

# Inhalt

Vorwort zur 2. Auflage .....	13
Materialien zum Buch .....	14
Vorwort .....	15

<b>1 Einleitung</b> .....	19
<b>1.1 Wozu neuronale Netze?</b> .....	19
<b>1.2 Über dieses Buch</b> .....	20
<b>1.3 Der Inhalt kompakt</b> .....	22
<b>1.4 Ist diese Biene eine Königin?</b> .....	25
<b>1.5 Ein künstliches neuronales Netz für den Bienenstaat</b> .....	26
<b>1.6 Von der Biologie zum künstlichen Neuron</b> .....	31
1.6.1 Das biologische Neuron und seine technische Kopie .....	32
1.6.2 Das künstliche Neuron und seine Elemente .....	33
<b>1.7 Einordnung und der Rest</b> .....	36
1.7.1 Big Picture .....	36
1.7.2 Artificial Intelligence (künstliche Intelligenz) .....	37
1.7.3 Geschichte .....	38
1.7.4 Machine Learning (maschinelles Lernen) .....	40
1.7.5 Deep Neural Networks .....	41
<b>1.8 Zusammenfassung</b> .....	43
<b>1.9 Referenzen</b> .....	44

## Teil I Up and running

<b>2 Das minimale Starterkit für die Entwicklung von neuronalen Netzen mit Python</b> .....	47
<b>2.1 Die technische Entwicklungsumgebung</b> .....	47
2.1.1 Die Anaconda-Distribution .....	47
2.1.2 Unser Cockpit: Jupyter Notebook .....	52

2.1.3	Wichtige Python-Module .....	64
2.1.4	Jupyter-Notebook-Cloud-Ressourcen .....	66
<b>2.2</b>	<b>Zusammenfassung .....</b>	<b>67</b>
<b>3</b>	<b>Ein einfaches neuronales Netz .....</b>	<b>69</b>
<b>3.1</b>	<b>Vorgeschichte .....</b>	<b>69</b>
<b>3.2</b>	<b>Her mit dem neuronalen Netz! .....</b>	<b>69</b>
<b>3.3</b>	<b>Neuron-Zoom-in .....</b>	<b>73</b>
<b>3.4</b>	<b>Stufenfunktion .....</b>	<b>78</b>
<b>3.5</b>	<b>Perceptron .....</b>	<b>80</b>
<b>3.6</b>	<b>Punkte im Raum – Vektorrepräsentation .....</b>	<b>81</b>
3.6.1	Aufgabe: Werte vervollständigen .....	82
3.6.2	Aufgabe: Iris-Datensatz als Scatter-Plot ausgeben .....	85
<b>3.7</b>	<b>Horizontal und vertikal – Spalten- und Zeilenschreibweise .....</b>	<b>88</b>
3.7.1	Aufgabe: Ermittlung des Skalarprodukts mithilfe von NumPy .....	89
<b>3.8</b>	<b>Die gewichtete Summe .....</b>	<b>91</b>
<b>3.9</b>	<b>Schritt für Schritt – Stufenfunktionen .....</b>	<b>91</b>
<b>3.10</b>	<b>Die gewichtete Summe reloaded .....</b>	<b>92</b>
<b>3.11</b>	<b>Alles zusammen .....</b>	<b>93</b>
<b>3.12</b>	<b>Aufgabe: Roboterschutz .....</b>	<b>96</b>
<b>3.13</b>	<b>Zusammenfassung .....</b>	<b>99</b>
<b>3.14</b>	<b>Referenzen .....</b>	<b>99</b>
<b>4</b>	<b>Lernen im einfachen Netz .....</b>	<b>101</b>
<b>4.1</b>	<b>Vorgeschichte: Man lässt planen .....</b>	<b>101</b>
<b>4.2</b>	<b>Lernen im Python-Code .....</b>	<b>102</b>
<b>4.3</b>	<b>Perceptron-Lernen .....</b>	<b>103</b>
<b>4.4</b>	<b>Trenngerade für einen Lernschritt .....</b>	<b>106</b>
<b>4.5</b>	<b>Perceptron-Lernalgorithmus .....</b>	<b>108</b>
<b>4.6</b>	<b>Die Trenngeraden bzw. Hyperplanes oder auch Hyperebenen für das Beispiel .....</b>	<b>113</b>

