

# Proseminar

## Künstliche neuronale Netze

Jens Ostertag

30. April 2022

### Inhaltsverzeichnis

<b>1</b>	<b>Computer lernen lassen</b>	<b>2</b>
1.1	Aufbau eines einfachen neuronalen Netzes [1] . . . . .	4
1.2	Drei Arten des maschinellen Lernens [2] . . . . .	5
1.2.1	Überwachtes Lernen . . . . .	5
1.2.2	Unüberwachtes Lernen . . . . .	6
1.2.3	Bestärktes Lernen . . . . .	6
1.3	Begriffskonventionen . . . . .	7
1.4	Umgang mit neuronalen Netzen [2] . . . . .	7
1.4.1	Vorbereiten eines Trainingsdatensatzes . . . . .	7
1.4.2	Trainieren . . . . .	8
1.4.3	Auswertung und Anwendung . . . . .	8
1.5	Programmiersprachen zur Entwicklung neuronaler Netze [2] . .	8
<b>2</b>	<b>Einfache Lernalgorithmen für einzelne Neuronen [3]</b>	<b>9</b>
<b>3</b>	<b>Entwicklung eines neuronalen Netzes [4]</b>	<b>9</b>
<b>4</b>	<b>Bildklassifikation mit tiefen neuronalen Netzen [5]</b>	<b>9</b>
<b>5</b>	<b>Resultate</b>	<b>9</b>

# 1 Computer lernen lassen

„Künstliche Intelligenz“, „Maschinelles Lernen“ und „Neuronale Netze“ - Alle diese Begriffe sind gewissermaßen miteinander verbunden, denn ihre Technik beruht darauf, dass Computer lernen können.

Bei einem künstlichen neuronalen Netz handelt es sich um die Simulation des menschlichen neuronalen Netzes, einer Struktur, die im Gehirn vorhanden ist. Sogenannte „Neuronen“ (Nervenzellen) sind durch Synapsen miteinander verbunden, welche den Elektronenfluss zwischen zwei Neuronen mithilfe eines „Synapsengewichts“ regeln. Dadurch lässt sich die Anfälligkeit eines Neurons auf die Elektronen des vorhergehenden Neurons verändern. [6]

Maschinelles Lernen beschreibt das Lernen eines solchen neuronalen Netzes. Wie auch der Mensch sind künstliche neuronale Netze dazu in der Lage, die unterschiedlichsten Dinge zu lernen. Das geschieht durch die Anpassung der zuvor genannten Synapsengewichte, im Folgenden bei künstlichen Netzen auch nur „Gewichte“ genannt.

Somit ist es neuronalen Netzen möglich, beispielsweise Muster in Bildern erkennen zu lernen und somit bestimmte Objekte voneinander unterscheiden zu können (Bildklassifizierung).

Die künstliche Intelligenz ist die undefinierteste Form neuronaler Netze, was hauptsächlich damit zusammenhängt, dass es keine sehr genaue Definition von Intelligenz gibt. Ist man intelligent, wenn man schnell rechnen kann? Somit ist bereits ein simpler Taschenrechner von vor 40 Jahren äußerst intelligent. Oder bedeutet Intelligenz, dass man eigenständig denken und fühlen können muss? Wegen dieser Undefiniertheit wird im Folgenden auf den Begriff der künstlichen Intelligenz verzichtet.

Obwohl die Theorie bereits länger existiert, erfolgte ein großer Vorsprung in der Entwicklung des maschinellen Lernens erst in den letzten Jahren. Auch wenn es zuerst nicht so scheint, kommt heute jeder täglich damit in Berührung, beispielsweise

- im Verkehr, durch schlau gesteuerte Ampelphasen oder nahezu selbst-fahrende Autos sowie Assistenzsysteme wie Spurhalteassistenten, Einparkhilfen oder Ähnliches,
- im Internet, wo jedem im Sekundentakt neue Inhalte vorgeschlagen werden,

- im medizinischen Bereich, wo Bilderkennung bei der Auswertung bildgebender Verfahren hilft,
- in der Industrie, wo Produktionsabläufe durch maschinelles Lernen optimiert wurden.

Im Folgenden soll es darum gehen, wie maschinelles Lernen funktioniert und wie man es für seine Ziele verwenden kann.

