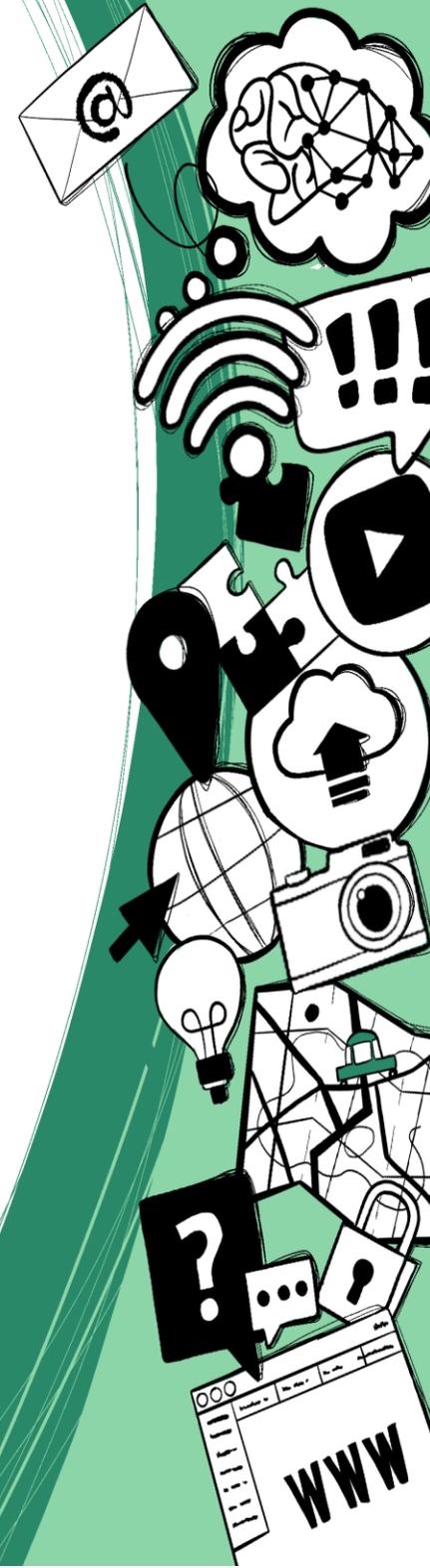


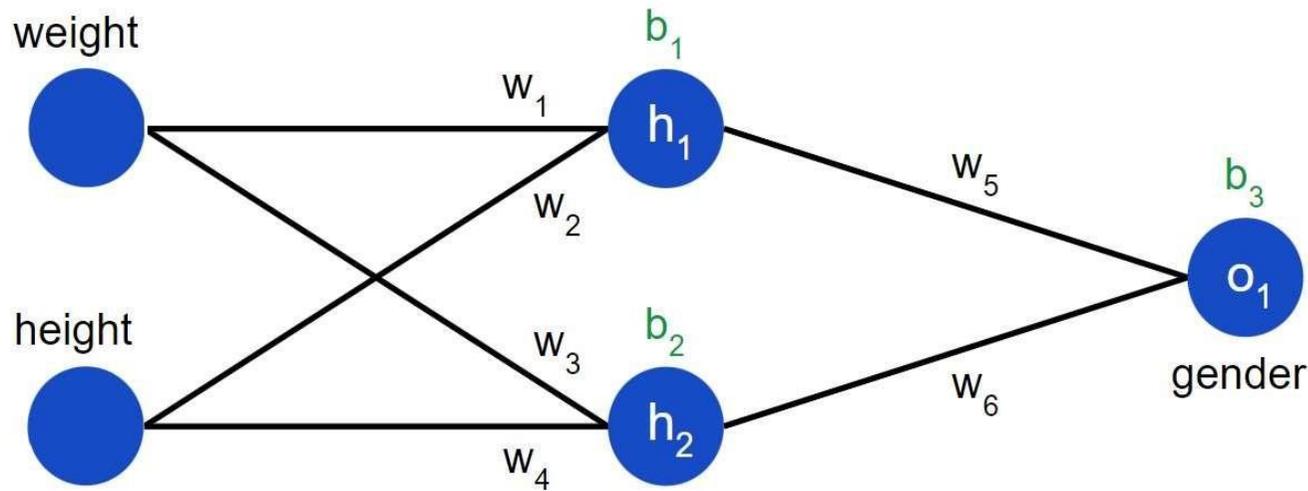
# Neuronale Netze III praktisches Beispiel



Wir wollen das bisher Gelernte vertiefen und einmal den Vorgang in einem Neuronalen Netz selbst durchrechnen.

