

# 「ニューラルネットワーク」

「AIによる制御設計の原理研究」(p. 8)に記載

技術本部 基盤技術研究所 運動制御研究室 芝原 杏奈

## 1

### ニューラルネットワークとは

#### 1.1 定義

AIにおけるニューラルネットワーク(以下NN)とは、生物の学習メカニズムを模倣した機械学習手法の一種である。生物の脳のニューラルネットワーク(神経網)の機能の一部をコンピュータ上で数理モデルとして模倣<sup>注1)</sup>することにより、学習機能を実現する。

NNの構成には多数の種類が存在するが、ここでは順伝播型のNN(一方向のみにデータが流れるNN)のみを対象として説明する。尚、以降で説明する各種の要素の構成などは全て一例であり、これら以外の構成も存在する。また、概要の説明に焦点を当てるため、演算に関する一部の要素などを省略している。

注1) 実用的なNNは生物の脳の厳密な模倣ではない場合

が多い。ここでは生物の脳を厳密に考慮したNNではなく、実用的な数理モデルとしてのNNを対象に説明する。

#### 1.2 NNの仕組み

ここでは2つのセンサ値を入力に取る制御システムの代用として働くNNを例として、NNの仕組みを説明する。NNは例えば図1のような構成を取ることができる。円で表現した要素はノードと呼ばれるものであり、これは演算ユニットである。ノードの縦方向の集まりは層と呼ばれる。層は先頭を入力層、最後尾を出力層と呼び、それらの間にある層は中間層と呼ぶ。中間層は任意の数を取ることができる。

入力層から入力されたセンサ値は中間層を通り、出力層から出力される。その際、通過する各ノード内でデータに対して演算が行われる。そのノード毎に繰り返される演算によって、出力層に到達したときには制御指令値となっている。

ノードでは、各入力に個別の重みを掛けたものを足し合わせ、その結果を活性化関数(NNの出力を

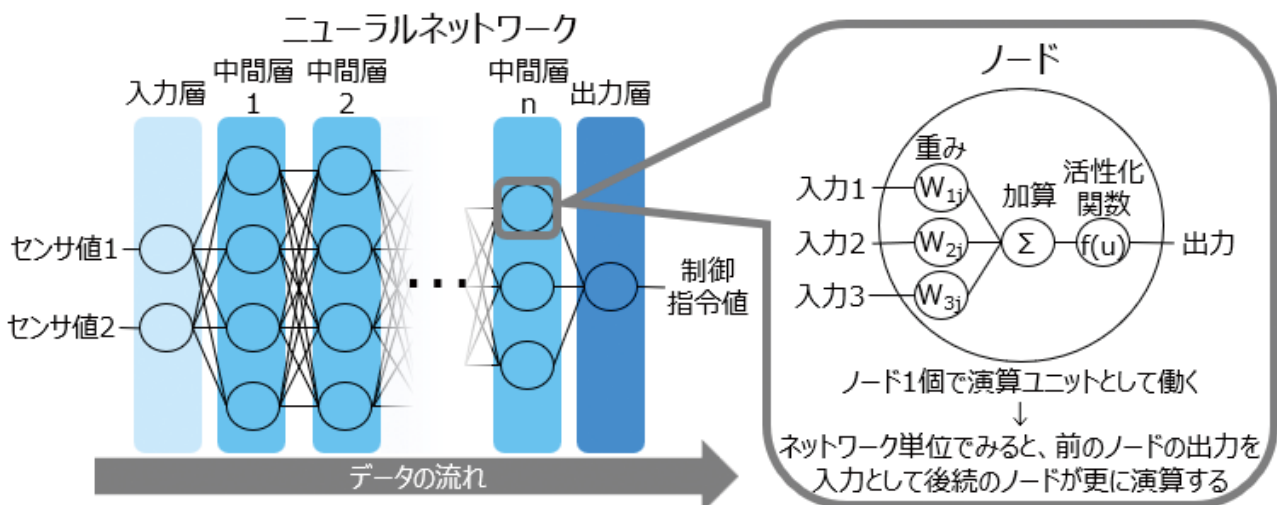


図1 NNの仕組み

決定するための関数)に通すという処理が行われる。活性化関数に非線形な関数を利用することで、入力に対して非線形なNNの出力を取ることができる。

### 1.3 学習

NNにおける学習とは、各ノードの持つ重みを更新する処理である。重みの更新には多くの場合、バックプロパゲーション（以下BP）という手法が使用されるため、ここではBPに関して説明する。BPとは、NNの各ノードの持つ重みを後ろの層から前の層へと順に更新する手法である。この手法により、NNは多数の中間層があっても学習することができる。

1.2節の例で説明すると、NNが出力した制御指令値が最適な制御指令値とどの程度乖離しているかを表す値を損失と呼ぶ。損失を元にして図2のように、各層に対する損失の影響の大きさを考慮しながら、その層の各ノードの重みを更新していくことによってNNは学習する。

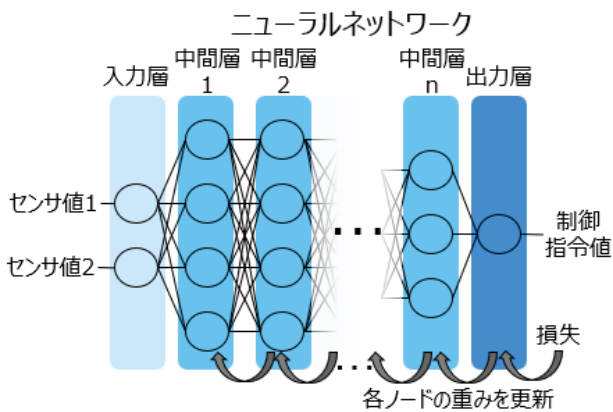


図2 NNの学習のイメージ

また、損失の算出方法はNNを適用する用途などによって異なり、算出方法によっては最適な制御指令値との直接比較ができなくても学習できる場合がある。そのような直接比較ができない例として、強化学習でNNを用いるケースが挙げられる。

強化学習でNNを用いる一例としてここでは、著名な強化学習アルゴリズムの一つであるDQN (Deep Q Network) の損失計算を取り上げる。DQNでは大まかには次のような式で損失を算出する。

$$E[(r + \gamma \max_a Q' - Q)^2]$$

( $E[ ]$  は  $[ ]$  内の期待値,  $r$ は報酬,  $\gamma$ は収益を考慮する区間の長さを制御するパラメータ,  $\max_a Q'$ は現在の入力から想定される収益の期待値の最大値,  $Q$ は最近の制御結果から想定される収益の期待値)

上記の式は説明を簡単にするためいくつかの要素を省略したが、このような考え方の式で損失算出することで、最適な制御指令値と直接比較せずともNNを学習することができる。

## 2 ニューラルネットワークの応用

### 2.1 機能の多様性

NNの構成には様々な種類が存在する。画像処理に向けたCNN (convolutional neural network) や時系列処理に向けたRNN (recurrent neural network) など、用途に応じた種類の層などを接続することでNNは多様な機能を実現することができる。

### 2.2 代表的な用途

NNは様々な用途に用いることができるが、代表的なものとしては次の用途が挙げられる：

- ①教師あり学習：分類問題、予測問題
- ②教師なし学習：次元削減、クラスタリング、データ生成
- ③強化学習：制御、プランニング

これらは単独で使用することもできるが、組み合わせることで更に複雑な動作を実現することも可能である。