## 1.1 Aufbau eines einfachen neuronalen Netzes [1]

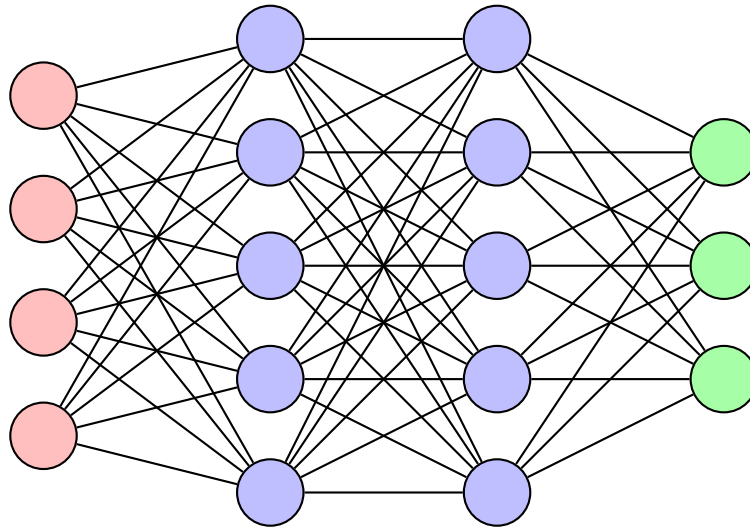


Abbildung 1: Aufbau eines einfachen neuronalen Netzes

Die kleinste Einheit eines neuronalen Netzes ist ein Neuron. Dabei wird zwischen Eingabe- (rot), Ausgabe- (grün) und verstecktem (blau) Neuron unterschieden. Da die Neuronen jedoch nicht strukturlos angeordnet sein sollen, werden sie zu Schichten zusammengefasst. Somit ergeben sich dann Eingabe-, Ausgabe- und versteckte Schichten.

Während die Funktion der Ein- und Ausgabeschicht trivial ist, nämlich das Eingeben eines Eingabevektors und das Auslesen des ermittelten Ausgabevektors, ist die Funktion von versteckten Schichten etwas komplexer.

Sie werden benötigt, da eine einzelne Schicht nur linear separierbare Funktionen anwenden könnte. Klassifikationen wären damit nahezu unmöglich. Im Umkehrschluss gilt: Je mehr versteckte Schichten es gibt, umso kompliziertere Dinge kann das Netz lernen, das ist allerdings auch mit einem höheren Zeitaufwand verbunden.

Die Neuronen sind, wie zuvor genannt, mit Synapsen verbunden. Das Synapsengewicht bestimmt im künstlichen Fall, welcher Anteil des Wertes des vorherigen Neurons in das folgende Neuron „propagiert“ (weitergeleitet) wird. Das ermöglicht, dass wie im dargestellten vollständigen neuronalen Netz einzelne Neuronen gegenüber anderen unsensibler oder sensibler werden.

## 1.2 Drei Arten des maschinellen Lernens [2]

Die Gewichtsanzpassung eines neuronalen Netzes, also sein Lernprozess, wird auch Training genannt. Dieses kann auf drei unterschiedliche Arten erfolgen.

### 1.2.1 Überwachtes Lernen

Das überwachte Lernen ist die wohl am häufigsten verwendete Methode, ein neuronales Netz zu trainieren. Es zielt darauf ab, auch zu bislang unbekannten Eingaben eine passende Ausgabe zu generieren, meistens geht es darum, die Eingaben in Gruppen bestimmter Eigenschaften einzuteilen.

Dafür wird während des Trainingsvorgangs mithilfe eines Trainingsdatensatzes (einige Eingaben mit zugehörigen erwarteten Ausgaben) angelernet, welche Muster in Eingabedaten zu bestimmten Ausgaben gehören. Erhält das Netz nach abgeschlossenem Training eine Eingabe, ist es in der Lage, ein Muster darin zu erkennen und anhand dessen eine passende Ausgabe zu generieren.

