

K[?]nstliche neuronale Netze

Leopold Eberhart


Der Anaesthetist

Cite this paper

Downloaded from [Academia.edu](#) 

[Get the citation in MLA, APA, or Chicago styles](#)

Related papers

[Download a PDF Pack](#) of the best related papers 



[Vorhersage von [?]belkeit und Erbrechen in der postoperativen Phase durch ein k[?]nstliches n...](#)
Leopold Eberhart

[K[?]nstliche neuronale Netze zur Steuerung von Heimbeatmungsger[?]ten](#)
Ekkehart Paditz

[Künstliche neuronale Netzwerke als Black Box. Verfahren der Explainable AI. Medienwissenschaftlich...](#)
Andreas Sudmann

Redaktion

D. M. Albrecht, Dresden
 E. Martin, Heidelberg

M. Traeger¹ · A. Eberhart² · G. Geldner³ · A. M. Morin³ · C. Putzke³ · H. Wulf³ · L. H. J. Eberhart³

¹ Klinik für Innere Medizin, Kreiskrankenhaus Günzburg

² Institut für Informatik, International University, Bruchsal

³ Universitätsklinik für Anästhesie und Intensivtherapie, Philipps-Universität Marburg

Künstliche neuronale Netze

Theorie und Anwendungen in der Anästhesie, Intensiv- und Notfallmedizin

Grundsätzliche Funktionsbeschreibung eines künstlichen neuronalen Netzwerks

Bei einem künstlichen neuronalen Netzwerk (KNN) handelt es sich um ein einfaches Modell des zentralen Nervensystems. Neurone sind dort hoch gradig untereinander verbunden. Sie lernen, auf Informationen zu reagieren und sich entsprechend anzupassen. Das menschliche Gehirn ist die komplizierteste Struktur, die uns bekannt ist. Umso beachtenswerter ist, dass im Gehirn nur ein Grundtypus von Zelle existiert, der Information übertragen und diese in gewisser Weise auch speichern kann. Beim Menschen entsteht durch dieses Netzwerk aus etwa 10^{11} Neuronen ein intelligentes Wesen mit den Fähigkeiten zu lernen, wiederzuerkennen und vorauszusehen. Mit neuronalen Netzwerken versucht man die Vorgänge im Gehirn durch Computeranwendungen zu simulieren. Im Gehirn sind die Nervenzellen in komplexer Weise miteinander verknüpft. Jede Zelle kann über Dendriten Signale von anderen Zellen empfangen und diese über Axone weiterleiten. Im Zellkörper der Nervenzelle summieren sich die Potenziale der einzelnen Dendriten auf. Wird ein bestimmter Schwellenwert überschritten, dann feuert das Neuron und sendet so ein Impulssignal über das Axon zu seinen Nachbarn; andernfalls bleibt es ruhig. Rein technisch gesehen funktioniert also eine Nervenzelle

verblüffend einfach und das zugrunde liegende Prinzip kann in einem KNN grundsätzlich imitiert werden.

Entscheidend für die enorme Leistungsfähigkeit des Gehirns sind zum einen die große Anzahl an Nervenzellen und zum anderen die extrem hohe Anzahl an Verknüpfungen (schätzungsweise 10^{14} Verbindungen). In einem neuronalen Netzwerk sind derzeit viel weniger Verknüpfungen (wenige 10.000) möglich. Eingabe- („input node“), innere („hidden node“) und Ausgabeknoten („output node“) sind meist in Schichten strukturiert und miteinander vernetzt. Zwischen dem Eingang und dem Ausgang des Netzwerks können ein bis mehrere Schichten innerer Knoten liegen – zur Vereinfachung wird in der Illustration nur eine Schicht innerer Knoten gezeigt. Dem Nervensystem ähnlich gehen die Signale in einem neuronalen Netzwerk zunächst bei den Eingabeknoten ein und werden an die inneren Knoten weitergeleitet. Anhand des bei ihm eingehenden Signals führt jeder innere Knoten Berechnungen durch und sendet das Resultat an die nächste Stelle weiter.

Der mathematische Modellansatz, ein Neuron als einen Summiervverstärker mit individueller Gewichtung seiner Eingänge zu verstehen, ist durchaus nahe liegend. Erste Ansätze zur Beschreibung neuronaler Netze wurden bereits 1943 von McCulloch u. Pitts gemacht. Im Modell werden Neuronen auch *Knoten* genannt. Das

etwas komplizierte Übertragungsschema von Axonen und Dendriten wird durch Verbindungen mit variablen Gewichten ersetzt. Die Signale eines Eingabeknoten zu einem Neuron werden jeweils mit einem variablen spezifischen Gewicht multipliziert. Die Berechnungsergebnisse der inneren Knoten muss man sich als eine Art Zwischenresultat vorstellen, die für die Arbeitsweise des Netzwerks entscheidend sind. Ab welchem Zwischenergebnis dann ein künstliches Neuron nach dem „Alles-oder-Nichts-Prinzip“ feuert, hängt von der verwendeten Aktivierungsfunktion sowie einem individuellen Schwellenwert ab.

Aufbau eines künstlichen neuronalen Netzwerks

Es gibt sehr viele Typen von KNN, aber alle lassen sich durch 3 Einheiten definieren: durch die einzelnen Neuronen, deren Verbindungen (Architektur oder Topologie) und die Lernregel innerhalb des Netzes [11, 19].

