Statistische Mechanik Neuronaler Netzwerke

Wolfgang Kinzel, Institut für Theoretische Physik, Universität Würzburg,
Oktober 2020

Die Künstliche Intelligenz (KI) erzeugt zur Zeit großes Interesse in den Medien, der Politik und in der Wirtschaft. In der Tat gibt es immer mehr Anwendungen der KI, die Erstaunliches leiten. Auch in der physikalischen Grundlagenforschung werden einige neue Projekte gefördert, die mit den Methoden der KI Daten analysieren und neue Erkenntnisse liefern.

Unter Künstlicher Intelligenz versteht man maschinelles Lernen, also Algorithmen, die anhand von Daten selbsttätig gewisse Aufgaben lernen. Und meistens orientieren sich solche Algorithmen an der Funktion und Struktur von Nervenzellen des Gehirns, deshalb nennt man sie Neuronale Netzwerke. Solche großen strukturierten Netzwerke mit mehreren Schichten von künstlichen Neuronen werden heute als *Deep Learning* bezeichnet.

In der Physik werden Neuronale Netze nicht nur angewendet, sondern die Theoretische Physik hat seit einigen Jahrzehnten dazu Modelle entwickelt, die mit den Methoden der Statistischen Mechanik analytisch gelöst werden können. In diesem Beitrag will ich kurz erläutern, wieso die Vielteilchenphysik einen Beitrag zum Verständnis Neuronaler Netze liefern kann. Mithilfe der Entropie bzw. der Freien Energie kann berechnet werden, wie viel Information ein Netzwerk speichern kann und wie viele Beispiele es lernen muss, um eine unbekannte Regel zu erkennen.

Die Statistische Mechanik Neuronaler Netze begann 1982 mit der Publikation von John Hopfield. Diese Arbeit wurde mittlerweile fast 10.000 mal zitiert. Hopfield wies darauf hin, dass ein einfaches Modell des Neuronalen Netzes eine gewisse Ähnlichkeit hat zu den Spingläsern, also Modellen zum ungeordneten Magneten. Damit konnten die Physiker ihre Methoden zu ungeordneten Systemen auf die Neuronalen Netze anwenden. Wie sieht ein solches Modell aus?

Das Hopfield Modell besteht aus N vielen nichtlinearen Einheiten, die miteinander wechselwirken. In Analogie zum biologischen Vorbild wird jede Einheit als Neuron bezeichnet. Im einfachsten Fall hat das Neuron nur zwei Zustände: $S_i = +1$ und $S_i = -1$. In der biologischen Sprache bedeutet das, ein Neuron sendet Impulse aus oder es ruht. Jedes Neuron S_i empfängt Signale von vielen anderen S_j, und diese Signale werden mit einer Kopplungsstärke w_{ij} gewichtet, in Analogie zur Synapse im Gehirn. Wenn die Summe der gewichteten Signale einen gewissen Schwellenwert überschreitet, so feuert das Neuron, anderenfalls bleibt es still. Mathematisch sieht das so aus:

$$S_i = \text{sign} \sum_{j=1}^{N} w_{ij} S_j \tag{1}$$

Eine solche logische Schwelleneinheit wurde schon 1943 von McCulloch und

Pitts publiziert. Diese Funktion trennt die beiden Werte $S_i = \pm 1$ durch eine Ebene im hochdimensionalen Raum, auf der der Vektor $w_i = (w_{i,1}, w_{i,2}, ...)$ senkrecht steht. Wie kann ein Netzwerk aus vielen solchen Neuronen Informationen speichern und aus Beispielen lernen? Zunächst zum ursprünglichen Hopfield Modell, das als Assoziativspeicher aufgefasst werden kann.