Dazu verwenden wir aber ein wesentlich einfacheres Netz und zeigen gleichzeitig den Anwendungsfall einer nicht-grafischen Aufgabenstellung.





Das Netz besteht aus nur 5 Neuronen.

- 2 Eingabe-Neuronen
- 2 Neuronen in einer versteckten Schicht (hidden Layer)
- 1 Ausgabe-Neuron

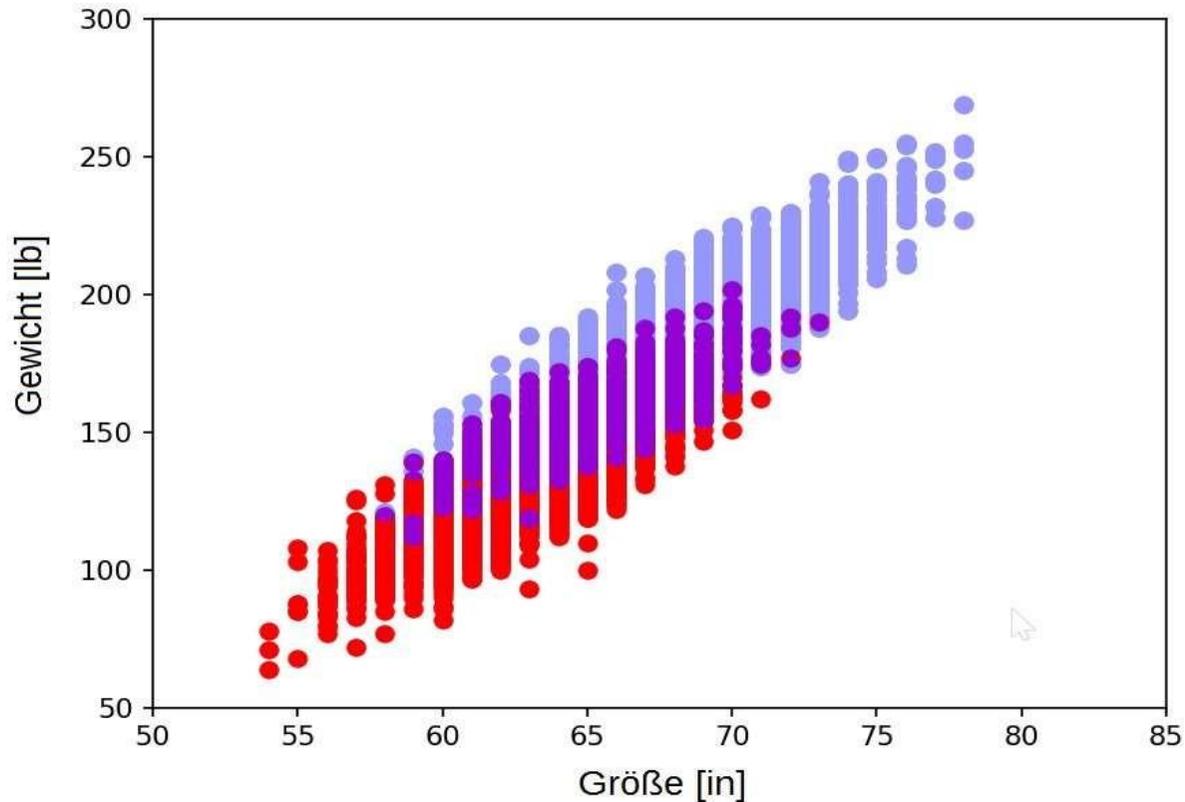
Das Netzwerk soll in den beiden Eingabe-Neuronen das

- Gewicht [kg] und die
- Größe [cm]

einer Person einlesen.



Im Ausgabe-Neuron soll angegeben werden, ob die entsprechende Person eher weiblich oder eher männlich ist. Ist das überhaupt möglich?



