

## バイアス項の無い多層フィードフォワードネットワークに関する一考察 A study on multilayer feedforward networks without bias terms

南葉 宗弘<sup>†</sup>  
NAMBA Munehiro

### 1. はじめに

2012 年頃から注目を集めるようになったディープラーニングでは、ニューラルネットワークの中間層を増やし、ネットワークモデルを深く構成する。深いニューラルネットワーク (DNN) は大規模データセットを用いて学習させると、これまでの機械学習モデルとは一線を画する極めて高い推論性能を発揮することが分かってきた。現在、ディープラーニングは画像処理、音声処理、自然言語処理のみならず、社会の様々な分野において活用が期待されている [1,2]。

ニューラルネットワークにおける各ニューロンでは、入力の重み付け総和に一定のバイアス値を加えたものを内部ポテンシャル値とし、これを活性化関数により変換した値を出力としている。ディープラーニングにおいても調整すべきパラメタが重みとバイアスであることに変わりはない。それにも関わらず、これまでの多くの研究ではニューラルネットワークの能力に関して、重みの影響だけが議論されてきた。

バイアスが注目されるようになったのはここ数年、DNN の推論根拠を説明する方法 [3] を模索する中で、重み以外のパラメタとしてバイアスが再発見されてからである。Wang らは DNN を説明するための視覚化法を提案し、その中で、重みだけでなく実はバイアスも DNN の推論結果に大きく影響していることを明らかにした [4]。また、Mohan らは画像中のノイズを抑制するための DNN から全てのバイアスを取り除くと、かえって DNN のノイズ抑制能力を向上させることができると報告している [5]。これら 2 つの研究結果は、DNN におけるバイアスの重要性を示すとともに、そもそも DNN にバイアスが真に必要なのかという疑問も提起している。

本研究では、まず古典的な普遍性定理を満たす中間層 1 層のみの多層フィードフォワードネットワークにおけるバイアス項の役割について考察し、その結果を元に、特定の条件を満たす活性化関数を用いる場合、中間層を 2 層に拡張することで、内部ポテンシャルの計算式からバイアス項を取り除いても多層フィードフォワードネットワークの関数近似能力の普遍性が失われないことを明らかにする。簡単な線形分離問題や MNIST データセットを用いた分類実験の結果をもとに本研究の妥当性を示す。

## 2. 多層フィードフォワードネットワーク

### 2.1 通常のネットワークモデル

多層フィードフォワードネットワークの入力  $\mathbf{x} \in \mathbb{R}^N$  に対する出力  $\mathbf{y} \in \mathbb{R}^M$  は以下の式で表すことができる。

$$\mathbf{y} = \phi_m(\mathbf{W}_m \phi_{m-1}(\mathbf{W}_{m-1} \phi_{m-2}(\cdots \phi_1(\mathbf{W}_1 \mathbf{x} + \mathbf{b}_1) \cdots) + \mathbf{b}_{m-1}) + \mathbf{b}_m) \quad (1)$$

ここで  $\mathbf{W}_i$ ,  $\mathbf{b}_i$  はそれぞれ  $i$  層の重みとバイアスを表す行列とベクトルであり、 $\phi_i$  は  $i$  層の活性化関数の出力を表すベクトル関数である。

古典的な普遍性定理では、活性化関数にシグモイド関数

$$s(x) = \frac{1}{1 + e^{-x}} \quad (2)$$

を用いたとき、 $m=2$ 、すなわち高々 2 層のネットワークで任意の関数を近似することができることを証明している。DNN においては、活性化関数はシグモイドのような有界な関数ではなく、ReLU (Rectified Linear Unit) 関数

$$r(x) = \max(0, x) \quad (3)$$

を代表とする深いモデルの学習に適した非有界な関数を用いる。

$i$  層における 1 つのニューロンに注目すると、その入力  $\mathbf{x}_i \in \mathbb{R}^{N_i}$  に対する出力  $y_j$  は

$$y_j = \phi_i(\mathbf{w}_j^t \mathbf{x}_i + b_j) \quad (4)$$

と表される。但し、 $i$  層の出力は  $\mathbf{y}_i \in \mathbb{R}^{M_i}$ ,  $j \in \{1, 2, \dots, M_i\}$  であり、また  $\mathbf{w}_j^t$  と  $b_j$  はそれぞれ  $\mathbf{W}_i$  と  $\mathbf{b}_i$  の  $j$  番目の行と要素である。 $\mathbf{x}_i$  で構成される特徴空間において、重み  $\mathbf{w}_j$  は原点を通る超平面を規定し、バイアス  $b_j$  は超平面を原点以外に平行移動させるパラメタとなる。図 1 に示すように、ニューロンの出力値  $y_j$  は活性化関数によってこの超平面を境に傾斜状に変化する。