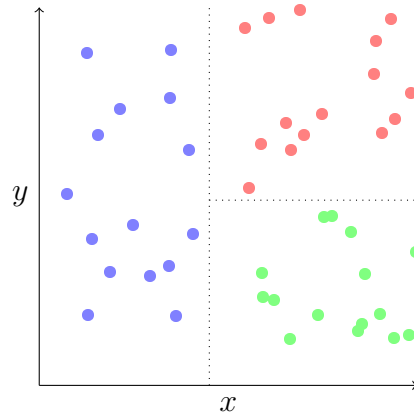


Abbildung 2: Klassifizierung von Daten

In Abbildung 2 entsprechen alle Punkte einer bestimmten Eingabe, gleichfarbige Punkte haben dieselbe erwartete Ausgabe. Während des Trainings lernt das Netz beispielsweise, dass es für Eingaben mit hohem  $x$ - und niedrigem  $y$ -Wert die Ausgabe grüner Punkte generieren soll.

### 1.2.2 Unüberwachtes Lernen

Das Ziel des unüberwachten Lernens ist es, innerhalb einer Menge von Eingaben Gruppen anhand ähnlicher Strukturen zu finden. Dafür wird keine Bearbeitung oder Sortierung der Daten benötigt.

Daher ist diese Lernart besonders interessant für die Clusteranalyse, welche dasselbe Ziel anstrebt. Sie spielt besonders im Internet eine Rolle, wo unterschiedliche Zielgruppen unterschiedliche Inhalte, Produktvorschläge oder Werbungen angezeigt bekommen sollen.

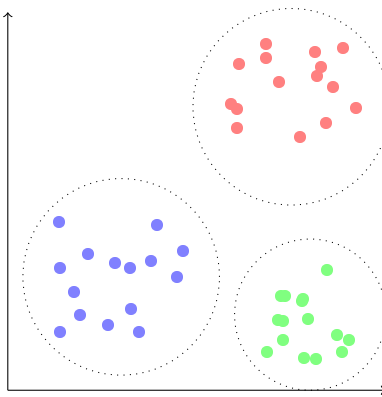


Abbildung 3: Gruppierung von Daten

In Abbildung 3 entsprechen sind alle Punkte unterschiedliche Daten, gleichfarbige Punkte haben jedoch ähnliche Eigenschaften. Das neuronale Netz soll diese Ähnlichkeiten erkennen und die Punkte gleicher Farbe jeweils in die gleiche Kategorie einteilen.

### 1.2.3 Bestärktes Lernen

Bestärktes Lernen dient dazu, den Entscheidungsprozess eines neuronalen Netzes bezüglich einer bestimmten Tätigkeit zu trainieren.

Dabei wird mit Belohnungen gearbeitet: Je besser eine Ausgabe war, umso höher ist die Belohnung für das Netz, was darin resultiert, dass die jeweilige Entscheidung verstärkt wird. Entschied sich das Netz falsch, ist die Belohnung gering und die Entscheidung wird geschwächt. Während des Lernvorgangs wird stets versucht, die Belohnung zu maximieren, um immer die bestmögliche Ausgabe zu erreichen.

Die Belohnungen sind hierbei eine Art Feedback für das Netz, weshalb man auch von einer Art des überwachten Lernens sprechen kann.

### 1.3 Begriffskonventionen

- **Gewichte:** Synapsengewichte zwischen einzelnen Neuronen, welche die Anfälligkeit eines Neurons auf ein vorheriges Neuron steuern
- **Training:** Lernvorgang; Prozess, in dem eine Anpassung der Gewichte des neuronalen Netzes stattfindet

### 1.4 Umgang mit neuronalen Netzen [2]

Vor der Implementation maschinellen Lernens in einem Programm muss ein neuronales Netz entwickelt werden, das die gewünschte Aufgabe zuverlässig erledigen kann. Für die einzelnen Lernarten existieren standardisierte Abläufe, die von Netz zu Netz nahezu gleich sind. Im Folgenden wird der des überwachten Lernens vorgestellt.

#### 1.4.1 Vorbereiten eines Trainingsdatensatzes

Zuerst müssen Daten gesammelt und zu einem Trainingsdatensatz zusammengefügt werden. Bei allen Daten handelt es sich um mögliche Eingaben für das Netz. Sollen also beispielsweise Bilder von unterschiedlichen Objekten unterschieden werden, sind die Daten ebenfalls Bilder, auf denen jeweils eines der zu klassifizierenden Objekte enthalten ist.