Neurone in einem künstlichen neuronalen Netzwerk

Die Neurone bilden die Grundbausteine eines KNN. Wie bereits angedeutet, kann ein Neuron durch seinen Aktivierungszustand zu einem bestimmten Zeitpunkt beschrieben werden. Darüber hinaus legt eine Aktivierungsfunktion fest, wie der

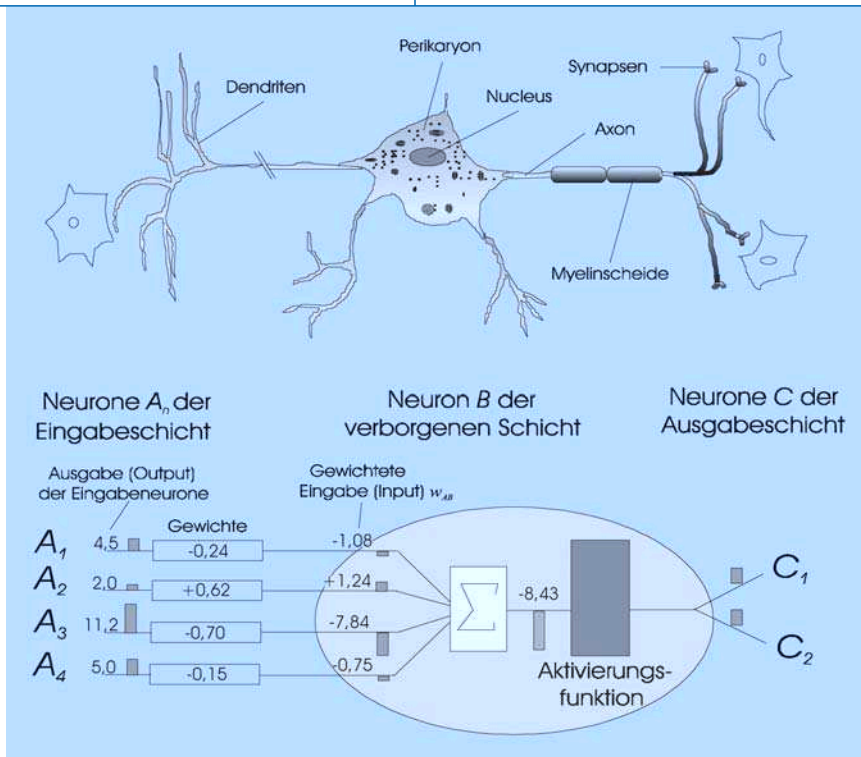


Abb. 1 ▲ Schematischer Aufbau eines Neurons in einem KNN, analog einer Nervenzelle (im Bild oben). Die Eingabe des Neurons entspricht dem Informationsfluss in den Dendriten, die Ausgabe der Signalweiterleitung im Axon. Im Neuron werden die ankommenden Impulse aufsummiert. Durch eine Aktivierungsfunktion ist festgelegt, ab welchem Schwellenwert eine Ausgabe erfolgen soll. Vor und nach dem Neuron sind weitere Einheiten geschaltet (A₁–A₄ bzw. C₁ und C₂). Die Werte und Gewichte sind Beispiele

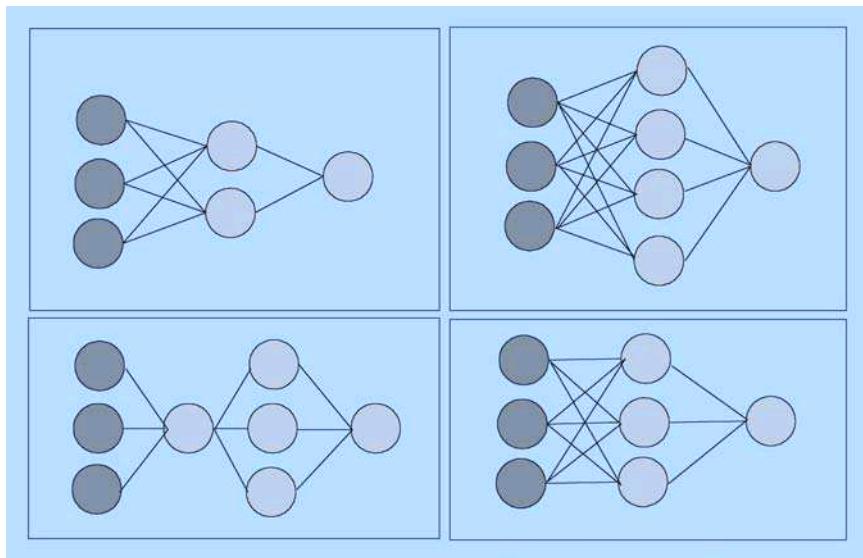


Abb. 2 ▲ Vier Beispiele für einfache Netzwerkarchitekturen. Links befinden sich die Eingabeneurone, in der Mitte die Neurone der versteckten Schicht („hidden layer“) und rechts das Neuron, das die Antwort des KNN repräsentiert

neue Zustand sein wird, der sich wiederum aus der bereits bestehenden Aktivierung und den aktuell einfließenden neuen Informationen über die Dendriten ergibt. Eine Ausgabefunktion legt fest, ob

die aufsummierten Reize zu einer Aktivierung des Neurons führen.

Die Verbindungen zwischen den einzelnen Neuronen, die sich im Laufe eines Trainingsprozesses herausbilden, sind un-

terschiedlich stark und werden als Gewichte bezeichnet. Die Summe der Eingaben an einem Neuron wird aus den Ausgaben der vorgeschalteten Neurone und den jeweiligen Verbindungsgewichtungen ermittelt und ergibt die Aktivierungsfunktion. Das Neuron wird dann seinerseits ein Signal weitergeben, wenn ein bestimmter Schwellenwert überschritten wird. Bei den Aktivierungsfunktionen handelt es sich i. Allg. um Sigmoid- oder Tangens-hyperbolicus-Funktionen, die bewirken, dass eine geringe Erregung des Neurons zu keiner Entladung führt, andererseits ab einer kritischen Grenze bereits eine leichte Zunahme des Inputs eine Aktivität des Neurons auslöst. ■ Abb. 1 zeigt schematisch den Aufbau eines Neurons.