<b>4.7</b>	<b>scikit-learn-kompatibler Estimator</b>	116
<b>4.8</b>	<b>scikit-learn-Perceptron-Estimator</b>	123
<b>4.9</b>	<b>Adaline</b>	126
<b>4.10</b>	<b>Zusammenfassung</b>	136
<b>4.11</b>	<b>Referenzen</b>	137
<b>5</b>	<b>Mehrschichtige neuronale Netze</b>	139
<b>5.1</b>	<b>Ein echtes Problem</b>	139
<b>5.2</b>	<b>XOR kann man lösen</b>	141
<b>5.3</b>	<b>Vorbereitungen für den Start</b>	147
<b>5.4</b>	<b>Der Plan für die Umsetzung</b>	149
<b>5.5</b>	<b>Das Setup (»class«)</b>	150
<b>5.6</b>	<b>Die Initialisierung (»__init__«)</b>	152
<b>5.7</b>	<b>Was für zwischendurch (»print«)</b>	154
<b>5.8</b>	<b>Die Auswertung (»predict«)</b>	155
<b>5.9</b>	<b>Die Verwendung</b>	157
<b>5.10</b>	<b>Zusammenfassung</b>	159
<b>6</b>	<b>Lernen im mehrschichtigen Netz</b>	161
<b>6.1</b>	<b>Wie misst man einen Fehler?</b>	161
<b>6.2</b>	<b>Gradientenabstieg an einem Beispiel</b>	163
6.2.1	Gradientenabstieg – die Idee	163
6.2.2	Algorithmus für den Gradientenabstieg	165
<b>6.3</b>	<b>Ein Netz aus sigmoiden Neuronen</b>	172
<b>6.4</b>	<b>Der coole Algorithmus mit Vorwärts-Delta und Rückwärts-Propagation</b>	174
6.4.1	»__init__«-Methode	174
6.4.2	»predict«-Methode	177
6.4.3	»fit«-Methode	182
6.4.4	»plot«-Methode	184
6.4.5	Alles im Konzert	185
<b>6.5</b>	<b>Ein »fit«-Durchlauf</b>	187
6.5.1	Initialisierung	189

6.5.2	Forward .....	190
6.5.3	Output .....	191
6.5.4	Hidden .....	192
6.5.5	Delta $W_{kj}$ .....	194
6.5.6	Delta $W_{ji}$ .....	195
6.5.7	$W_{ji}$ .....	195
6.5.8	$W_{kj}$ .....	195
<b>6.6</b>	<b>Zusammenfassung</b> .....	196
<b>6.7</b>	<b>Referenz</b> .....	196
<b>7</b>	<b>Convolutional Neural Networks</b> .....	197
<b>7.1</b>	<b>Aufbau eines CNN</b> .....	199
<b>7.2</b>	<b>Der Kodierungsblock</b> .....	200
7.2.1	Convolutional Layer .....	200
7.2.2	Activation Function .....	203
7.2.3	Pooling Layer .....	204
7.2.4	Überlappen, ausfüllen und Schrittlänge .....	205
<b>7.3</b>	<b>Der Prädiktionsblock</b> .....	207
7.3.1	Flatten .....	207
7.3.2	Softmax .....	208
<b>7.4</b>	<b>Trainieren von Convolutional Neural Networks</b> .....	209
7.4.1	Das Problem der explodierenden/verschwindenden Gradienten .....	210
7.4.2	Das Optimierungsverfahren .....	214
7.4.3	Verhindern von Overfitting .....	216
<b>7.5</b>	<b>Zusammenfassung</b> .....	218
<b>7.6</b>	<b>Referenzen</b> .....	219
<b>8</b>	<b>Programmierung von Convolutional Neural Networks mit TensorFlow 2</b> .....	221
<b>8.1</b>	<b>Convolutional Networks zur Handschriftenerkennung</b> .....	221
8.1.1	Der Datensatz .....	221
8.1.2	Ein einfaches CNN .....	225
8.1.3	Die Ergebnisse .....	230

<b>8.2</b>	<b>Transfer Learning mit Convolutional Neural Networks</b> .....	237
8.2.1	Das vortrainierte Netzwerk .....	239
8.2.2	Datenvorbereitung .....	240
8.2.3	Das vortrainierte Netz .....	241
8.2.4	Die Ergebnisse .....	244
<b>8.3</b>	<b>Zusammenfassung</b> .....	246
<b>8.4</b>	<b>Referenzen</b> .....	247