Neben dem einfachen Zusammenstellen des Datensatzes kann es jedoch auch vorkommen, dass Daten zuerst bearbeitet werden müssen, sodass einzelne Details auffälliger sind. Das kann geschehen, indem zum Beispiel ein Bild auf die relevanten Pixel reduziert wird. Dadurch lernt das Netz nicht nur besser, sondern auch schneller und mit einer geringeren Datenmenge als würden Daten nicht optimiert werden.

Sämtliche Anpassungen der Daten müssen einheitlich und auch bei der Anwendung des Netzes mit unbekannten Daten erfolgen.

Außerdem ist es für die bevorstehende Auswertung wichtig, dass ebenfalls ein Testdatensatz erstellt wird. Sein Aufbau ist gleich wie der des Trainingsdatensatzes, es sind lediglich andere Daten enthalten.

### **1.4.2 Trainieren**

Mit dem fertigen Trainingsdatensatz kann das neuronale Netz trainiert werden. Dafür existieren unterschiedliche Trainingsalgorithmen, die sich auch zwischen besonderen Aufbauten neuronaler Netze und deren spezifischen Aufgaben unterscheiden können.

Welcher der Algorithmen der Beste für die jeweilige Aufgabe ist, kann nicht pauschal beantwortet werden, sondern es sollte einzeln mit Bezug darauf getestet werden, welcher Algorithmus am besten abschneidet.

### **1.4.3 Auswertung und Anwendung**

Nachdem das Training abgeschlossen wurde, ist es interessant zu wissen, mit welcher Genauigkeit oder Zuverlässigkeit eine Aufgabe ausgeführt wird. Dafür kommt der zuvor erstellte Testdatensatz in Einsatz, mit dem die Anzahl der falschen bzw. unerwünschten Ausgaben ermittelt wird und der somit eine Einschätzung des Netzes möglich macht.

Wenn die Aufgabe zufriedenstellend abgeschlossen wird, werden die Gewichte und die Konfiguration des Netzes in Dateien abgespeichert und in einem Anwendungsprogramm importiert. Das ermöglicht eine Rekonstruktion des neuronalen Netzes ohne großen Zeitaufwand und ohne erneut ein Training durchführen zu müssen.

## **1.5 Programmiersprachen zur Entwicklung neuronaler Netze [2]**

Ein neuronales Netz kann in jeder beliebigen Programmiersprache entwickelt werden. Am beliebtesten ist jedoch Python, da es über eine große Nutzerbasis und viele Bibliotheken in Bezug auf maschinelles Lernen verfügt, insbesondere Google's TensorFlow zum Trainieren tiefer neuronaler Netze.

Ich persönlich bevorzuge jedoch Java, was weniger an Vor- oder Nachteilen liegt, sondern an persönlichen Präferenzen und Programmiererfahrungen.



- 2 Einfache Lernalgorithmen für einzelne Neuronen [3]
- 3 Entwicklung eines neuronalen Netzes [4]
- 4 Bildklassifikation mit tiefen neuronalen Netzen [5]
- 5 Resultate

## Literatur

- [1] CALLAN, ROBERT: *Neuronale Netze im Klartext*. Pearson Studium, 1. Auflage, 2003.
- [2] RASCHKA, SEBASTIAN und VAHID MIRJALILI: *Python Machine Learning*. Packt, 3. Auflage, 2015. Kapitel 1: Giving Computers the Ability to Learn from Data.
- [3] RASCHKA, SEBASTIAN und VAHID MIRJALILI: *Python Machine Learning*. Packt, 3. Auflage, 2015. Kapitel 2: Training Simple Machine Learning Algorithms for Classification.
- [4] RASCHKA, SEBASTIAN und VAHID MIRJALILI: *Python Machine Learning*. Packt, 3. Auflage, 2015. Kapitel 12: Implementing a Multilayer Artificial Neural Network from Scratch.
- [5] RASCHKA, SEBASTIAN und VAHID MIRJALILI: *Python Machine Learning*. Packt, 3. Auflage, 2015. Kapitel 15: Classifying Images with Deep Convolutional Networks.
- [6] SPITZER, MANFRED: *The Mind Within the Net: Models of Learning, Thinking and Acting*, Seite 21. MIT Press, 2. Auflage, 1999.  
<https://books.google.de/books?id=lniTl1DbRX4C&pg=0>, 23.04.2022.