

ニューラルネットワークの基礎

概要

本稿では、現代の人工知能技術の中核をなすニューラルネットワークの基本的な概念について解説する。最小単位であるニューロン（パーセプトロン）のモデルから始め、それらを組み合わせた多層構造、非線形性を導入する活性化関数の役割、そしてネットワークがどのようにして学習を行うかの概要までを段階的に説明する。

1 はじめに

ニューラルネットワークは、人間の脳の神経回路網を数学的に模倣したモデルであり、画像認識、自然言語処理、音声認識など、多岐にわたる分野で驚異的な成果を上げてきた。その本質は、多数の単純な計算ユニットを階層的に結合させることで、複雑なデータの中からパターンや関係性を自動的に学習する能力にある。本稿では、この強力なモデルの根底にある基本的な仕組みを解き明かしていく。

2 基本単位：ニューロン（パーセプトロン）

ニューラルネットワークの最も基本的な構成要素は、「ニューロン」または「パーセプトロン」と呼ばれる計算ユニットである。これは、生物の神経細胞が他の細胞から信号を受け取り、それに応じて信号を発火させるプロセスをモデル化したものである。

ニューロンは、複数の入力信号 (x_1, x_2, \dots, x_n) を受け取る。各入力には、その重要度を示す「重み」 (w_1, w_2, \dots, w_n) が乗じられる。これらの重み付き入力の総和に、「バイアス」 (b) と呼ばれる切片項が加えられる。この総和が、ニューロンの発火しやすさを決定する。

$$u = \sum_{i=1}^n w_i x_i + b$$

最後に、この値 u は「活性化関数」 f に入力され、最終的な出力 y が決定される。

$$y = f(u) = f\left(\sum_{i=1}^n w_i x_i + b\right)$$

重みとバイアスは、ネットワークが学習する際に調整されるパラメータである。

3 ネットワークの構造：多層パーセプトロン

個々のニューロンは単純な決定しかできないが、これらを多数、階層的に配置することで、非常に複雑な問題に取り組むことが可能になる。このようにニューロンを層状に配置した構造を「多層パーセプトロン (MLP)」と呼ぶ。

MLP は、主に 3 種類の層から構成される。

- **入力層 (Input Layer):** 外部からデータを受け取る最初の層。ここには計算を行うニューロンはなく、入力値を次の層に渡す役割のみを担う。
- **隠れ層 (Hidden Layers):** 入力層と出力層の間に位置する層。隠れ層が複数ある場合、それを「深い (Deep)」ニューラルネットワークと呼ぶ。ここで入力データの複雑な特徴抽出が行われる。
- **出力層 (Output Layer):** ネットワークの最終的な計算結果を出力する層。タスクの種類 (分類、回帰など) に応じて、ニューロンの数や活性化関数が設計される。

情報は入力層から隠れ層を経て出力層へと、一方向にのみ伝播する。このプロセスを「順伝播 (Feedforward)」と呼ぶ。

4 活性化関数の役割

活性化関数はニューラルネットワークにおいて極めて重要な役割を担う。もし活性化関数がなければ、各層の計算は単なる線形変換 (行列の乗算と加算) の繰り返しとなり、ネットワーク全体としても一つの線形変換で表現できてしまう。これでは、複雑な非線形関係を学習することができない。

活性化関数は、この線形和に**非線形性**を導入する。これにより、ネットワークは曲線的な決定境界や複雑なデータ分布をモデル化する表現力を獲得する。

4.1 代表的な活性化関数

- **シグモイド関数 (Sigmoid Function):** 古くから用いられてきた関数で、出力を 0 から 1 の間の値に変換する。確率的な解釈がしやすいため、出力層で用いられることがあった。

$$\sigma(x) = \frac{1}{1 + e^{-x}}$$

- **ReLU (Rectified Linear Unit):** 現在、隠れ層で最も広く使われている活性化関数。入力が 0 以下なら 0 を、0 より大きければその値をそのまま出力する。計算が非常に高速であるという利点がある。

$$\text{ReLU}(x) = \max(0, x)$$

- **ソフトマックス関数 (Softmax Function):** 主に多クラス分類問題の出力層で用いられる。複数のニューロンの出力を、合計が 1 となる確率分布に変換する。

$$y_k = \frac{e^{u_k}}{\sum_{i=1}^n e^{u_i}}$$

5 ニューラルネットワークの学習プロセス概要

ニューラルネットワークの「学習」とは、与えられたデータに対して望ましい出力をするように、ネットワーク内の重みとバイアスを自動的に調整するプロセスである。これは、以下のステップで進行する。

1. **損失関数 (Loss Function) の定義:** ネットワークの出力と正解 (教師データ) がどれだけ異なっているかを定量的に測るための関数を定義する。代表的なものに、回帰問題で使われる「平均二乗誤差」や、分類問題で使われる「交差エントロピー誤差」がある。

2. **勾配の計算:** 損失を各重みパラメータで偏微分し、損失を最も大きく減少させる方向（勾配）を計算する。この計算を効率的に行うためのアルゴリズムが、まさに**誤差逆伝播法（Backpropagation）**である。
3. **パラメータの更新:** 計算した勾配に基づき、損失が小さくなるように重みとバイアスを少しだけ更新する。この更新プロセスは「勾配降下法（Gradient Descent）」などの最適化アルゴリズムによって行われる。

この3ステップを多数のデータに対して繰り返し行うことで、ネットワークは徐々に賢くなり、未知のデータに対しても高い精度で予測ができるようになる。

6 結論

ニューラルネットワークは、脳の神経細胞を模した単純な計算ユニット「ニューロン」を多層に組み合わせ、非線形な活性化関数を挟むことで高い表現力を獲得した数理モデルである。その学習は、損失関数を最小化するように、誤差逆伝播法と勾配降下法を用いて膨大な数のパラメータを自動調整するプロセスに基づいている。本稿で述べた基本概念は、より高度なネットワーク構造や最新の深層学習技術を理解するための重要な土台となる。