

深層学習の整理帳

tomixy

2025 年 6 月 20 日

目次

第 1 章	ニューラルネットワークの基本	2
	ニューロン	2
	パーセプトロン	3
	ベクトルと行列を用いた表現	4

第 1 章

ニューラルネットワークの 基本



ニューロン

ニューロンは動物の脳の基本ユニットである

脳はほとんどがニューロンで作られており、

ref: ニューラルネット
ワーク自作入門 p48～

電気信号がニューロンから別のニューロンに渡される

ことの連鎖で、たとえば光、音、熱などを感知することができる

ニューロンのモデル化

ニューロンはまず電氣的な入力を受け取り、その後に別の電気信号を放出する

これは、

入力を受け取り、処理を行い、何かを出力する

という関数としてモデル化できる



観察によれば、ニューロンは入力にすぐには反応せずに、入力が大きくなって出力を引き起こすまで、入力を抑制する
この現象を、科学者たちは

入力が**閾値**に達するとニューロンが**発火**する

と表現する

直観的には、ニューロンは小さなノイズ信号は通過させず、強く意図的な信号のみを通過させる

入力信号を受け取り、出力信号を生成するが、何らかの閾値を考慮した関数を**活性化関数**という

単純な**階段関数**も活性化関数だが、境界が滑らかな**シグモイド関数**などの方がよく使われる



パーセプトロン

パーセプトロンは、複数の信号を入力として受け取り、一つの信号を出力するアルゴリズム

ref: ゼロから作る
Deep Learningp21~

ここでいう**信号**とは、電流や川のような「流れ」をもつものをイメージするとよい

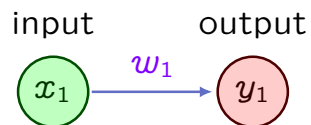
電流が導線の流れ、電子を先に送り出すように、パーセプトロンの信号も流れを作り、情報を先へと伝達していく

ただし、パーセプトロンの信号は、「流す／流さない」の二値の値である
ここでは、0を「流さない」、1を「流す」とする

一つの入力を受け取るパーセプトロン

x_1 は入力信号、 y_1 は出力信号、 w_1 は重みを表す

図中の円はニューロンやノードと呼ばれる



入力信号は、ニューロンに送られる際に、それぞれに固有の重みが乗算される

ニューロンでは、送られた信号の総和が計算され、その総和がある限界値（閾値 θ ）を超えた場合にのみ 1 を出力する（ニューロンが発火する）

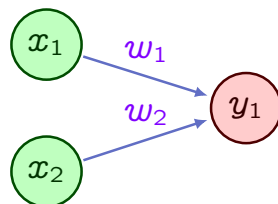
$$y = \begin{cases} 0 & (w_1 x_1 \leq \theta) \\ 1 & (w_1 x_1 > \theta) \end{cases}$$

二つの入力を受け取るパーセプトロン

パーセプトロンは、複数ある入力信号のそれぞれに固有の重みを持つ

そしてその重みは、各信号の重要性をコントロールする要素として働く

つまり、重みが大きければ大きいほど、その重みに対応する信号の重要性が高くなる



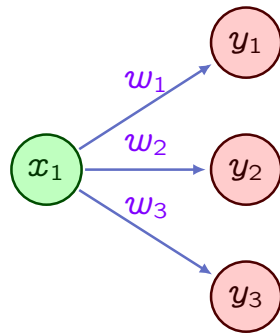
$$y = \begin{cases} 0 & (w_1 x_1 + w_2 x_2 \leq \theta) \\ 1 & (w_1 x_1 + w_2 x_2 > \theta) \end{cases}$$



ベクトルと行列を用いた表現

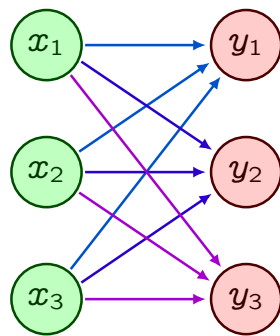
複数の出力

次のような場合は、それぞれ同じ入力を受け取る 3 つの独立したコンポーネントとして動作する

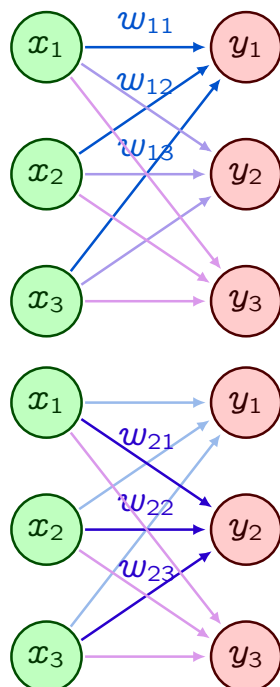


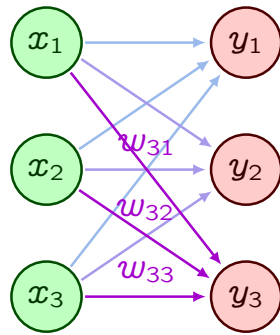
$$\begin{pmatrix} y_1 \\ y_2 \\ y_3 \end{pmatrix} = \begin{pmatrix} w_1 x_1 \\ w_2 x_1 \\ w_3 x_1 \end{pmatrix}$$

複数の入力と出力



「3つの重みが各出力ノードに渡される」という見方で、3つの独立した内積として考えてみる





$$\begin{pmatrix} y_1 \\ y_2 \\ y_3 \end{pmatrix} = \begin{pmatrix} w_{11}x_1 + w_{12}x_2 + w_{13}x_3 \\ w_{21}x_1 + w_{22}x_2 + w_{23}x_3 \\ w_{31}x_1 + w_{32}x_2 + w_{33}x_3 \end{pmatrix} \\
 = \begin{pmatrix} w_{11} & w_{12} & w_{13} \\ w_{21} & w_{22} & w_{23} \\ w_{31} & w_{32} & w_{33} \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \\ x_3 \end{pmatrix}$$