

## Vertiefung zu neuronalen Netzen



### Arbeitsauftrag Module:

Notieren Sie im nachfolgenden Feld welche Aufgaben die jeweils genannten Module haben:



```
import numpy
```

vereinfacht und verbessert das arbeiten mit arrays

```
import scipy.special
```

baut auf numpy drauf auf  
manipulieren und visualisieren von daten

```
import matplotlib.pyplot
```

statische, interaktive oder animierte visualisierungen (graphen)



### Arbeitsauftrag Bibliotheken vs. Frameworks:

Unterschieden Sie Bibilotheken von Frameworks:

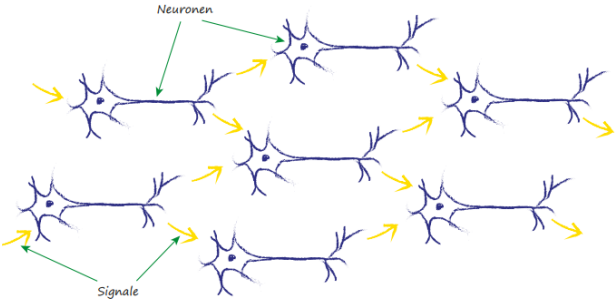
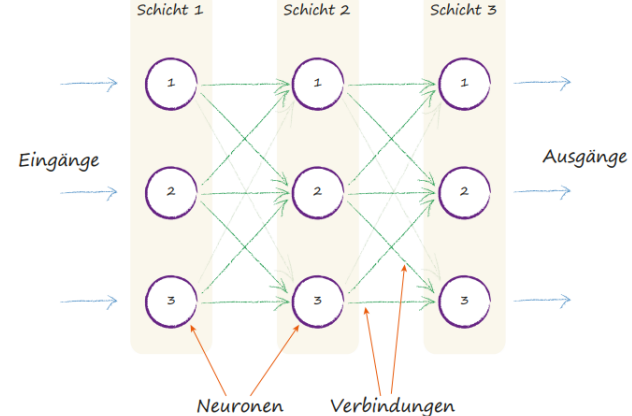
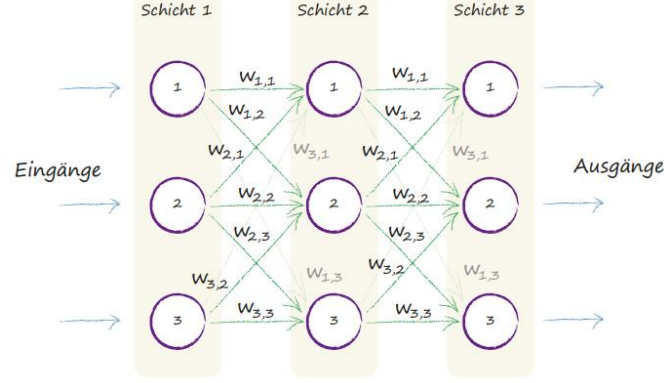
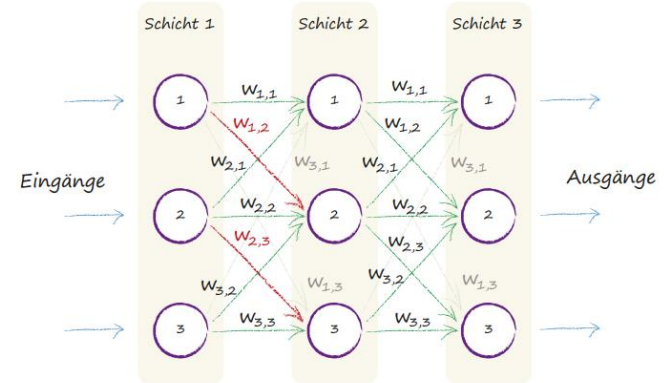


Bibilotheken	Frameworks
programm benutzt fertige code abschnitte von der heruntergeladenen Bibliothek	Framework benutzt eigen geschriebenes Programm



## Arbeitsauftrag Knoten verstehen:

Ergänzen Sie nachfolgende Übersicht mit den wesentlichen Informationen aus dem Erklärvideo.

