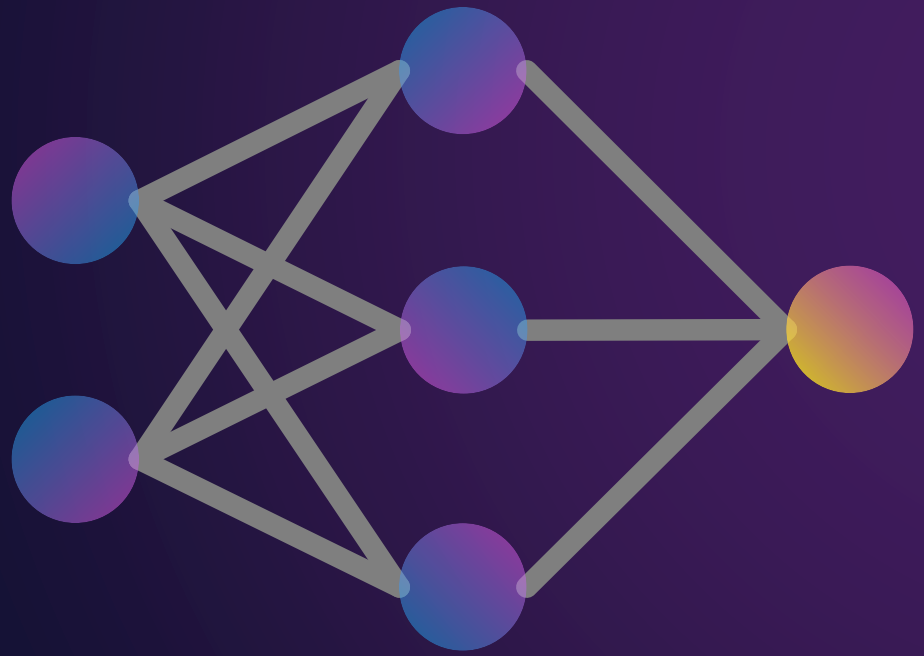
Modell



Keras

```
model.compile(loss='mse',
              optimizer=adam(lr=0.001))
```

Modell

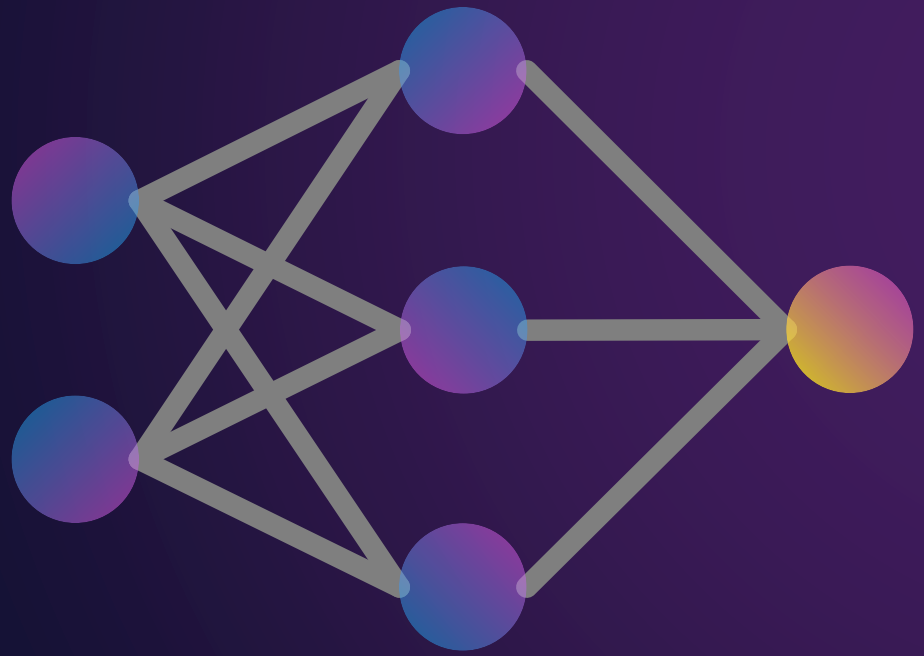


Keras

```
model.compile(loss='mse',
              optimizer=adam(lr=0.001))

model.fit(x=x_train, y=y_train,
          batch_size=128,
          epochs=10)
```