Topologie eines künstlichen neuronalen Netzwerks

Ein KNN baut sich aus zahlreichen Neuronen und deren Verbindungen auf. Man unterscheidet 3 Arten von Neuronen: Eingabe-, Ausgabe- und versteckte Neuronen. Eingabeneurone erhalten die externen Reize bzw. Variablen, die verarbeitet werden sollen. Ausgabeneurone zeigen uns die Ergebnisse. Dazwischen liegen die verborgenen Zellen, die sich weitgehend der Beobachtung entziehen, da sie nur interne Impulse empfangen, verrechnen und weitergeben. Die Anzahl der Eingabeneurone entspricht in der Regel der Zahl der Eingabevariablen. Für den Aktivierungszustand dieser Neurone wählt man Werte aus einem kontinuierlichen Bereich (z. B. 0–1) [12]. Während dichotome Variablen (z. B. trifft zu/trifft nicht zu) dadurch gut und auch intuitiv nachvollziehbar abgebildet werden können, müssen kontinuierliche Werte (z. B. das Alter eines Patienten, Dauer der Narkose) erst durch eine z-Transformation in diesen Bereich normalisiert werden. Andernfalls würden kontinuierliche Variablen, die unter Umständen hohe Werte annehmen können, wesentlich stärker berücksichtigt werden als Variablen, deren Ja-Nein-Informationsgehalt mit 1–0 kodiert ist. Qualitative (nominale) Eingangsvariablen, z. B. braune Augenfarbe eines Patienten, können dagegen nur durch mehrere Eingabeneurone mit jeweils nur einer di-

chotomen Information (z. B. braune Augen = 1; blaue Augen = 0; grüne Augen = 0) abgebildet werden.

Die Neurone der Eingabeebene leiten ihre Informationen an die Neurone mindestens einer verborgenen Schicht weiter. Jedes Neuron einer Schicht ist mit jedem Neuron der nächsten Ebene strukturell verknüpft. Funktionell kann eine Verbindung zumindest temporär unterbrochen sein, wenn z. B. die Gewichtung einer Verbindung genau Null ist. **Abb. 2** zeigt vereinfacht den Aufbau einiger einfacher Netzwerke.

Die Problemstellung beeinflusst wesentlich die Zahl der Schichten und die der gesamten Neuronenzahl. Diese sollte so gewählt werden, dass sie einerseits groß genug ist, um ein tolerables Ergebnis zu erzielen, andererseits wiederum nicht zu groß, um die Generalisierungsfähigkeit nicht zu beeinträchtigen. Eine Faustregel besagt, dass für die Zahl der versteckten Neurone die Hälfte der Zahl der addierten Eingangs- und Ausgangsneurone eine gute Performance liefert [11]. Ein deterministischer Ansatz zur Ermittlung der optimalen Netzwerkstruktur existiert aber bislang noch nicht, sodass in der Praxis stets mehrere Netzwerktopologien getestet werden müssen.

Lernregeln innerhalb eines künstlichen neuronalen Netzwerks

Mit Hilfe einer Lernregel versucht das KNN, eine Gewichtungsmatrix (Summe aller Gewichtungen) zu finden, bei der das gewünschte Ergebnis mit einem tolerablen Fehler realisiert wird. Das Lernen erfolgt in neuronalen Netzen durch Modifikation der Verbindungsgewichte im Rahmen der wiederholten Präsentation verschiedener Beispielmuster. Ziel des Trainingsvorgangs ist es, den Fehler zwischen erwarteter Ausgabe und tatsächlicher Ausgabe sukzessive zu minimieren. Als Maß für den gesamten Fehler der Neuronenmatrix dient üblicherweise der mittlere quadratische Fehler [19].

Grundlage für das Lernverfahren eines KNN ist die Hebb-Lernregel, die besagt, dass 2 Zellen, die gleichzeitig aktiviert sind, eine Verstärkung ihrer Verbindung (bzw. deren Gewichtung) erfahren. Das Backpropagation-Lernverfahren stellt

Anaesthesist 2003 · 52:1055–1061
DOI 10.1007/s00101-003-0576-x
© Springer-Verlag 2003

M. Traeger · A. Eberhart · G. Geldner · A. M. Morin · C. Putzke · H. Wulf · L. H. J. Eberhart

Künstliche neuronale Netze. Theorie und Anwendungen in der Anästhesie, Intensiv- und Notfallmedizin

Zusammenfassung

Mit einem künstlichen neuronalen Netzwerk (KNN) wird versucht, die Vorgänge im Zentralnervensystem (ZNS) höherer Lebewesen zu simulieren. In aller Regel erfolgt dies durch eine spezielle Software, die das Verhalten einzelner Neurone und deren Interaktion miteinander simuliert. Der wesentliche Unterschied zu klassischen statistischen Berechnungsverfahren besteht in der Lernfähigkeit eines KNN. Das bedeutet, dass ein solches System anfänglich keinerlei Informationen enthält, sondern sich diese erst aus einer gewissen Zahl bekannter Beispiele extrahiert. Im Idealfall kann ein KNN durch wiederholtes Training generalisieren, d. h., es verändert die Verknüpfungen innerhalb des Neuronenverbandes so, dass es später unbekannte Daten anhand der erlernten Regeln richtig klassifizieren kann. Darüber hinaus reagieren neuronale Netze wenig empfindlich gegen gestörte oder unvollständige

Daten. Künstliche neuronale Netze haben bereits in anderen Bereichen gezeigt, dass sie zur Vorhersage von Ereignissen und Modellierung komplexer zeitabhängiger Systeme geeignet sind. Außerhalb der Medizin werden sie v. a. dann eingesetzt, wenn die Einflussfaktoren für einen bestimmten Ausgang nicht oder nur unvollständig bekannt und die Zusammenhänge komplex und nichtlinear sind (z. B. bei Finanz- oder Wetterprognosen). Dieser Artikel soll eine kurze Übersicht über die grundsätzliche Funktionsweise von KNN geben und potenzielle Einsatzmöglichkeiten in Anästhesie, Intensiv- und Notfallmedizin aufzeigen.

