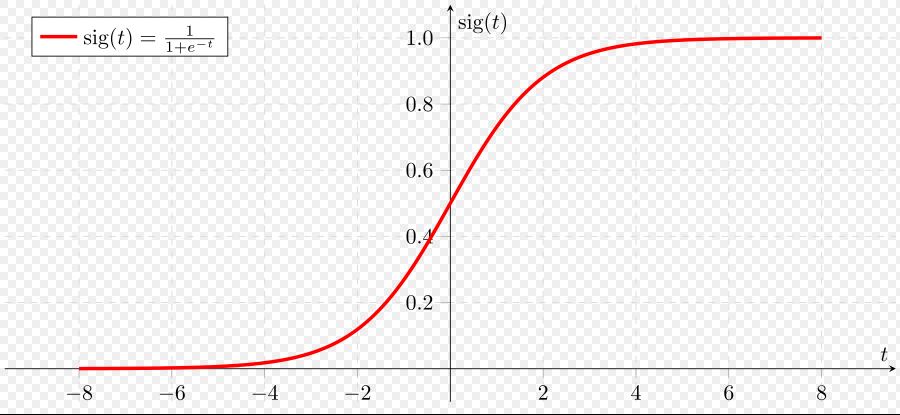
**Einleitung**

In Sinne von Maschinelles Lernen, Künstliche Neurale Netze (engl. ANN, *artificial neural network*) handelt sich um Netze, welche aus Neuronen und die Neuronen miteinander verbindenden Synapsen fast wie Zentralnervensystem (nämlich Gehirn) von Lebewesens besteht. ANN wird zum Einschätzung und Approximation der Funktionen angewendet, die eventuell mit einer großen Menge von Input abhängig sind. [vgl. 1] Die ANN können durch drei Eigenschaften festgelegt werden.

**Architektur:** bestimmt welche Variablen in Netzwerk und seiner Topologie beteiligt sind. Anzahl der verdeckten Schichten außer eingabe- und Ausgabeschicht oder Anzahl der Neuronen in jeweiliger Schicht sind beispielweise Variablen, die die Struktur der Netzwerke aufweisen. Es wurde hier Mehrlagiges Perzeptron (engl. MLP, *Multilayer perceptron*) mit eins verdeckter Schicht ausgesucht.

**Aktivierungsregel:** bestimmt wie die Aktivitäten der Neuronen in Reaktion aufeinander verändert. Z.b.s. verfügen Jede verdeckte Schicht und die Ausgabeschicht über eine (eigene) Aktivierungsfunktion. Diese können linear oder nicht-linear sein. Nicht-lineare Aktivierungsfunktionen machen das Netzwerk besonders mächtig. [vgl. 2]

In diesem Projekt wurde Sigmuidfunktion als Aktivierungsfunktionen angewendet, da der Einsatz von differenzierbaren Funktionen die Verwendung von Lernmechanismen (Backpropagation-Algorithmus) ermöglicht. Als Aktivierungsfunktion eines künstlichen Neurons wird die Sigmoidfunktion auf die Summe der gewichteten Eingabewerte angewendet, um die Ausgabe des Neurons zu erhalten. Die Sigmoidfunktion wird vor allem aufgrund ihrer einfachen Differenzierbarkeit als Aktivierungsfunktion bevorzugt verwendet.



**Lernregeln:** bestimmt die Weise, in der die Gewichte des Netzes durch Training im Laufe der Zeit angepasst werden. Es wurde hier **Backpropagation** als Lernalgorithmus angewendet, welches ein verbreitetes Verfahren für das Einlernen von künstlichen neuronalen Netzen ist. [vgl. 3]

**Verwandte Arbeiten**

Schneider und Rosa haben sich beschäftigt, 4 heuristische und Algorithmen zu implementieren, die relativ fähig 4-Gewinnt spielen können, um Daten Set zum Training des Neuronalen Netzes zu generieren. Nebenbei haben sie darum gekümmert, dass Qualität des Netzes durch Variation der Parameter sowie Netzwerk Topologie und Größe von Trainingsmenge zu verbessern. [vgl. 4]

Chellapilla und fogel haben sich damit befasst, Ansatz der künstlichen Intelligenz durch Implementierung der KIs für ein Nullsummenspiel (Tic-Tac-Toe) und ein nicht Nullsummenspiel (Gefangenendilemma) ausprobieren. Dafür haben sie zwei Methode von Maschinellen Lernens nämlich Neuronale Netze und Evolutionärer Algorithmus angewendet, und miteinander verglichen. [vgl. 5]

1. MacKay, David, J.C. (2003). Information Theory, Inference, and Learning Algorithms. Cambridge University Press. ISBN 9780521642989.
2. Neural Networks FAQ. (am 5. September 2015) (<ftp://ftp.sas.com/pub/neural/FAQ2.html#A_act>).
3. Rumelhart, David E.; Hinton, Geoffrey E.; Williams, Ronald J. (8 October 1986). "Learning representations by back-propagating errors". Nature. 323 (6088): 533–536. doi:10.1038/323533a0
4. Marvin Oliver Schneider, João Luís Garcia Rosa, Neural Connect 4 – A Connectionist Approach to the Game
5. Kumar Chellapilla, David B. Vogel, Evolution Neural Networks Games and Intelligence, IEEE invited paper