<p><b>Menschliches Netz</b></p>		
<p><b>Künstliches Netz</b></p>		<p>knoten werden gebaut mehrere schichten jedes neuron ist mit einem aus der nächsten verbunden</p>
<p><b>Künstliches Netz - Schichtenmodell</b></p>	 <p>Abbildung 1-35: Schichtenmodell, in dem den Verbindungen zwischen den Knoten Gewichte zugeordnet sind</p>	<p>Statisch</p>
<p><b>Künstliches Netz - Verbindungen</b></p>	 <p>Abbildung 1-36: Die im Text beschriebenen Verbindungen sind hier rot gekennzeichnet.</p>	<p>wird optimiert im verlauf des Trainings</p>

**Arbeitsauftrag Knoten (Nodes) im Netz:**

Ergänzen Sie nachfolgende Abbildungen stichpunktartig mit Erklärungen aus dem Text der Learning Snack.

```
# initialise the neural network
def __init__(self, inputnodes, hiddennodes, outputnodes, learningrate):
    # set number of nodes in each input, hidden, output layer
    self.inodes = inputnodes
    self.hnodes = hiddennodes
    self.onodes = outputnodes

    # learning rate
    self.lr = learningrate
    pass
```

```
# number of input, hidden and output nodes
input_nodes = 3
hidden_nodes = 3
output_nodes = 3
```

```
# learning rate is 0.3
learning_rate = 0.3
```

```
# create instance of neural network
n = neuralNetwork(input_nodes,hidden_nodes,output_nodes, learning_rate)
```

**Arbeitsauftrag Gewichte im Netz:**

Ergänzen Sie nachfolgende Abbildungen stichpunktartig mit Erklärungen aus dem Text der Learning Snack.

```
# link weight matrices, wih and who
# weights inside the arrays are w_i_j, where link is from node i to node j in the next layer
# w11 w21
# w12 w22 etc
self.wih = numpy.random.normal(0.0, pow(self.inodes, -0.5), (self.hnodes, self.inodes))
self.who = numpy.random.normal(0.0, pow(self.hnodes, -0.5), (self.onodes, self.hnodes))
```



**Arbeitsauftrag:** Notieren Sie sich Ihre Antworten zu der Formsbefragung und korrigieren Sie diese, falls Ihre Antworten fehlerhaft waren, anschließend.



1. Python nutzt folgende Übersetzungsart:

(1 Punkt)

- ☐ Compiler
- ☒ Interpreter
- ☐ Mehrfachinterpreter
- ☐ Translator
- ☐ Constructor

2. Python ist eine durchgängig objektorientierte Programmiersprache (1 Punkt)

- ☐ Ja
- ☒ Nein

3. Ein neuronales Netz in einer Klasse besteht grundsätzlich aus folgenden Methoden: (1 Punkt)

- ☒ train()
- ☒ query()
- ☐ flip()
- ☐ sleep()
- ☐ fire()

4. Um ein neuronales Netz in Python nutzen zu können, sind folgende Module nötig: (1 Punkt)

- ☒ numpy
- ☒ scipy.special
- ☐ math
- ☐ network
- ☐ train

5. Welche Beschreibungen des Unterschieds zwischen einem Framework und einer Bibliothek sind korrekt? (1 Punkt)

- ☐ Beide Begriffe werden als Synonyme verwendet.
  - ☒ Ein Framework ist kein fertiges Programm, sondern muss an vorgesehenen Stellen ergänzt werden.
  - ☒ Eine Bibliothek ist eine Sammlung von Klassen und Funktionen
- Bei der Verwendung einer Bibliothek bleibt die Kontrolle beim Entwickler, der in der Anwendung Bibliotheksfunktionen aufrufen lässt. Bei der Verwendung eines Frameworks ist die Kontrolle umgekehrt, d. h. das Framework sagt dem Entwickler, wo Code bereitgestellt werden muss, und ruft diesen nach Bedarf auf.

6. Welche Arten von Knoten gibt es in neuronalen Netzen? (1 Punkt)

- ☒ Hidden nodes
- ☐ Public nodes
- ☒ Output nodes
- ☒ Input nodes
- ☐ Private nodes

7. Markieren Sie alle korrekten Aussagen, die die Funktionsweise von Gewichten in neuronalen Netzen beschreiben: (1 Punkt)

- ☒ Gewichte bestimmen die Signalanteile, die das Netz in Vorwärtsrichtung weiterleitet, und die Fehleranteile, die das Netz in Rückwärtsrichtung durchlaufen.
- ☒ Es sind die Verknüpfungsgewichte, die verfeinert werden, um die Reaktion des Netzes auf Eingaben zu verbessern.
- ☐ Das Gewicht eines neuronalen Netzes ist zwischen 0 und 1
- ☐ Die Anfangswerte der Verknüpfungsgewichte sollten klein und zufällig sein
- ☐ Die numpy-Funktion generiert ein Array von Werten, die zufällig aus einem Bereich zwischen 0 und 1 ausgewählt werden.