Schlüsselwörter

Künstliches neuronales Netz · Künstliche Intelligenz · Vorhersagesysteme

Artificial neural networks. Theory and applications in anaesthesia, intensive care and emergency medicine

Summary

Artificial neural networks (ANN) are constructed to simulate processes of the central nervous system of higher creatures. An ANN consists of a set of processing units (nodes) which simulate neurons and are interconnected via a set of „weights“ (analogous to synaptic connections in the nervous system) in a way which allows signals to travel through the network in parallel. The nodes (neurons) are simple computing elements. They accumulate input from other neurons by means of a weighted sum. If a certain threshold is reached the neuron sends information to all other connected neurons otherwise it remains quiescent. One major difference compared with traditional statistical or rule-based systems is the learning aptitude of an ANN. At the very beginning of a training process an ANN contains no explicit information. Then a large number of cases with a known outcome are presented to the system and the weights of the inter-neuronal connections are changed by a training algorithm designed to

minimise the total error of the system. A trained network has extracted rules that are represented by the matrix of the weights between the neurons. This feature is called generalisation and allows the ANN to predict cases that have never been presented to the system before. Artificial neural networks have shown to be useful predicting various events. Especially complex, non-linear, and time depending relationships can be modelled and forecasted. Furthermore an ANN can be used when the influencing variables on a certain event are not exactly known as it is the case in financial or weather forecasts. This article aims to give a short overview on the function of ANN and their previous use and possible future applications in anaesthesia, intensive care, and emergency medicine.

Keywords

Artificial neural networks · Artificial intelligence · Predictive models

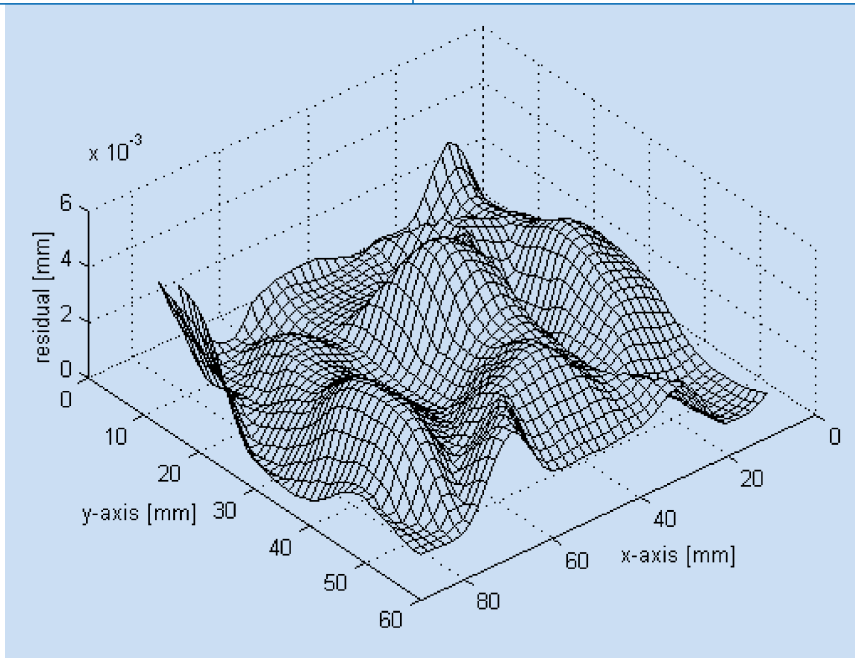


Abb. 3 ▲ Veranschaulichung des Lern- und Generalisierungsverhalten eines KNN. Die statistischen Verteilungen einzelner Variablen erzeugen im mehrdimensionalen Raum eine virtuelle Gebirgslandschaft. Im Laufe des Trainingsprozesses bewegt sich ein Ball, der in diese Landschaft geworfen wird, immer weiter nach unten (das gesamte KNN reduziert den gesamten Fehler) bis er in einem Tal zum Stillstand kommt

eine Verallgemeinerung dieser Regel dar. Bei dieser Methode wird während des Trainings ein *Korrektursignal* rückwärts durch die Schichten geführt, um den Ausgabefehler zu minimieren. Das Korrektursignal wird dabei zunächst an die verborgene Schicht zurückgeleitet, und die entsprechenden Gewichtungen werden verändert. Dann wird der Fehler der nächstfolgenden Schicht gemeldet. So wird bis zur Eingabeschicht verfahren [12]. Auch die Natur bevorzugt dieses rückgekoppelte (Feedback-)Netz, bei dem das Ausgangssignal eines Neurons über mehrere Zwischenneurone als Eingabesignal erneut auf das Ausgangsneuron rückwirken kann. Die Vorteile dieses Typs für die sensomotorische und vegetative Steuerung im Sinne eines Regelkreises leuchten unmittelbar ein.

Training künstlicher neuronaler Netzwerke

Zu Beginn des Trainings eines KNN ist der Informationsgehalt des Systems gleich Null. Es werden also von Seiten des „Trainers“ keine festen Regeln oder bereits ein bestimmtes Wissen vorgegeben. Die Gewichte zwischen den Neuronen werden zufällig zugeteilt. Nun werden dem Netz