### 2.2 バイアス項を取り除いたネットワークモデル

バイアス項を取り除いたニューロンを階層的に複数組み合わせることで式 (4) と同じ出力値を作り出すことができる。構成はさまざまに考えられるが、簡単には 3 つのバイアス項を取り除いたニューロンを 2 層で組み合わせ、バイアス項有りの 1 つのニューロンを代替することで実現できる。図 2(a) にその構成を示す。計算式は

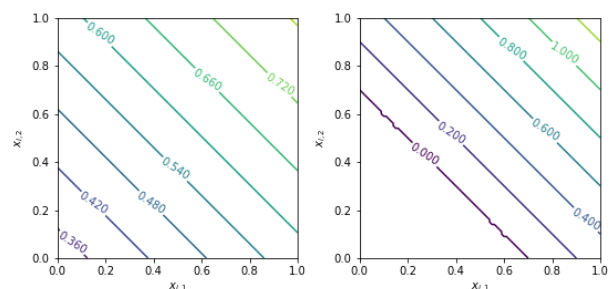


図 1 ニューロンの出力値の変化

$y_j = \phi(x_{i,1} + x_{i,2} - 0.7)$  の等高線図を表している。左は  $\phi(x) = s(x)$ , 右は  $\phi(x) = r(x)$  をそれぞれ採用した場合である。

<sup>†</sup> 東京学芸大学 技術・情報科学講座 情報科学分野  
Tokyo Gakugei University, Department of Technology and  
Information Science

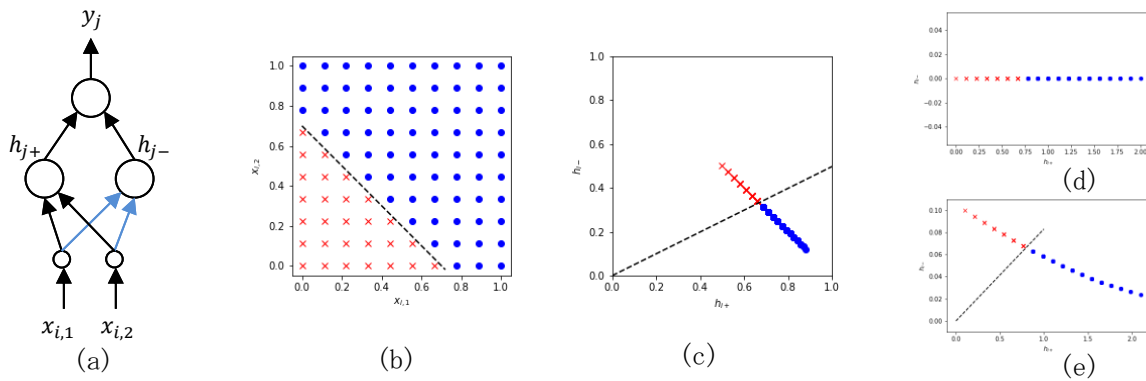


図2 バイアス項の無いニューロンで構成したネットワークモデルの例

(a) 2層モデルの例。(b) 鎖線で示す  $x_{i,1} + x_{i,2} - 0.7 = 0$  が境界。○と×で表す点群が入力として与えられる。(c) シグモイド関数を採用した場合の中間層の出力の様子。出力層で破線で線形に分離できる。(d) ReLU 関数を採用した場合の中間層の出力の様子。出力層で線形に分離できない。(e) 式(7)に示す関数 ( $\gamma = 0.1$ ) を採用した場合の中間層の出力の様子。出力層は破線で線形に分離できる。

$$\begin{aligned} h_{j+} &= \phi_i(\mathbf{w}_{j+}^t \mathbf{x}_i) \\ h_{j-} &= \phi_i(\mathbf{w}_{j-}^t \mathbf{x}_i) \\ \mathbf{h}_j &= [h_{j+}, h_{j-}]^t \\ y_j &= \phi_i(\mathbf{w}_j^t \mathbf{h}_j) \end{aligned} \quad (5)$$

と書ける。但し、

$$\mathbf{w}_{j+} = -\mathbf{w}_{j-} \quad (6)$$

である。また、活性化関数  $\phi_i(x)$  は  $\phi_i(x) \geq 0$  を満たす単調増加関数であるか、あるいは  $x \rightarrow -\infty$  のとき  $\phi(x) \rightarrow 0$  であるなら  $x < 0$  では  $\phi_i(x) \leq 0$  を満たす単調減少関数であってもよい。