Modell



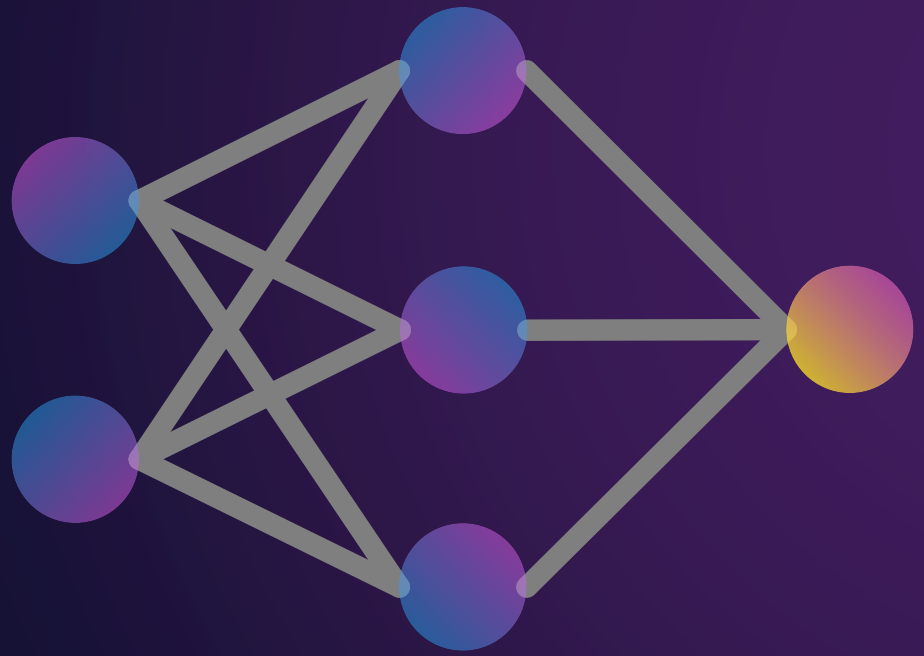
Keras

```
model.compile(loss='mse',
              optimizer=adam(lr=0.001))

model.fit(x=x_train, y=y_train,
          batch_size=128,
          epochs=10)

model.evaluate(x=x_test, y=y_test)
```

Modell



Keras

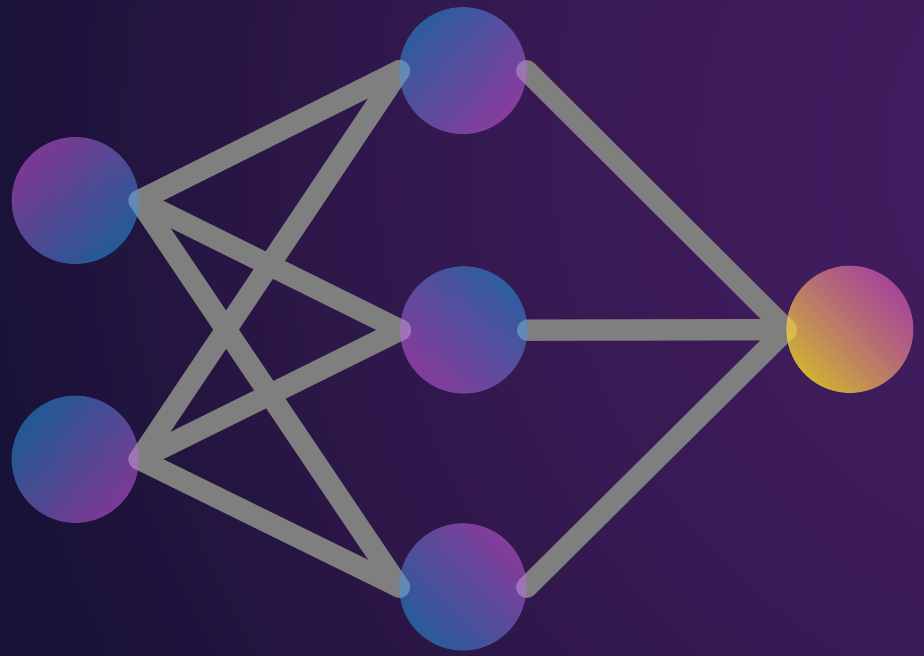
```
model.compile(loss='mse',
              optimizer=adam(lr=0.001))

model.fit(x=x_train, y=y_train,
          batch_size=128,
          epochs=10)

model.evaluate(x=x_test, y=y_test)

model.pedict(x=x_new)
```

Modell



Keras

```
model.compile(loss='mse',
              optimizer=adam(lr=0.001))
```

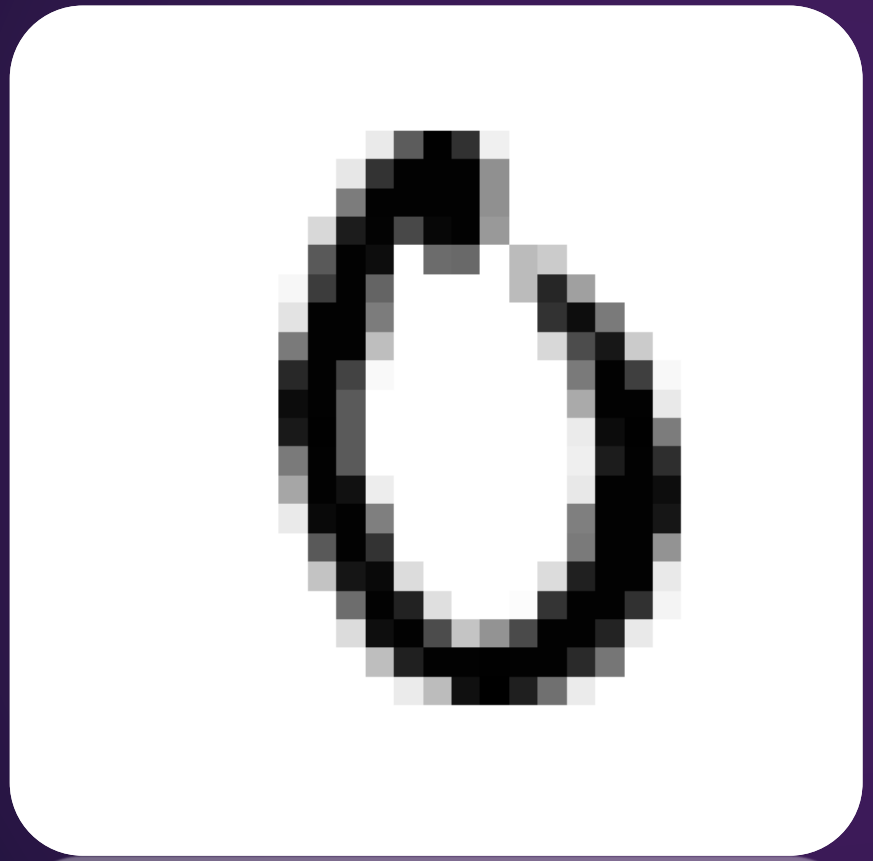
```
model.fit(x=x_train, y=y_train,
          batch_size=128,
          epochs=10)
```