sukzessive verschiedene bekannte Fälle (z. B. Patientendaten) angeboten. Nach jedem trainierten Fall werden die Gewichte durch ein Korrektursignal so angepasst, dass der gesamte Fehler des Netzwerks Schritt für Schritt minimiert wird. Eine Lernkonstante oder Lernrate gibt dabei den Betrag an, mit dem die Synapsengewichte in die korrekte Richtung verändert werden. Auch hier ist die Wahl einer geeigneten Konstante entscheidend für ein erfolgreiches Training. Wird sie z. B. zu groß gewählt, dann passen sich die Gewichte innerhalb des KNN in jeweils nur einem einzigen Schritt so an, dass das Ergebnis immer für den im Moment angebotenen Beispieldatensatz optimiert ist. Enthält dann zufälligerweise der darauf folgende Datensatz eine widersprüchliche Information, würden alle bislang erlernten Zusammenhänge auf einmal umgeworfen und die Gewichte sprängen so im ungünstigsten Fall stets zwischen 2 Extremen hin und her, ohne dass das KNN den richtigen „Mittelwert“ aus allen angebotenen Informationen herausfiltern könnte. Das KNN kann in einem solchen Fall nicht generalisieren, sondern sich nur einzelne Muster merken [11, 19]. Eine zu niedrige Lernrate verzögert andererseits den Lernprozess und der Vorrat an Beispieldaten

kann erschöpft sein, bevor das KNN ausreichend Informationen aus den angebotenen Daten extrahieren konnte.

Probleme beim Training neuronaler Netzwerke

Da einem KNN natürlich nie alle Patienten einer Grundgesamtheit zum Training angeboten werden können, muss das Netz zwischen den einzelnen Punkten aus der Trainingsmenge interpolieren. Der Backpropagation-Algorithmus kann aber nicht garantieren, die optimale Interpolation zu finden. Das Generalisierungsverhalten ist zwar meist sehr gut, zeigt jedoch eine charakteristische Besonderheit. Prüft man während des Trainings mit einer unabhängigen Testmenge kontinuierlich den Generalisierungsfehler, so erreicht dieser irgendwann ein Minimum, an dem das Netz am besten generalisiert. Anschließend steigt der Fehler für die unabhängigen Fälle aber oft wieder an, obwohl sich der Fehler für die Trainingsmenge weiter verringert. Dieser Effekt wird „overfitting“ genannt und kann dann auftreten, wenn ein zu kleiner Trainingsdatensatz zu genau gelernt wird, also zu oft präsentiert wird oder auch bei einer zu großen Zahl von Neuronen. Die einzelnen Muster werden dann nur exakt gelernt, so dass das Resultat während des Trainings sehr gut ist. Bei der Verwendung von unbekannten Daten werden dann allerdings nur unbefriedigende Ergebnisse erzielt, weil das KNN nicht generalisiert hat, sondern sich nur frühere Muster gemerkt hat. Daher muss auch der Testdatensatz ausreichend groß sein. Erfahrungsgemäß soll dieser mindestens zehnmal so viele Beispiele enthalten wie das trainierte Netz Verbindungen aufweist. Um Overfitting zu vermeiden ist es günstig, während des Trainings regelmäßig das KNN mit dem Validationsdatensatz zu überprüfen, um das Training am optimalen Punkt abbrechen zu können [19].

Theoretisch ist das Generalisierungsverhalten noch nicht gut verstanden. Ein allgemein akzeptiertes und recht anschauliches Modell ist es, die statistischen Verteilungen von Variablen in mehrdimensionalen Räumen als konvexe Formen zu betrachten. Die statistischen Fluktuationen in diesen Konturen bilden kleinere Inseln und Einbuchtungen (■ Abb. 3).

Stellt man sich das untrainierte Netz als Kugel auf einer Erhebung dieser Verteilungslandschaft vor, dann rollt diese angestoßen durch die Interpolation aus den ersten angebotenen Beispieldatensätzen entlang der konvexen Konturen der Verteilung abwärts. Mit fortschreitender Lerndauer werden aber auch die statistischen Fluktuationen erlernt, und die Generalisierung beginnt sich zu verschlechtern. Bildlich gesprochen rollt ein Ball von einem Berg in ein Tal, ohne aber die nächste kleinere Anhöhe überwinden zu können, um in ein noch tieferes Tal zu gelangen. Das Netzwerk bleibt bildlich gesprochen in einem „lokalen Minimum“ hängen.

Ein weiteres Problem besteht durch eine überlange Trainingsdauer. Ein Hinzufügen von Neuronen ist verbunden mit einem Anstieg der Trainingsdauer in der 3. Potenz [11]. Deshalb sollte die Zahl der Eingabevariablen, z. B. durch Vorselektion der Eingabeparameter mit klassischen statistischen Verfahren, auf ein vernünftiges Maß reduziert werden.

Praktischer Ablauf beim Training eines künstlichen neuronalen Netzwerks

In der Praxis wird zum Training eines KNN der zur Verfügung stehende Gesamtbestand an bekannten Mustern in 3 Mengen aufgeteilt:

- Der Trainingsdatensatz ist zum Erlernen der optimalen internen Gewichtungen nötig. Den Vorhersagefehler auf diesem Datensatz gilt es zu minimieren. Der Datensatz wird dem KNN dabei mehrfach präsentiert.
- Der Validationsdatensatz dient noch nicht, wie der Name suggeriert, zur Überprüfung der Leistung des KNN. Stattdessen wird damit lediglich das Training überwacht, um diesen Vorgang bei Erreichen einer optimalen Generalisierung zu stoppen. Die Daten des Validierungsdatensatzes selbst haben keinen Einfluss auf das Lernen.
- Nach dem abgeschlossenen Lernvorgang wird die Vorhersage-Performance des Netzwerks auf unbekannte Daten getestet. Diese unbekannten Daten bilden das Testset [11, 12].