Assoziativspeicher

Das Netz besteht aus N Neuronen, die miteinander gekoppelt sind und mit der obigen Gleichung (1) wechselwirken. Allerdings wird vorausgesetzt, dass die Kopplungen symmetrisch sind, $w_{ij} = w_{ji}$. Dann kann man zeigen, dass die obige Dynamik die folgende Funktion minimiert

$$H = -\sum_i \sum_{j<i} w_{ij} S_i S_j$$

Hier sehen wir schon den Zusammenhang mit einem Ising-Modell mit ferro- ($w_{ij} > 0$) oder antiferro- ($w_{ij} < 0$) magnetischen Kopplungen bei dem die Funktion H die Energie des Magneten bedeutet, und wir werden hier ebenfalls den Begriff Energie verwenden. Nun soll in diesem Netz eine Menge von Mustern (Daten, Bilder, ...) gespeichert werden. Dazu soll jedes Muster möglichst ein Minimum der Energie H werden. Es sollen also p viele Muster $\xi_i^k = \pm 1$ $(i = 1, ..., N; k = 1, ..., p)$ mit jeweils N Bits gespeichert werden.

Hopfield hat folgende Kopplungen vorgeschlagen:

$$w_{ij} = \frac{1}{N} \sum_{k=1}^{p} \xi_i^k \xi_j^k$$

Jede Kopplung des Hopfield-Modells speichert somit die Information über sämtliche Muster. Im Gegensatz zum Computer sind die Daten nicht in numerierten Schubladen sondern völlig verteilt gespeichert. Wie wird diese Information abgerufen? Dazu startet man mit einem Bruchteil der Daten aus einem Muster und lässt das Netzwerk mit der Gleichung (1) relaxieren. Die Abb.(1) zeigt, dass das Netz in wenigen Schritten zum gespeicherten Muster relaxiert. Das Netz funktioniert somit als Assoziativspeicher, als inhaltadressierbarer Speicher: es vervollständigt die eingegebenen Daten, wenn sie einen Überlapp zu einem der gespeicherten Muster haben. Allerdings funktioniert das nur fehlerfrei, wenn das Netz nicht überladen ist; sonst hat das Minimum von H einen Fehler zum Muster oder jedes Minimum hat gar keinen Überlapp mehr zu den Mustern.

Wie findet man das Minimum der Energie H? Für ein einziges Muster ist das einfach: $S_i = \xi_i^1$ liefert das absolute Minimum $H = -\frac{N-1}{2}$. Die Transformation $T_i = S_i \xi_i^1$ liefert offensichtlich einen Ferromagneten mit unendlicher Reichweite der Wechselwirkungen. Für unendlich viele unkorrelierte Muster erhält man dagegen ein Spinglas, einen ungeordneten Magneten. Die Kopplungen sind zufällig verteilt und es gibt einen Wettbewerb zwischen positiven

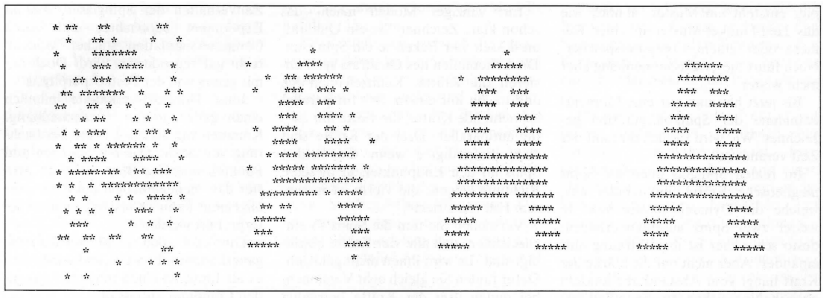

Abbildung 1: Ein Hopfield-Netz mit 20x20 Neuronen hat 30 Muster gelernt. In wenigen Schritten relaxiert das Netz zum vollständigen gespeicherten Muster.

und negativen Kräften, der bei tiefen Temperaturen zu einer ungewöhnlichen magnetischen Phase führt. Die Anzahl der Minima von H wächst exponentiell mit der Systemgröße N, und die Minima haben nur einen winzigen Überlapp zu den Mustern.

Wie viele Muster können so gespeicht werden und wie viel Überlapp zum Muster hat das Minimum der Energie H? Das wurde mit den Methoden der Statistischen Mechanik beantwortet. Dazu werden die Neuronen-Konfigurationen $S = (S_1, S_2, ..., S_N)$ mit einem Boltzmann-Faktor gewichtet und es wird über alle S summiert.

$$F(T, N) = T \ln \sum_S \exp\left(-\frac{H(S)}{T}\right)$$

Dabei ist T ein Parameter, der der Temperatur beim Magneten entspricht, und $F(T, N)$ entspricht der Freien Energie. Für $T \to 0$ erhält man die gesuchten Grundzustände der Energie H. Für endliche Temperaturen gibt es stationäre Zustände, die mit einer entsprechenden stochastischen Dynamik erzeugt werden: die Neuronen reagieren nur mit einer gewissen Wahrscheinlichkeit auf die Gleichung (1).