## Teil II Deep Dive

<b>9</b>	<b>Vom Hirn zum Netz</b> .....	251
<b>9.1</b>	<b>Ihr Gehirn in Aktion</b> .....	251
<b>9.2</b>	<b>Das Nervensystem</b> .....	252
<b>9.3</b>	<b>Das Gehirn</b> .....	253
9.3.1	Die Teile .....	253
9.3.2	Ein Ausschnitt .....	254
<b>9.4</b>	<b>Neuronen und Gliazellen</b> .....	255
<b>9.5</b>	<b>Eine Übertragung im Detail</b> .....	257
<b>9.6</b>	<b>Darstellung von Zellen und Netzen</b> .....	260
<b>9.7</b>	<b>Zusammenfassung</b> .....	262
<b>9.8</b>	<b>Referenzen</b> .....	263
<b>10</b>	<b>Die Evolution der neuronalen Netze</b> .....	265
<b>10.1</b>	<b>1940er</b> .....	265
10.1.1	1943: McCulloch-Pitts Neurons .....	266
10.1.2	1949: Donald Hebb .....	267
<b>10.2</b>	<b>1950er</b> .....	268
10.2.1	1951: Marvin Minsky und Dean Edmonds – SNARC .....	268
10.2.2	1956: Artificial Intelligence .....	268
10.2.3	1957: Rosenblatts Perceptron .....	268
10.2.4	1959: Bernard Widrow und Marcian Hoff – Adaline und Madaline ....	269
<b>10.3</b>	<b>1960er</b> .....	270
10.3.1	1969: Marvin Minsky und Seymour Papert .....	270

<b>10.4</b>	<b>1970er</b>	270
10.4.1	1972: Kohonen – assoziativer Memory	270
10.4.2	1973: Lighthill Report	270
10.4.3	1974: Backpropagation	271
<b>10.5</b>	<b>1980er</b>	271
10.5.1	1980: Fukushimas Neocognitron	271
10.5.2	1982: John Hopfield	273
10.5.3	1982: Kohonens SOM	283
10.5.4	1986: Backpropagation	284
10.5.5	1987: NN-Konferenz	284
<b>10.6</b>	<b>1990er</b>	284
10.6.1	1997 Sepp Hochreiter und Jürgen Schmidhuber – LSTM	284
<b>10.7</b>	<b>2000er</b>	285
10.7.1	2006: Geoffrey Hinton et al.	285
<b>10.8</b>	<b>2010er</b>	285
10.8.1	2014: Ian J. Goodfellow et al. – Generative Adversarial Networks (GAN)	285
<b>10.9</b>	<b>Zusammenfassung</b>	287
<b>10.10</b>	<b>Referenzen</b>	288
<b>11</b>	<b>Der Machine-Learning-Prozess</b>	289
<b>11.1</b>	<b>Das CRISP-DM-Modell</b>	289
11.1.1	Geschäfts(prozess)-Verständnis	290
11.1.2	Datenverständnis	291
11.1.3	Datenvorbereitung	291
11.1.4	Modellierung	292
11.1.5	Evaluation	292
11.1.6	Einsatz	293
<b>11.2</b>	<b>Feature Engineering</b>	293
11.2.1	Feature-Kodierung	295
11.2.2	Feature-Extraktion	306
11.2.3	Der Fluch der Dimensionalität	317
11.2.4	Feature-Transformation	317
11.2.5	Feature-Auswahl	322

<b>11.3 Zusammenfassung</b> .....	324
<b>11.4 Referenzen</b> .....	324
<b>12 Lernverfahren</b> .....	325
<b>12.1 Lernstrategien</b> .....	325
12.1.1 Überwachtes Lernen (Supervised Learning) .....	326
12.1.2 Unüberwachtes Lernen (Unsupervised Learning) .....	330
12.1.3 Verstärkendes Lernen (Reinforcement Learning) .....	344
12.1.4 Teilüberwachtes Lernen (Semi-supervised Learning) .....	360
<b>12.2 Werkzeuge</b> .....	361
12.2.1 Confusion Matrix .....	361
12.2.2 ROC-Curves .....	363
<b>12.3 Zusammenfassung</b> .....	366
<b>12.4 Referenzen</b> .....	366
<b>13 Anwendungsbereiche und Praxisbeispiele</b> .....	367
<b>13.1 Warmup</b> .....	367
<b>13.2 Bildklassifikation</b> .....	370
13.2.1 Begriffsbestimmung .....	370
13.2.2 Von Bienen und Hummeln .....	372
13.2.3 (Vor-)Trainierte Netze .....	383
<b>13.3 Erträumte Bilder</b> .....	391
13.3.1 Der Algorithmus .....	392
13.3.2 Die Implementierung .....	394
<b>13.4 Zusammenfassung</b> .....	402
<b>13.5 Referenzen</b> .....	402
<b>A Python kompakt</b> .....	403
<b>B Mathematik kompakt</b> .....	433
<b>C TensorFlow 2 und Keras</b> .....	455
Index .....	467