図2(b)に示す境界面を実現するには、 $\mathbf{w}_{j+} = [1, 1]^t$  とし、式(2)のシグモイド関数を活性化関数とすればよい。このときの中間層の出力  $h_{j+}$  と  $h_{j-}$  をプロットしたものが図2(c)である。出力層のニューロンにおいて破線で示す位置で線形に、つまりバイアス項無しに重み  $w_j$  だけで分離可能であることが分かる。

式(3)の ReLU 関数は前述の条件を満たさないため、図2(d)に示すように出力層でバイアス項無しには線形に分離できない。しかし、例えば  $\gamma$  を小さな値として

$$r_+(x) = \begin{cases} s(x) \frac{\gamma}{2}, & x \leq 0 \\ r(x) + \gamma, & x > 0 \end{cases} \quad (7)$$

のように式(2)と式(3)を組み合わせることでバイアス項無しのネットワーク構成に適した活性化関数となる。図2(e)は  $\gamma = 0.1$  としたときの中間層の出力の様子である。

### 2.3 実験

活性化関数に式(7)を用いたニューロン2個から成る線形層を2層、これに式(2)を活性化関数として用いたニューロン1個の線形層を重ねたバイアス項の一切無いネットワークモデル (2-2-1) に対し、XNOR を学習させた結果を図3に示す。最初の線形層の重みが式(6)の関係を概ね満たしており、図2(c)と同じ方向で線形分離を試みていることが分かる。但し、原点方向に大きく歪んだ曲線になっているのは式(7)の効果である。原点付近の値がXNORの「0」、線の端が「1」に対応するのだが、まだバイアス無しでは線形分離できない。次の線形層の出力で初めてバイアス無しで線形分離可能となっていることが分かる。

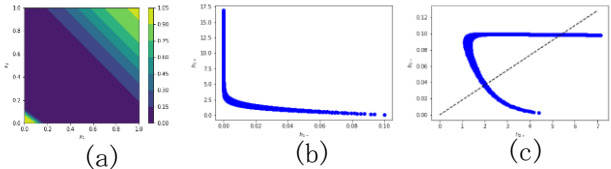


図3 XNOR の学習結果

(a) 学習後の出力の等高線図。(b) 1層目の出力の様子。(c) 2層目の出力の様子。出力層では破線で線形に分離できる。

ニューロンの数 100-100-100-10 の4層構成で MNIST データセットを用いた分類実験も行った。バイアス項無しで活性化関数に ReLU を用いた場合のテスト認識率が 96.92% であるのに対し、式(7)を活性化関数として用いた場合は 97.16% と微少ながら上回った。なお、通常のバイアス項有りで ReLU を用いた場合は 96.73% であった。(いずれも 20 エポック学習を 10 回行ったときの平均値)

### 3. おわりに

バイアス項を全て取り除いた多層フィードフォワードネットワークにおいても、特定の条件を満たす活性化関数を用いることで関数近似能力の普遍性を失わずに済むことを明らかにした。また、簡単な分類実験の結果から、通常のバイアス項有りのモデルと同等の性能を有する可能性があることが示唆された。

#### 参考文献

- [1] Yann LeCun, Yoshua Bengio, and Geoffrey Hinton, "Deep learning," Nature, vol. 521, pp. 436-444, (2015).
- [2] Samira Pouyanfar, Saad Sadiq, Yilin Yan, et al., "A Survey on Deep Learning: Algorithms, Techniques, and Applications," ACM Computing Surveys, vol. 51, no. 5, Article 92, 36 pages, (2018).
- [3] Christoph Molnar, "Interpretable Machine Learning: A Guide for Making Black Box Models Explainable," (2018). [Online] Available: <https://christophm.github.io/interpretable-ml-book/> [Accessed: May 28, 2021].
- [4] Shengjie Wang, Tianyi Zhou, and Jeff Bilmes, "Bias Also Matters: Bias Attribution for Deep Neural Network Explanation," Proceedings of the 36th International Conference on Machine Learning, in Proceedings of Machine Learning Research 97, pp. 6659-6667, (2019).
- [5] Sreyas Mohan, Zahra Kadkhodaie, Eero P. Simoncelli, et al., "Robust And Interpretable Blind Image Denoising Via Bias-Free Convolutional Neural Networks," In International Conference on Learning Representations, (2020).