Vorteile und Nachteile der Anwendung eines künstlichen neuronalen Netzwerks

Wie bereits angedeutet, besteht ein wesentlicher Vorteil bei einem neuronalen Netzwerk darin, dass dem System keine Vorgaben im Sinne vorgeformter Regeln gemacht werden müssen. Vielmehr extrahiert das System selbständig die Informationen angebotener Fallbeispiele und generiert selbstständig eine Lösung. Dabei kann ein KNN auch sehr gut nichtlineare Zusammenhänge modellieren und komplexe Zusammenhänge zwischen verschiedenen Eingabevariablen entdecken. Gerade diese beiden letztgenannten Eigenschaften bilden einen wesentlichen Vorteil gegenüber klassischen mathematischen Verfahren, wie z. B. einer Regressionsanalyse. Andererseits lassen sich, bedingt durch die multiplen Verknüpfungen der Neuronen untereinander, die Regeln, die das KNN aus den Beispieldatensätzen generiert hat, nur sehr schwer ableiten. Ein KNN bleibt somit eine „black box“, und Ergebnisse, die damit generiert werden, können nur sehr bedingt nachvollzogen werden. Somit lässt sich, anders als bei einer Regressionsanalyse, der relative Einfluss eines Eingabefaktors auf das Endergebnis nicht genau bestimmen.

Darüber hinaus kann man bei einem KNN niemals sicher sein, ob die gewählten Netzwerkeinstellungen tatsächlich optimal sind. Zwar gibt es Ansätze, um gute Einstellungen einzugrenzen [4], dennoch ist die Zahl der möglichen Einstellungen so groß, dass es denkbar und sogar wahrscheinlich ist, dass noch bessere Netzwerkeinstellungen existieren. Dies kann sowohl für die Netzwerktopologie, für die Lernfunktion („backpropagation“ oder „resilient propagation“) als auch für die Werte der Lernparameter zutreffen. ■ **Tabelle 1** listet wesentliche Vor- und Nachteile von KNN auf.

Einsatzmöglichkeiten von künstlichen neuronalen Netzwerken in Anästhesie, Intensiv- und Notfallmedizin

Künstliche neuronale Netzwerke werden seit etlichen Jahren in der Anästhesie sowie in der Intensiv- und Notfallmedizin

für die verschiedensten Indikationen eingesetzt. Eine Literaturrecherche in der Datenbank „Medline“ ergibt z. B. unter dem Suchprofil: („Critical Care“ [MESH] OR „Anesthesia and Analgesia“ [MESH] OR „emergency medicine“ [MESH]) AND „Neural Networks (Computer)“ [MESH] 44 Publikationen seit dem Jahr 1992. Der folgende Abschnitt erhebt nicht den Anspruch, eine vollständige Übersicht über sämtliche Studien zu geben. Vielmehr soll sich der Leser einen Eindruck verschaffen, welche Fragestellungen bislang mit dieser Methode bearbeitet wurden.

Anästhesie

Eine typische Anwendung von KNN in der Anästhesie stellt die Entwicklung von regelbasierten Infusionssystemen dar. Da die pharmakologische Wirkung von Muskelrelaxanzien durch die Relaxometrie sehr genau quantifiziert werden kann, wurden für diese Substanzklasse mehrere KNN-basierte Steuersysteme entwickelt.

Bei Verwendung mittellang wirkender Muskelrelaxanzien, wie Vecuronium, konnte gezeigt werden, dass der mittlere Fehler bei der Vorhersage im Steady-State lediglich knapp 6% betrug [8]. In eine neuere Studie wurde das kurz wirkende Mivacurium mit Hilfe eines KNN dosiert [7]. Bei Patienten mit größeren abdominalchirurgischen Eingriffen war das eingesetzte KNN nach einer Trainingsphase bei 30 Patienten in der Lage, die kontinuierliche Mivacuriuminfusion über den gesamten Zeitraum der Operation auf einen vorgegebenen Sollwert hin zu regeln. Das Quadrat des gemittelten Prädiktionsfehlers für den jeweils nächsten T1-Wert bei 20 Patienten des Testsets lag bei nur 0,1%.

Durch den Einsatz von KNN kann darüber hinaus auch die Erholung des neuromuskulären Blocks sehr genau vorhergesagt werden. In einer jüngst erschienenen Arbeit war das verwendete KNN einer Vorhersage basierend auf klassischen statistischen Verfahren oder gar einer rein klinischen Beurteilung überlegen [14]. Weiterhin wurde untersucht, ob ein KNN nach Antagonisierung des neuromuskulären Blocks eine bei der Extubation des Patienten noch bestehende Relaxanzienwirkung vorhersagen kann. Dem Netz

Tabelle 1

Vorteile und Nachteile eines künstlichen neuronalen Netzwerkes [1, 3, 5, 19]

Vorteile	Nachteile
Lernfähigkeit und Generalisierungsfähigkeit anhand von Beispielen und Mustern	Probleme im Training („overfitting“) Bei komplexen Problemen u. U. sehr lange Rechendauer (bis zu Tagen)
Kein Vorwissen über bestehende Zusammenhänge erforderlich	Keine mathematische Präzision bei Ergebnisprognosen
Gute Modulierung nichtlinearer Zusammenhänge Gute Modellierung beliebiger Interaktionen zweier Variablen	Kausale Beziehungen und logische Schlussfolgerungen nur sehr bedingt formulierbar
Anpassungsfähigkeit (bei Änderungen des beobachteten Prozesses können jederzeit neue Daten präsentiert werden)	Daten der Test- und der Anwendungsphase müssen sich ähneln
Robustheit (verzerrte oder unvollständige Daten können mit akzeptabler Genauigkeit verarbeitet werden)	Schwierige Implementation in die klinische Praxis Ergebnisse eines KNN haben „Black-box-Charakter“, sodass die Ergebnisse von Anwendern evtl. nur bedingt akzeptiert werden

KNN Künstliches neuronales Netzwerk

wurden dazu die Reizantwort des Patienten vor der Antagonisierung sowie die Zeit zwischen der Neostigmingabe und der Extubation angeboten. Während die narkeoseführenden Anästhesisten immer davon ausgingen, dass nach der Applikation von Neostigmin keine relevante Relaxationswirkung mehr vorlag, prognostizierte das KNN 25 der 26 Patienten korrekt, bei denen die Train-of-four (TOF-)Ratio noch unter 0,7 lag [10].