Problem

Handgeschriebene
Zahlen
klassifizieren

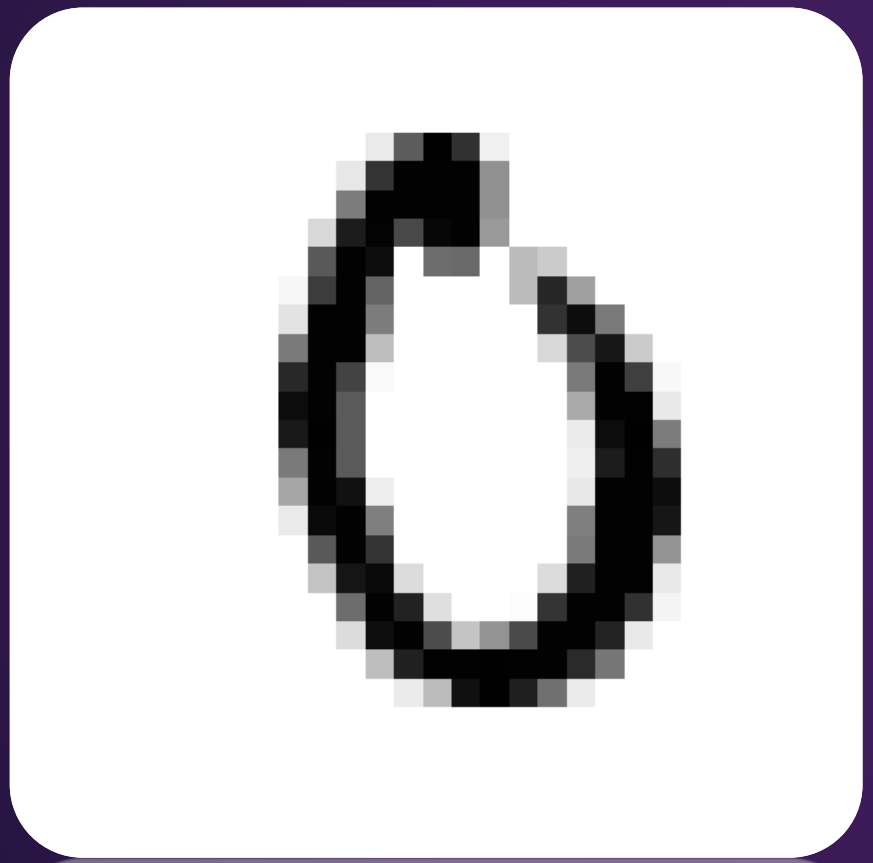
0 1 2 3 4

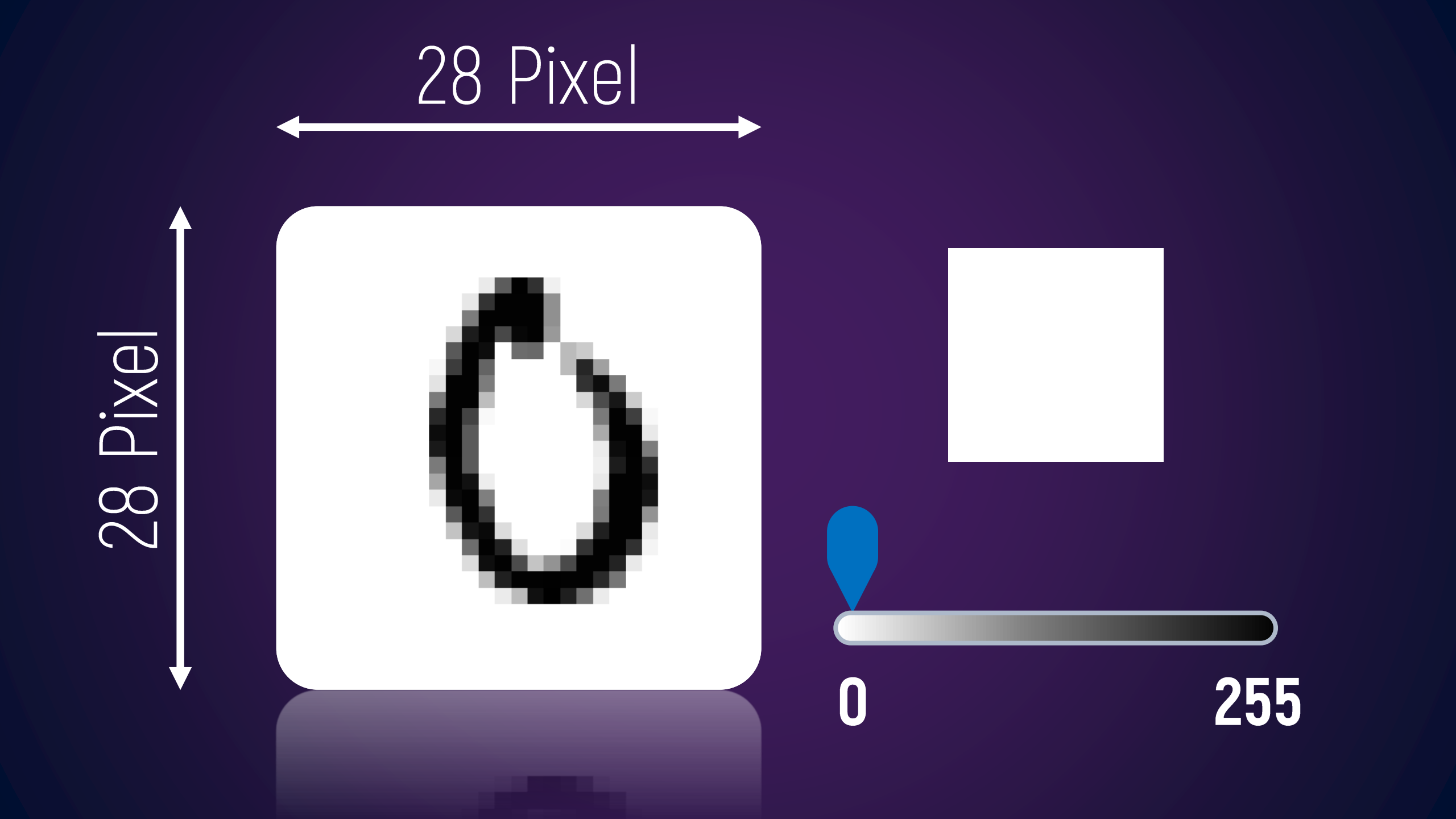
5 6 7 8 9

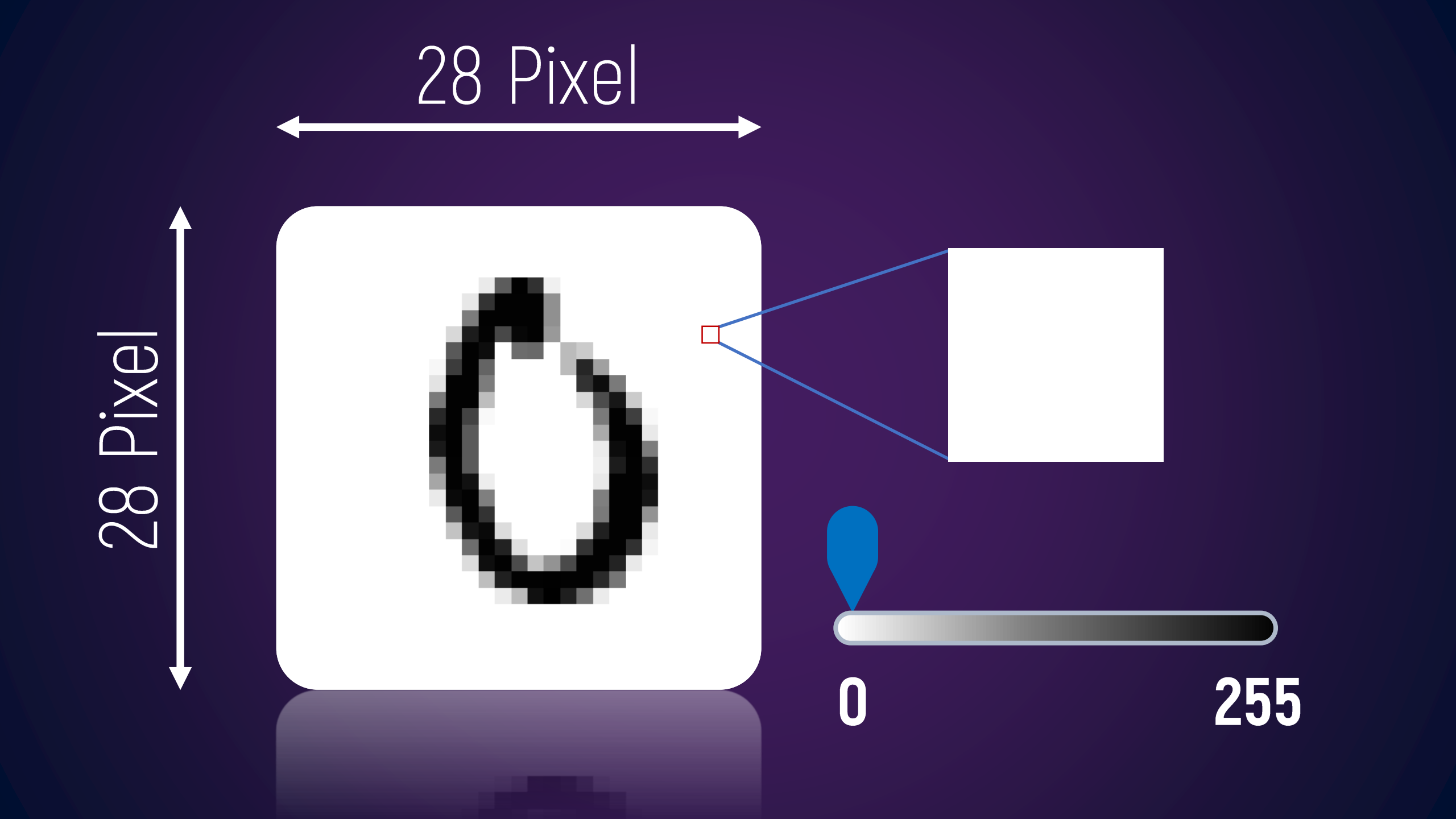


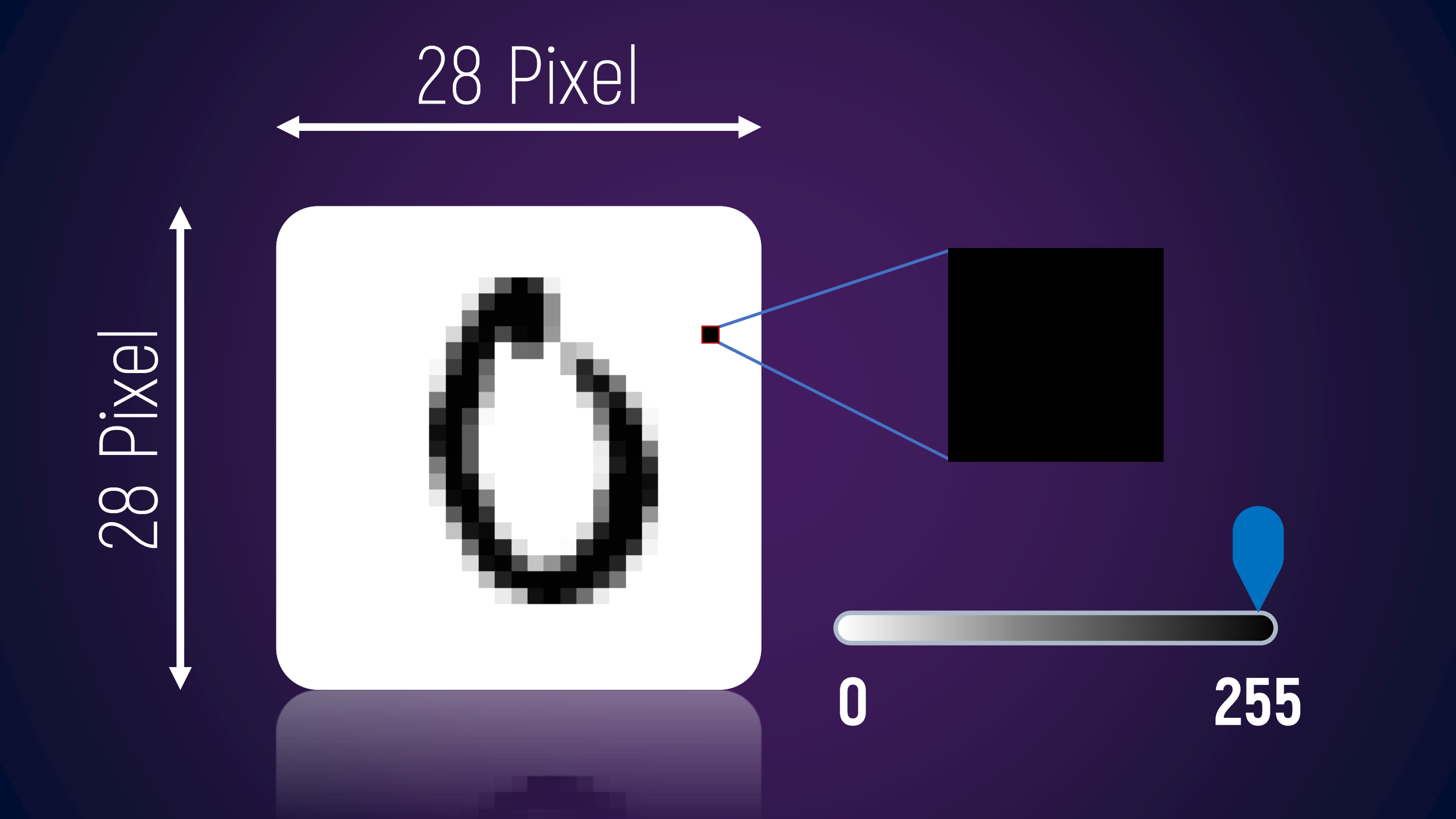