Das vollständige Hopfield-Modell wurde 1985 von Amit, Gutfreund und Sompolinsky mit den Methoden der Spinglas-Theorie gelöst. Allerdings erfordert eine analytische Lösung zufällig verteilte Muster und den thermodynamischen Limes $N \to \infty$. Dabei bestimmt der Parameter $\alpha = p/N$ ob im unendlich großen Netzwerk nur endlich oder unendlich viele Muster gespeichert werden. Da jedes Neuron mit jedem anderen wechselwirkt, kann man das Modell mit einer Molekularfeld-Theorie mit vielen Ordnungsparametern lösen. Dabei wird die Freie Energie über die zufällig verteilten Muster gemittelt (Replika-Methode). Die Minima der Freien Energie als Funktion der Ordnungsparameter liefern stabile Strukturen im Raum der Neuronenzustände, analog zur Magnetisierung im Ferromagneten.

Das analytische Ergebnis zeigt eine komplexe Struktur der Energie. Für endlich viele Muster, $\alpha = 0$, ist das Modell ein perfekter Speicher, alle Muster sind globale Minima der Freien Energie. Allerdings gibt es weitere höherlie-

gende lokale Minima, die sich aus mehreren Mustern zusammensetzen. Mit zunehmenden Wert von α relaxiert die Dynamik (1) nicht mehr in die perfekten Muster sondern in Zustände dicht daneben. Wie beim Magneten gibt es einen Phasenübergang: Der Überlapp zu einem Muster nimmt mit zunehmender Temperatur ab bis er plötzlich völlig verschwindet, analog zum Physenübergang 1. Ordnung. Bei $\alpha \geq 0.14$ funktioniert der Speicher plötzlich nicht mehr, selbst bei $T = 0$. Es gibt einen Phasenübergang in eine Spinglas-Phase, deren Minima keinen Überlapp mehr zu den gespeicherten Mustern haben.

Aber selbst innerhalb der geordneten Phase gibt es einen weiteren Übergang zu einer Phase bei der die Muster nur noch lokale Minima beschreiben und das globale Minimum die Spinglas-Phase ist. Die Ordnung koexistiert mit der Unordnung.

Kapazität

Das Hopfield-Modell mit den speziell gewählten obigen Kopplungen hat eine Speicherkapazität von 14 % für zufällig gewählte Daten. Gibt es Kopplungen mit höherer Speicherkapazität? Diese Frage wurde ebenfalls mit den Methoden der Statistischen Mechanik für unendlich große Netzwerke beantwortet. Die leider sehr jung verstorbene Elisabeth Gardner hat 1988 dazu Folgendes berechnet: wie viele Kopplungen w_{ij} gibt es, die einen Satz von p Mustern mit den Gleichungen (1) perfekt abbilden können? Bei der Lösung des Hopfield-Modells wurde über alle Konfigurationen S der Neuronen summiert, bei der Berechnung der Kapazität wird nun über alle Konfiguration w der Synapsen summiert bzw. integriert und es wird die Entropie berechnet. Für Zufallsmuster lautet das Ergebnis $\alpha_c = 2$, es können somit doppelt so viele Muster gespeichert werden wie es Neuronen gibt. Alle Muster sind in sämtlichen Kopplungen gleichzeitig gespeichert. Für korrelierte Muster wächst die Speicherkapazität mit dem Grad der Korrelation.

Allerdings sind die Kopplungen nicht mehr symmetrisch, $w_{ij} \neq w_{ji}$, es gibt keine Energie mehr im Neuronenraum. Die Gleichungen (1) zerfallen in N unabhängige Gleichungen. Jede Gleichung enthält N Eingabe- und ein einzelnes Ausgabe-Neuron, und ein solches einfaches Neuronales Netzwerk wurde *Perzeptron* genannt und schon in den 60er Jahren ausführlich untersucht. Das Perzeptron enthält interessante Mathematik, und tatsächlich hat Cover 1965 die Kapazität des Perzeptrons für jedes (p, N) mit geometrischen Methoden berechnet, mit dem Ergebnis $\alpha_c = 2$ für unendlich große Netzwerke.