Notfallmedizin

In einigen Untersuchungen wurde versucht, ein KNN zur Vorhersage von Myokardinfarkten heranzuziehen [1, 2, 3, 9]. Zum einen sollte ermittelt werden, welche Variablen hierzu relevant sind. Der Einfluss eines einzelnen Risikofaktors ist wegen der Nichtlinearität und Komplexität eines KNN, wie bereits erwähnt, nicht sofort ersichtlich. Um den Einfluss doch zu erhalten, wurde folgendes Verfahren angewendet: Eine Eingabevariable nach der anderen wurde in jedem der 706 Muster in eine Richtung verändert, während die anderen Variablen zunächst konstant blieben. Baxt nannte dies „stepwise perturbation“. Die Methode enthüllte, dass das KNN zur Vorhersage v. a. die Variablen EKG-Veränderungen, Synkopen, Halsve-

nenstau, Ansprechen auf Nitroglyzerin oder Übelkeit/Erbrechen verwendete. Mit dem Modell wurden eine Sensitivität und Spezifität von je 97% erreicht [2].

Dieses Netzwerk wurde in einer weiteren Studie mit retrospektiven Daten von 331 Patienten mit Brustschmerzen trainiert und sollte Myokardinfarkte bei diesen Patienten erkennen. Von 331 Patienten hatten 36 tatsächlich einen Infarkt. Das KNN prognostizierte dies mit einer Sensitivität von 97% und einer Spezifität von 96%. Zum Vergleich wurde die Prognose von erfahrenen Ärzten angeführt, die die gleichen Informationen erhalten hatten wie das Netzwerk. Hier lagen die Sensitivität bei lediglich 78% und die Spezifität bei 85%. Darüber hinaus konnte das KNN auch mit fehlenden oder ungenauen Daten arbeiten.

In Fortführung dieser Untersuchungen wurde jüngst eine weitere Studie vorgestellt, bei der nun nicht mehr nur retrospektive Daten als Testdatensatz verwendet wurden, sondern das KNN prospektiv mit aktuellen Fällen aus der Notaufnahme konfrontiert wurde [3]. Das Netzwerk war mit den Datensätzen von über 2.000 Patienten mit Brustschmerzen trainiert. An 128 Patienten mit Myokardinfarkt wurde dann die Kliniktauglichkeit des KNN getestet. Es wurden 121 Patienten korrekt

klassifiziert, obwohl immerhin 5% der Daten fehlten. Hierbei wurden 40 Variablen zum Training verwendet. Es wurden eine Sensitivität von 94,5% und eine Spezifität von 95,9% erzielt.

In einer weiteren Untersuchung [9] erzielte ein ebenfalls zur Herzinfarktdiagnostik entworfenes KNN deutlich bessere Vorhersagewerte als traditionelle Regressionsanalysen.

Intensivmedizin

Ein Schwerpunkt beim Einsatz von Netzwerken in der Intensivmedizin lag bisher bei der Vorhersage der Länge von Intensivaufenthalten, v. a. nach koronarchirurgischen Eingriffen [16, 17]. Eine solche Prognose ist zur besseren Steuerung der Auslastung v. a. dann interessant, wenn die zur Verfügung stehende Intensivkapazität limitiert ist.

Hierzu wurde ein KNN mit den Daten von 713 Patienten trainiert und mit 691 Datensätzen getestet [16]. Es wurden 15 Variablen, die potenziell relevant für einen verlängerten Aufenthalt sind (u. a. das Alter, die linksventrikuläre Funktion, Art und Dringlichkeit des Eingriffes sowie Begleiterkrankungen), zum Training mit einem Backpropagation-Algorithmus verwendet. Bei der Vorhersage, ob ein Patient länger als 2 Tage auf der Intensivstation verbleiben musste, erzielte das KNN einen Wert von 0,71 als Fläche unter einer ROC-Kurve (ROC = „receiver-operating characteristic“), während ein Vorhersagescore, der auf den Ergebnissen einer logistischen Regressionsanalyse von lediglich 5 Variablen beruhte, einen nur unwesentlich geringeren Wert von 0,69 aufwies.

Die Fläche unter einer ROC-Kurve gibt dabei die Wahrscheinlichkeit an, mit der 2 Patienten, von denen einer ein bestimmtes Ereignis aufweist, richtig klassifiziert werden. Eine rein willkürliche Zuteilung hat somit eine Trefferchance von 50%, so dass AUC-Werte (AUC = Fläche unter der Kurve) über 0,5 (bis maximal 1,0) eine gegenüber dem Zufall verbesserte Prognose eines Vorhersagesystems bedeutet.

Ebell et al. verglichen retrospektiv die Prognose eines KNN für das Überleben nach 218 innerklinischen kardiopulmonalen Reanimationen [5] mit der Vorhersage des bestehenden Scores (PAR =

„prognosis after resuscitation“). Letzterer verwendet 8 Variablen, das KNN insgesamt 24. Dabei war die Prognose durch das KNN etwas besser: Die Flächen unter der ROC-Kurve lagen mit 0,77 über der des PAR-Scores mit 0,72.

Auch die Vorhersagbarkeit der Mortalität von Intensivpatienten wurde mit Hilfe von KNN untersucht [4, 13, 18]. Eine große Untersuchung mit den Daten von 114.000 Patienten verglich die Vorhersageleistung eines KNN mit der des Injury Severity Score (ISS). Das KNN wurde mit den 17 Variablen des ISS und den ICD-9-Diagnosen der Patienten trainiert und erreichte eine etwas bessere Performance als der modifizierte ISS.