28 Pixel

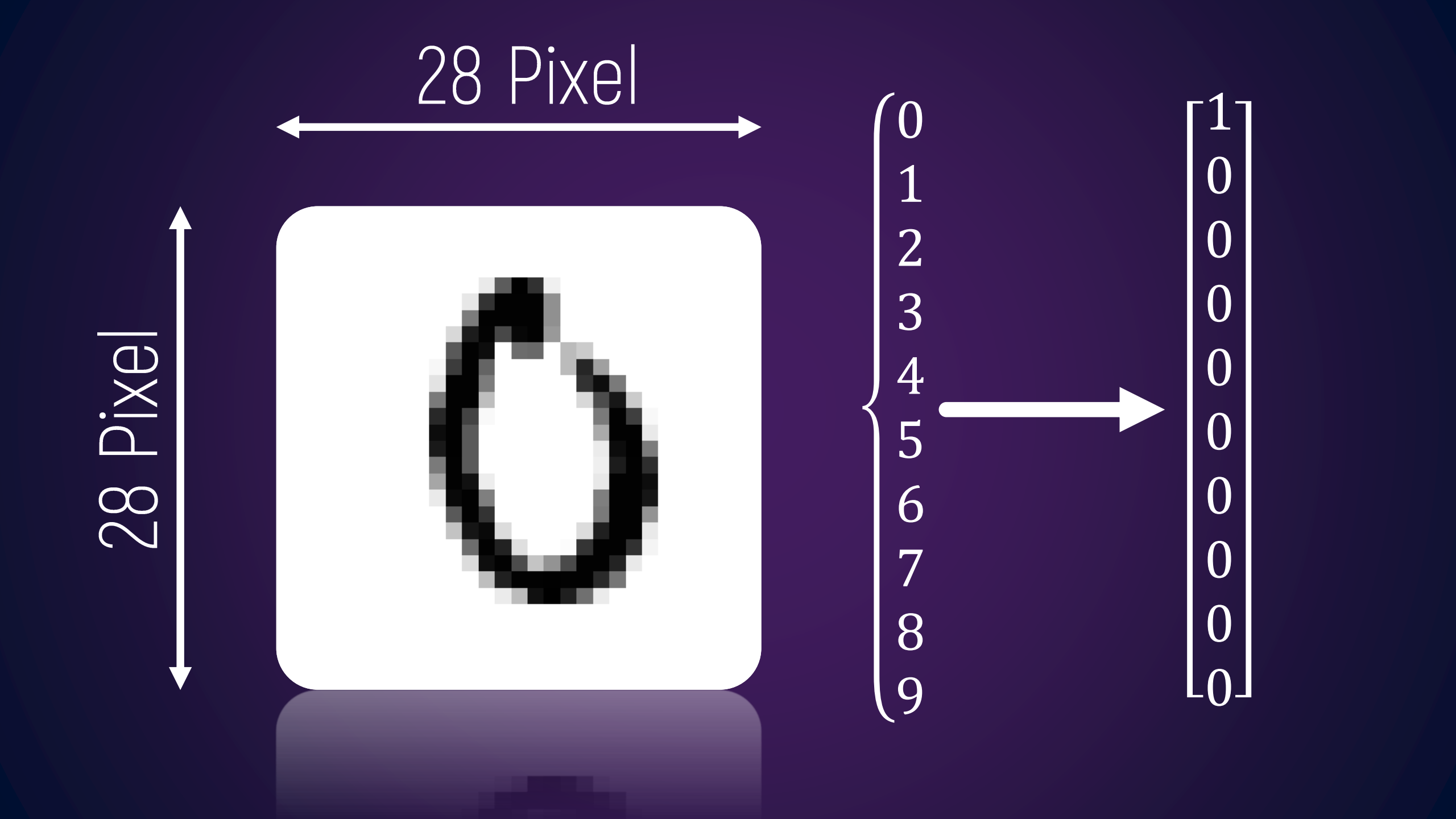
28 Pixel

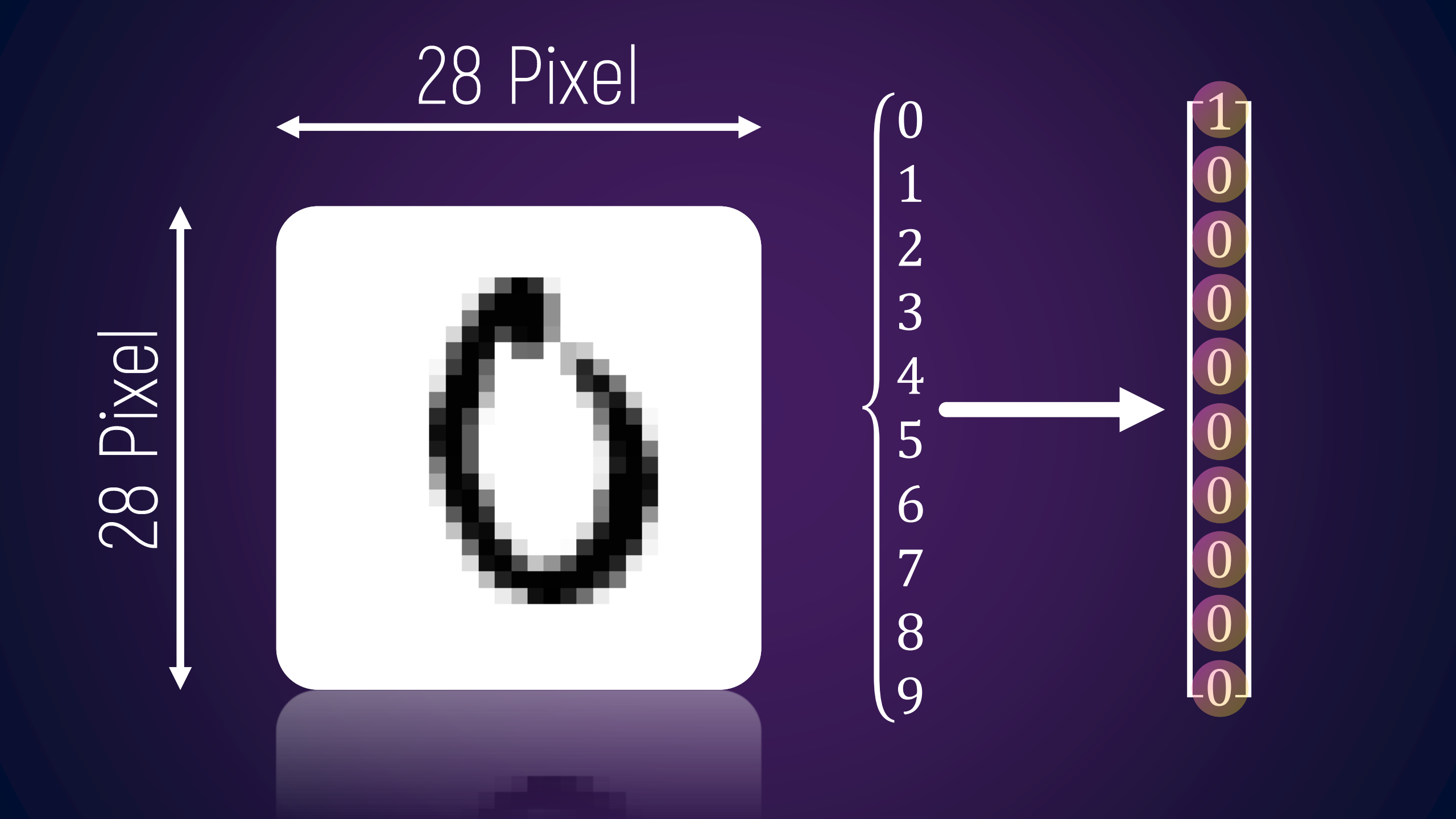


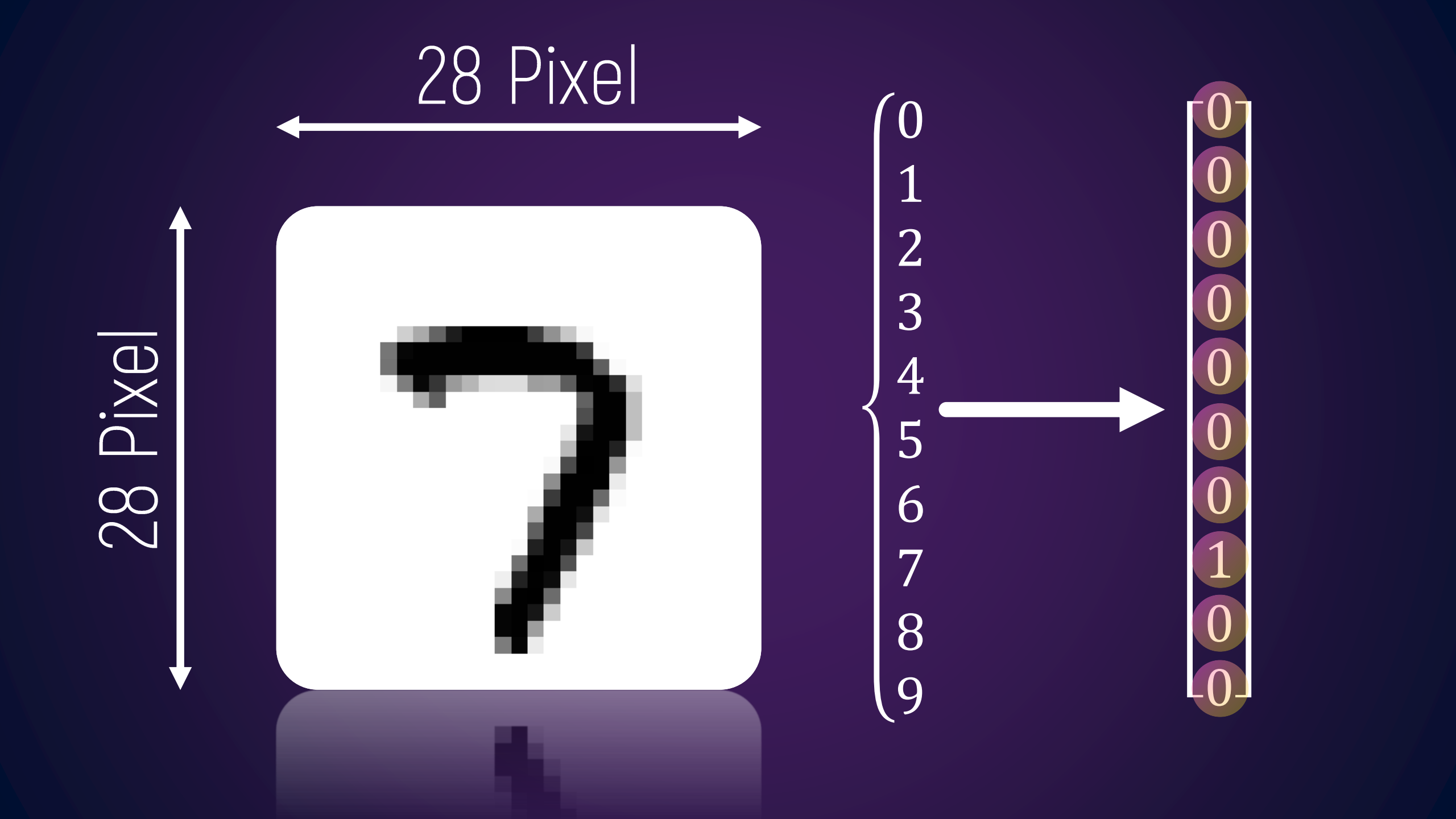

28 Pixel
28 Pixel
0
255

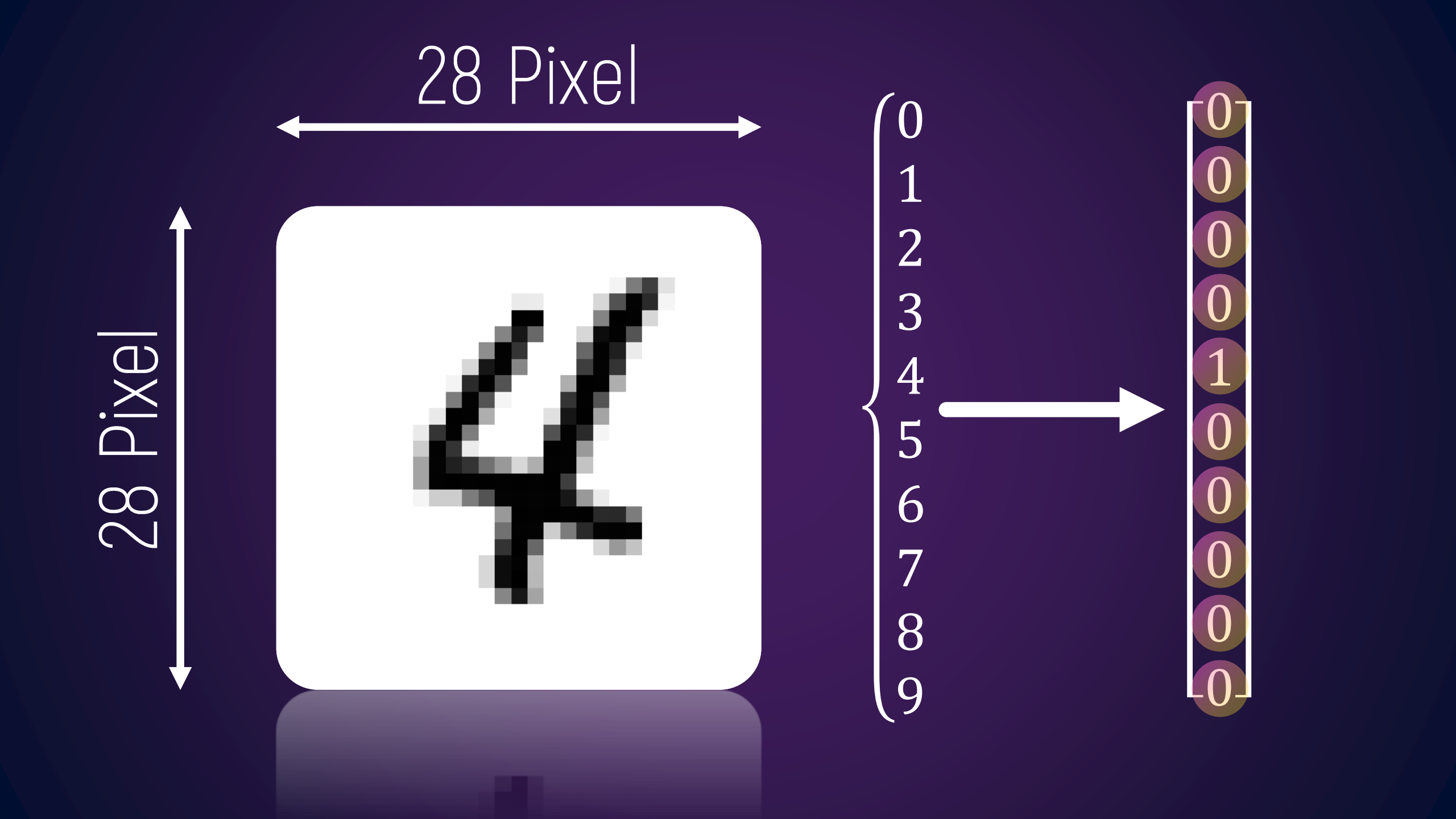

28 Pixel
28 Pixel
0
255