Die Statistische Mechanik sagt uns, wie viele Kopplungen es gibt, die sämtliche Muster exakt abbilden. Aber wie findet man diese Kopplungen? Das wurde auch schon von den Mathematikern bewiesen: man findet solche Kopplungen mit einer Lernregel, die 1949 vom Psychologen Donald Hebb vorgestellt wurde: die Stärke der Synapsen ändert sich entsprechend der neuronalen Aktivitäten an ihren Enden. In diesem Modell bedeutet das: wenn ein Muster ξ_j^k gelernt werden soll, und Gleichung (1) ist nicht erfüllt, so wird die Kopplung proportional zur Aktivität an ihren Enden geändert:

$$\Delta w_{ij} = \frac{1}{N} \xi_i^k \xi_j^k \tag{2}$$

1962 hat Rosenblatt mathematisch beweisen: wenn es Kopplungen gibt, die einen Satz von Muster exakt abbilden, dann findet diese Lernregel auch eine dieser Lösungen.

Verallgemeinern

Der Assoziativspeicher findet nur wenige Anwendungen, aber er ist vielleicht ein erster Schritt zum allgemeinen Verständnis biologischer Neuronennetze. Viel wichtiger und mittlerweile weit verbreitet für die Anwendungen der Künstlichen Intelligenz sind die Neuronalen Netze, die eine unbekannte Regel aus einem Satz von Beispielen erkennen können. Das Netz lernt eine Menge von Daten, indem es seine Kopplungen an die jeweiligen Eingabe-Ausgabe-Bits anpasst. Nach der Lernphase macht das Netz eine Vorhersage für einen unbekannte Eingabe, es verallgemeinert. Mit schnellen Computern und riesigen Datenmengen können damit heutzutage eindrucksvolle Ergebnisse erzielt werden. Deep-Learning besiegt Schachgroßmeister, erkennt Gesichter, analysiert Röntgenbilder, findet neue Materialien und vieles mehr.

Die Theoretische Physik hat auch in diesem Fall eine analytische Lösung geliefert. Gardner und Derrida haben mit der Statistischen Mechanik der Kopplungen berechnet, wie sich der Verallgemeinerungsfehler mit der Anzahl der gelernten Beispiele verringert. Dazu betrachten wir wieder das einfache Perzeptron, Gleichung (1) für ein einzelnes Ausgabe-Bit $i = 0$. Um den Verallgemeinerungsfehler zu berechnen, benötigen wir Beispiele, die durch eine Regel erzeugt wurden, und das Neuronale Netz w soll lernen, diese Regel zu erkennen. Wir nehmen einen Satz von p Beispielen, die selbst wieder durch ein Neuronales Netz v erzeugt worden sind:

$$\xi_0^k = \text{sign} \sum_{j=0}^{N} v_j \xi_j^k, \quad k = 1, ..., p$$

Das Neuronale Netz v wird Lehrer genannt, während das Netz w der Schüler ist, der die vom Lehrer gegebenen Beispiele lernen soll. Man kann zeigen, dass der Verallgemeinerungsfehler ε, also die Wahrscheinlichkeit, dass der Schüler auf eine unbekannte Eingabe eine falsche Antwort gibt, durch das Skalarprodukt zwischen den beiden Vektoren v und w gegeben ist. Gardner und Derrida ist es gelungen, für unendlich große Netzwerke die Entropie der Schülervektoren zu berechnen, also das Volumen im Raum der Vektoren w, die einen vorgegebenen Überlapp zwischen Schüler und Lehrer haben. Das Maximum der Entropie als Funktion des Überlapps liefert damit den Verallgemeinerungsfehler ε. Die Wahrscheinlichkeit, einen Fehler zu machen, nimmt natürlich mit der Anzahl der gelernten Beispiele ab. Die Statistische Mechanik liefert das Resultat, dass der Verallgemeinerungsfehler mit der inversen Anzahl der gelernten Beispiele abnimmt, $\varepsilon \sim 1/\alpha$. Mit den Kopplungen von Hopfield findet man dagegen einen langsameren Abfall $\varepsilon \sim 1/\sqrt{\alpha}$.