Auch andere in der Intensivmedizin übliche Scoringssysteme (APACHE II, SAPS II) wurden mit den Ergebnissen von neuronalen Netzen verglichen. Hierzu wurde ein fünfschichtiges Netzwerk mit 168 Datensätzen trainiert und mit weiteren 90 Mustern getestet [4]. Die Resultate der künstlichen neuronalen Netzwerke wurden mit dem Ergebnis einer logistischen Regressionsanalyse verglichen. Diese erreichte eine Fläche unter der ROC-Kurve von 0,753; dagegen erzielte das KNN Werte zwischen 0,753 und 0,846. Wong u. Young [18] verglichen die Fähigkeit eines KNN, die Mortalität der Patienten in einer Intensivstation vorherzusagen, mit der des APACHE II. Es wurden dieselben Variablen verwendet. Die Gesamtleistung war bei beiden Modellen ähnlich gut. Das Netzwerk war nur in einigen Teilaspekten dem APACHE II überlegen. Dass ein KNN aber nicht zwangsläufig besser abschneidet als klassische statistische Verfahren zeigt die Untersuchung von Flanagan et al. [6]. Hier schnitt das KNN mit einem Backpropagation-Lernverfahren bei der Überlebensprognose von 173 Patienten im Multiorganversagen schlechter ab als eine logistische Regressionsanalyse. Das KNN zeigte zwar eine gering höhere Spezifität als die logistische Regressionsanalyse, Letztere war jedoch deutlich sensitiver als das Netzwerk.

Fazit für die Praxis

Künstliche neuronale Netzwerke eignen sich für Prognosen und Klassifizierungen, insbesondere wenn die abhängige Variable nicht-linear und in komplexer Art und Weise von

unabhängigen Variablen beeinflusst wird. Die Leistungsfähigkeit eines KNN sollte allerdings immer mit der klassischen statistischer Verfahren verglichen werden [18]. Im konkreten Einzelfall muss dann abgewogen werden, ob der potenzielle Zugewinn bei der Vorhersagegenauigkeit den erhöhten Aufwand bei der Implementation eines KNN rechtfertigt. Insgesamt stellen künstliche neuronale Netze eine interessante Ergänzung zu klassischen Vorhersagesystemen dar. Ihre Praxistauglichkeit muss aber erst in weiteren Untersuchungen validiert werden.

Korrespondierender Autor

PD Dr. L. H. J. Eberhart

Klinik für Anästhesie und Intensivtherapie,
Philipps-Universität, Baldingerstr. 1, 35033 Marburg
E-Mail: eberhart@mail.uni-marburg.de

Literatur

1. Baxt WG (1991) Use of an artificial neural network for the diagnosis of myocardial infarction. *Ann Intern Med* 115:843–848
2. Baxt WG (1992) Analysis of the clinical variables driving decision in an artificial neural network trained to identify the presence of myocardial infarction. *Ann Emerg Med* 21:1439–1444
3. Baxt WG, Shofer FS, Sites FD, Hollander JE (2002) A neural computational aid to the diagnosis of acute myocardial infarction. *Ann Emerg Med* 39:366–373
4. Dybowski R, Weller P, Chang R, Gant V (1996) Prediction of outcome in critically ill patients using artificial neural network synthesised by genetic algorithm. *Lancet* 347:1146–1150
5. Ebell H (1993) Artificial neural networks for predicting failure to survive following in-hospital cardiopulmonary resuscitation. *J Fam Pract* 36:297–303
6. Flanagan JR, Pittet D, Li N, Thievent B, Suter PM, Wenzel RP (1996) Predicting survival of patients with sepsis by use of regression and neural network models. *Perform Qual Health Care* 4:96–103
7. Geldner G, Schwarz U, Ruoff M, Romeiser J, Lendl M, Schütz W, Georgieff M (1999) Entwicklung eines neuen Closed-Loop Systems. Regelung einer Mivacurium-induzierten neuromuskulären Blockade. *Anaesthesist* 48:157–162
8. Kamangar FA, Behbehani K (1997) An artificial neural network-based controller for the control of induced paralysis using vecuronium bromide. *Ann Biomed Eng* 25:1040–1051
9. Kennedy RL, Harrison RF, Burton AM et al. (1997) An artificial neural network system for diagnosis of acute myocardial infarction (AMI) in the accident and emergency department: evaluation and comparison with serum myoglobin measurements. *Comput Methods Programs Biomed* 52:93–103
10. Laffey JG, Tobin E, Boylan JF, McShane AJ (2003) Assessment of a simple artificial neural network for predicting residual neuromuscular block. *Br J Anaesth* 90:48–52
11. Lawrence J (1992) Neuronale Netze – Computersimulation biologischer Intelligenz. *Systema*, München
12. Patterson, Dan W (1997) Künstliche neuronale Netze – Das Lehrbuch. Prentice Hall, München London Mexiko

13. Rutledge R (1995) Injury severity and probability of survival assessment in trauma patients using a predictive hierarchical network model derived from ICD-9 codes. *J Trauma* 38:590–601
14. Santanen OAP, Svartling N, Haasio J, Paloheimo MPJ (2003) Neural nets and prediction of the recovery rate from neuromuscular block. *Eur J Anaesth* 20:87–92
15. Tu JV (1996) Advantages and disadvantages of using artificial neural networks versus logistic regression for predicting medical outcomes. *J Clin Epidemiol* 49:1225–1231
16. Tu JV, Guerriere MRJ (1993) Use of a neural network as a predictive instrument for length of stay in the intensive care unit following cardiac surgery. *Comput Biomed Res* 26:220–229
17. Tu JV, Mazer CD, Levinton C, Armstrong PW, Naylor CD (1994) A predictive index for length of stay in the intensive care unit following cardiac surgery. *Can Med Assoc J* 151:177–185
18. Wong LS, Young JD (1999) A comparison of ICU mortality prediction using the APACHE II scoring system and artificial neural networks. *Anaesthesia* 54:1048–1054
19. Zell A (1996) Simulation neuronaler Netze. Addison-Wesley, München