28 Pixel
28 Pixel
0
255


28 Pixel
28 Pixel
0
1
2
3
4
5
6
7
8
9
1
0
0
0
0
0
0
0
0
0


28 Pixel
28 Pixel
0
1
2
3
4
5
6
7
8
9
1
0
0
0
0
0
0
0
0


28 Pixel
28 Pixel
0
1
2
3
4
5
6
7
8
9
0
0
0
0
0
0
1
0
0


28 Pixel
28 Pixel
0
1
2
3
4
5
6
7
8
9
0
0
0
0
1
0
0
0
0
0

Live Demo

Deep Learning

Summary

- Was ist Deep Learning
- Warum brauchen wir Deep Learning
- Typische Deep Learning Probleme
- Wie funktioniert Deep Learning

Neuronales Netze

Summary

- Geschichte
- Aufbau
  - Schichten
  - Neuronen

Lernalgorithmus

- Supervised Learning
- Gradient Descent
- Hyperparameter

Summary

Keras und TensorFlow

- TensorFlow - Namensgebung
- Keras baut auf TensorFlow auf
- Vor- und Nachteile von Keras und TensorFlow

Programmieren einer KI in Keras

Summary

- Quellcode
- Keras und TensorFlow Codevergleich
- Neuronales Netz in Keras trainieren

Inhalt

- Resourcen für Selbststudium
- Das eben gezeigt Jupyter Notebook
- Diese Präsentation im PDF-Format

Fragen