Mathematisch interessant ist der Fall, dass die Kopplung nur diskrete Werte annehmen können, $w_{0j} = \pm 1$. Dann gibt es einen Phasenübergang, der Verallgemeinerungsfehler springt bei $\alpha > \alpha_c \simeq 1.245$ zum Wert null. Wenn der Schüler also mehr als $1.245N$ viele Beispiele lernt, dann reproduziert er perfekt die Meinung des Lehrers - ein Traum manches Pädagogen. Bei dieser Rechnung wird vorausgesetzt, dass alle Beispiele perfekt gelernt werden. Wie schon erwähnt, ist das mit der Hebbschen Lernregel Gl.(2) garantiert, denn das Problem ist nach Definition lernbar.

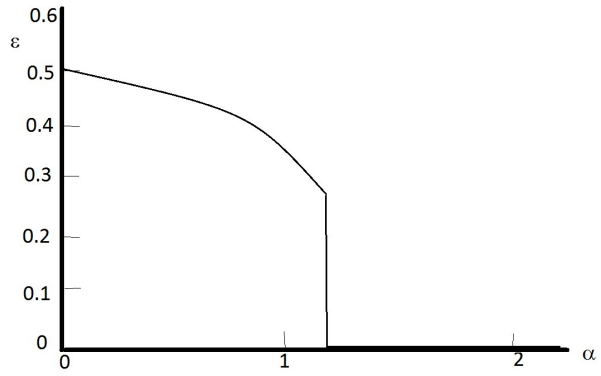

Abbildung 2: Der Verallgemeinerungsfehler des Netzes mit binären Kopplungen als Funktion der gelernten Muster. Werden mehr als $1.24N$ Muster gelernt, macht das Netz keinen Fehler mehr.

Mit den Methoden der Statistischen Mechanik konnten später noch einige andere Fragen analytisch gelöst werden, zum Beispiel: wie gut kann der Schüler lernen, wenn der Lehrer Fehler macht? Wie nimmt der Verallgemeinerungsfehler ab, wenn der Schüler die Beispiele nicht perfekt lernt? Was geschieht, wenn ein komplexer Lehrer (Mehrschicht-Netzwerk) die Beispiele für einen simplen Schüler (Perzeptron) liefert, oder umgekehrt? Wie verbessert sich die Lerngeschwindigkeit, wenn der Schüler Fragen stellt?

Interessant für Anwendungen ist auch der Fall, dass jedes Beispiel nur einmal präsentiert wird und der Gewichtsvektor mit der obigen Lernregel daran angepasst wird. Dadurch gelangt man zu einer Differenzialgleichung für den Verallgemeinerungsfehler, also zur Theorie dynamischer Systeme.

In den fünf Jahrzehnten nach der Publikation von Hopfield wurden zahlreiche Probleme mit dem Methoden der Statistischen Mechanik und der Nichtlinearen Dynamik untersucht. Nur wenige Modelle können exakt gelöst werden, aber wie immer in der Physik können mit guten Näherungen und Computersimulationen neue Erkenntnisse gewonnen werden. Die Physik stellt allgemeinere Fragen als die Biologie oder die Informatik, deshalb werden die Neuronalen Netze auch weiterhin in der physikalischen Forschung eine Rolle spielen.

