

## Handbuch: 7.2.2. Feed-Forward-Netz

Die meisten neuronalen Netze gehen im Allgemeinen davon aus, dass die Datenpunkte, die zum Training verwendet werden, unabhängige Messungen sind. Das ist ein Schlüsselpunkt beim Modellieren und verdient einige Erläuterungen.

Nehmen wir einmal an, wir haben eine Sammlung digitaler Bilder, die wir in zwei Gruppen unterteilen: Solche, auf denen menschliche Gesichter zu sehen sind, und solche, auf denen irgendetwas anderes zu sehen ist. Neuronale Netze können diese Klassifizierung erlernen, sofern die Sammlung groß genug ist. Die Bilder haben wenig miteinander zu tun; es gibt auch keine Ursache-Wirkungs-Beziehung zwischen irgendwelchen Bildern – jedenfalls keine, die bei der Unterscheidung zwischen menschlichen Gesichtern und anderen Bildern relevant wären.

Aber nehmen wir einmal an, wir wollten Landschaftsbilder in Winterbilder und Sommerbilder klassifizieren und diese Bilder zeigten dieselbe Landschaft, und zwar in kurzer Folge. Nun sind die Bilder nicht mehr unabhängig voneinander, sondern unterliegen einer Ursache-Wirkungs-Beziehung. Das impliziert, dass die Funktionsgleichung  $f(\dots)$ , die wir suchen, ganz anders aufgebaut sein muss. Ihre Genauigkeit wäre viel besser, wenn sie nicht jedes Bild für sich beurteilen würde, sondern wenn sie ein Gedächtnis hätte von den letzten paar Bildern der Serie; denn auf die ersten paar Sommerbilder würden wahrscheinlich weitere Sommerbilder folgen. Bei dieser Version stellen wir fest, dass die Bilder abhängig von einem historischen Verlauf sind, so dass wir vom System ein zeit-orientiertes Gedächtnis erwarten würden, und zwar über eine gewisse Zeitspanne hinweg, die wir irgendwie festlegen müssten.

Wir haben folglich Netz-Modelle, die gut geeignet sind für Datensätze mit unabhängigen Datenpunkten und andere Modelle, die gut geeignet sind für Datensätze, bei denen die Datenpunkte zeitabhängig sind. Netze, die mit unabhängigen Punkten zu tun haben, nennt man Feed-Forward-Netze, und sie bilden nicht nur den historischen Anfang des Feldes der neuronalen Netze ab, sondern auch ihre vorrangigsten Methoden. Die Netze, die sich mit zeitabhängigen Punkten befassen, werden Rekurrente Netze genannt, die moderner, aber schwieriger anzuwenden sind.

Das populärste neuronale Netz nennt man das Mehrschicht-Perceptron, das die folgende Form annimmt:  $y = a_N(W_N \cdot a_{N-1}(W_{N-1} \cdot \dots \cdot a_1(W_1 \cdot x + b_1) \dots + b_{N-1}) + b_N)$  wobei  $N$  für die Anzahl der Schichten steht und  $W_i$  für die Gewichts-Matrizen;  $b_i$  sind Befangenheits-Vektoren,  $\tanh(\dots)$  sind Aktivierungsfunktionen und  $x$  steht wie gehabt für die Eingabegrößen.

Die Gewichts-Matrizen und die Befangenheits-Vektoren sind Platzhalter für die Parameter des Modells. Die sogenannte Topologie des Netzes bezieht sich auf die Freiheit, die wir bei der Auswahl der Matrizengrößen und Vektorengrößen haben, und auch auf die Anzahl der Schichten  $N$ . Die einzige Einschränkung, die wir haben, ist die Problem-inhärente Größe der Eingabe- und Ausgabe-Vektoren. Haben wir uns einmal auf die Topologie festgelegt, hat das Modell eine spezifische Anzahl von Parametern, die in diesen Matrizen und Vektoren residieren.

Um ein solches Netz zu trainieren, müssen wir erst die Topologie des Netzes sowie

die Art der Aktivierungsfunktionen auswählen. Danach haben wir den Wert der Parameter innerhalb der Gewichts-Matrizen und der Befangenheits-Vektoren zu bestimmen. Der erste Schritt ist eine Frage der menschlichen Vorentscheidung und setzt erhebliche Erfahrungen voraus. Selbst nach Jahrzehnten der Forschung auf diesem Gebiet stellen die topologischen Entscheidungen eines neuronalen Netzes immer noch eine nur dunkel verstandene Kunst dar. Es gibt viele Ergebnisse, von denen sich die Praktiker dieser Kunst in ihren Ritualen leiten lassen können, aber das näher zu erläutern würde den Umfang dieses Handbuchs sprengen. Der zweite Schritt kann durch standardisierte Trainingsalgorithmen erreicht werden, die wir zuvor erwähnt haben und die wir hier auch nicht weiter behandeln wollen.

Ein Einzelschicht-Perceptron ist somit  $y = \tanh(Wx + b)$ , und war eines der ersten neuronalen Netze, die man untersucht hat. Das Einzelschicht-Perceptron kann ausschließlich linear trennbare Muster darstellen. Es kann bewiesen werden, dass ein Zwei-Schichten-Perceptron praktisch jede gewünschte Funktion mit nahezu beliebiger Genauigkeit darstellen kann, wenn nur die Gewichts-Matrizen und die Befangenheits-Vektoren jeder Schicht groß genug gewählt wurden.