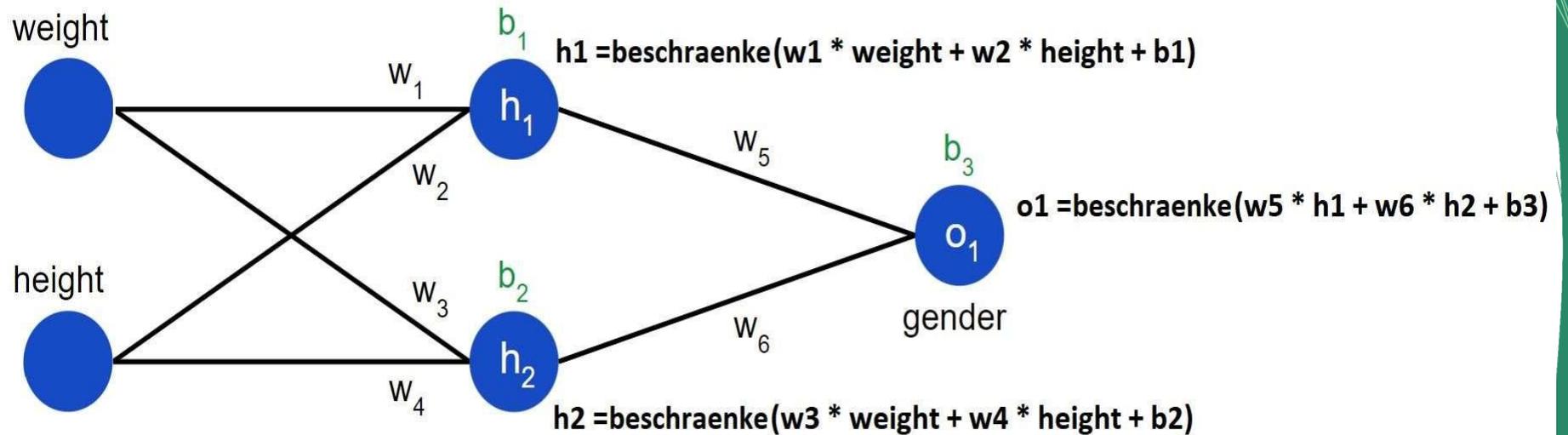
Ja, zumindest teilweise. Obige Grafik zeigt die Verteilung von Gewicht und Größe aus einer Liste von 500 zufälligen Personen.



Wir sehen zwar einen Überlappungsbereich in dem keine guten Vorhersagen möglich sein werden aber grundsätzlich, wenn es nur um Wahrscheinlichkeiten geht, kann eine gewisse Vorhersage gemacht werden. Das Netzwerk soll in der Ausgabe-Schicht angeben ob eine Frau oder ein Mann vermutet wird.

- Werte nahe 1 sollen auf eine Frau hindeuten
- Werte nahe 0 auf einen Mann
- Werte um 0.5 sollen ein unbestimmtes Ergebnis zeigen





- $h_1$  und  $h_2$  sind die Aktivierungen der 2 inneren (versteckten) Neuronen.
- In  $o_1$  ist der Ausgabewert
- $w_1$ - $w_6$  sind die Gewichte
- $b_1$ - $b_3$  die Biases.

$h_1$ ,  $h_2$  und  $o_1$  werden wie oben gezeigt berechnet.



Die Funktion "beschraenke()" setzen wir ein, damit die Werte in den Knoten immer im Intervall  $[-1, 1]$  bleiben.

Sonst könnten die Aktivierungen zu stark ansteigen. Werte, die sowieso schon zwischen 0 und 1 liegen werden kaum oder nicht verändert, Werte die zu groß oder zu klein sind werden so gestaucht, dass sie in die gewünschten Grenzen fallen.



Erweiterter Inhalt, Training des Netzwerks: **training.pptx**

Im Simulator (siehe index.html im Verzeichnis simulator) kannst du selbst für verschiedene Eingaben die Antwort des Netzes ausprobieren.



# Links

Machine Learning for Beginners: An Introduction to Neural Networks <https://towardsdatascience.com/machine-learning-forbeginners-an-introduction-to-neural-networksd49f22d238f9>

Wie oben auf der Seite des Autors <https://victorzhou.com/blog/intro-to-neural-networks/>

Quelle der Gewichtsdaten <https://www.kaggle.com/>

Python Programm dieses Netzwerks [simple\\_nn.py](#) Eine weiter Version mit "leaky ReLU" statt mir Sigmoid: [simple\\_nn\\_leaky\\_relu.py](#)