Zeitreihen-Vorhersage

Neuronale Netze werden auch zur Analyse und Vorhersage von Zeitreihen

eingesetzt. Kann ein einfaches Perzeptron lernen menschliche Reaktionen vorherzusagen? Diese Frage habe ich mit den Studierenden des Lehramtes Physik untersucht, um sie spielerisch mit den Neuronalen Netzen vertraut zu machen. Jede Student*in wurde aufgefordert, eine Folge von einigen hundert 0 und 1 auf eine Datei zu schreiben. Das Perzeptron hat dann mit den letzten 10 Bits eine Vorhersage für die folgende Eingabe gemacht und danach das eingegebene Bit mit Hebbschen Regel gelernt, wie in der Abb.(3) skizziert ist. Nach der Hälfte der eingegebenen Daten wurde dann die Trefferquote registriert. Durch zufälliges Raten wird eine Trefferquote von 50 % erreicht. Wenn die Student*in schlauer ist als das Perzeptron, kann sie abschätzen, ob das Perzeptron null oder eins sagt und dann das Gegenteil nehmen, damit erreicht sie eine Trefferquote von weniger als 50 %. Wenn der Algorithmus aber schlauer ist, erreicht er einen Wert über 50%. Die Abb.(4) zeigt, dass eine bewegliche Hyperebene in der Lage ist, menschliche Reaktion vorherzusagen, mit einem Mittelwert von etwa 65 %. Auf meiner Webseite kann jeder mit dem Perzeptron spielen, alternativ gibt es das App *perceptron* im Google PlayStore.

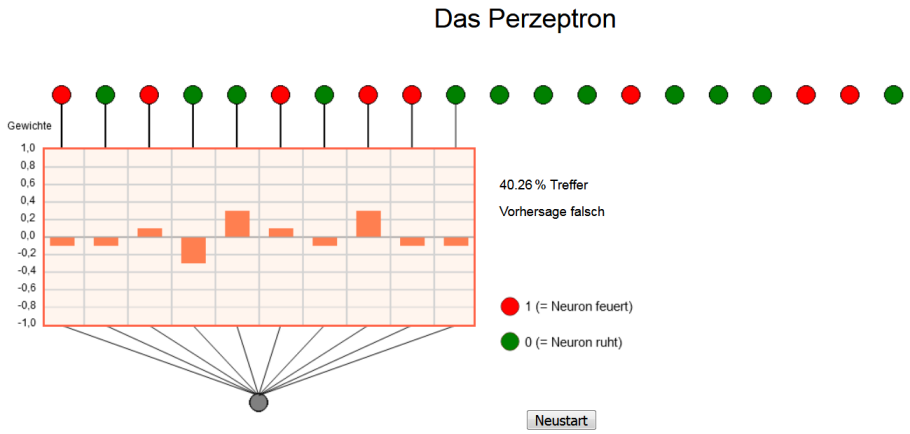

Abbildung 3: Ein Perzeptron lernt einen Zeitreihe und macht Vorhersagen für das folgende Bit.

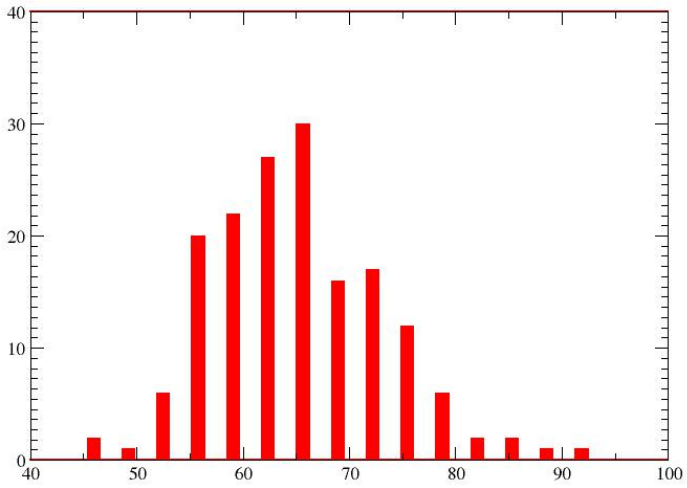

Abbildung 4: Anzahl der Studierenden als Funktion der Treffer-Wahrscheinlichkeit des Perzeptrons.

Bibliographie:

J. Herz, A. Krogh und R.G. Palmer, Introduction to the Theory of Neural Computation, Addison Wesley (1991)

J.J. Hopfield, Proc. Nat. Acad. Sciences, USA79, 2554 (1982)

D. Amit, H. Gutfreund und H. Sompolinsky, Phys. Rev. Lett. 55, 1530 (1985)

E. Gardner, J. Phys. A21, 257 (1988)

T. M. Cover, IEEE Transactions on Electronic Computers, 14, 326 (1965)

E. Gardner und B. Derrida, j. Phys. A22, 1983 (1989)

https://www.physik.uni-wuerzburg.de/tp3/lehre/applets-zur-physik/