G-007

深層残差ネットワークの H_∞ 学習

The H_∞-Learning of Deep Residual Networks

西山 清　菅原 康滉 *
Kiyoshi NISHIYAMA and Yasuhiro SUGAWARA

岩手大学理工学部システム創成工学科
Faculty of Science and Engineering, Iwate University

1　はじめに

近年、深層ニューラルネットワーク [1] は画像認識や音声認識の分野などにおいて画期的な成果をあげている。本研究では、H_∞ ノルムを目的関数にもつ H_∞ 学習が深層化した残差ネットワークの学習に非常に有効であることを明らかにする。

2　H_∞ 学習

H_∞ 学習問題とは、$\gamma_f > 0$ が与えられたとき、

$$\sup_{w_0,\{\boldsymbol{v}_p\}} \frac{\sum_{p=0}^{k}\|\boldsymbol{e}_{f,p}\|^2_{(\sigma_v^2\boldsymbol{I})^{-1}}}{\|\boldsymbol{w}-\check{\boldsymbol{w}}_0\|^2_{\boldsymbol{\Sigma}_0^{-1}}+\sum_{p=0}^{k}\|\boldsymbol{v}_p\|^2_{(\sigma_v^2\boldsymbol{I})^{-1}}} < \gamma_f^2 \quad (1)$$

を満たす H_∞ 準最適な学習アルゴリズム $\mathcal{F}_f$ を求める問題である。ここで、$\boldsymbol{e}_{f,p}$ は出力誤差、$\boldsymbol{w}$ は重みベクトル、$\boldsymbol{v}_p$ は線形化誤差等である。 ■

この H_∞ 学習問題の解は、ニューラルネットワークを線形化した状態空間モデルに H_∞ フィルタ (EHF) [2] を適用して得られる。この学習アルゴリズムはニューラルネットワーク全体に関して H_∞ 準最適な解を与えることから、g-EHF 学習アルゴリズムと呼ばれる [3]。次に、出力層のニューロン数が一つの場合を示す。

$$\begin{aligned}
\hat{\boldsymbol{w}}_{k+1} &= \hat{\boldsymbol{w}}_k + \boldsymbol{K}_{s,k+1}(y_{k+1} - h_{k+1}(\hat{\boldsymbol{w}}_k)) \quad (2)\\
\boldsymbol{K}_{s,k+1} &= \hat{\boldsymbol{P}}_{k+1|k}\boldsymbol{H}_{k+1}^T(\boldsymbol{H}_{k+1}\hat{\boldsymbol{P}}_{k+1|k}\boldsymbol{H}_{k+1}^T+1)^{-1}\\
\hat{\boldsymbol{P}}_{k+1|k} &= \hat{\boldsymbol{P}}_{k|k-1} - \hat{\boldsymbol{P}}_{k|k-1}\\
&\quad\times\begin{bmatrix}\boldsymbol{H}_k^T & \boldsymbol{H}_k^T\end{bmatrix}\boldsymbol{R}_{\boldsymbol{e},k}^{-1}\begin{bmatrix}\boldsymbol{H}_k\\ \boldsymbol{H}_k\end{bmatrix}\hat{\boldsymbol{P}}_{k|k-1} \quad (3)
\end{aligned}$$

ただし、

$$\begin{aligned}
\boldsymbol{H}_k &= \left.\frac{\partial h_k(\boldsymbol{w})}{\partial \boldsymbol{w}}\right|_{\boldsymbol{w}=\hat{\boldsymbol{w}}_{k-1}}, \hat{\boldsymbol{P}}_{k|k-1} = \hat{\boldsymbol{\Sigma}}_{k|k-1}/\sigma_v^2 \quad (4)\\
\boldsymbol{R}_{\boldsymbol{e},k} &= \boldsymbol{R} + \begin{bmatrix}\boldsymbol{H}_k\\ \boldsymbol{H}_k\end{bmatrix}\hat{\boldsymbol{P}}_{k|k-1}\begin{bmatrix}\boldsymbol{H}_k^T & \boldsymbol{H}_k^T\end{bmatrix}\\
\boldsymbol{R} &= \begin{bmatrix}1 & 0\\ 0 & -\gamma_f^2\end{bmatrix} \quad (5)
\end{aligned}$$

*現東芝デジタルソリューションズ株式会社

3　残差ネットワーク (ResNet)

ResNet は、ある 2 つの層間の出力をショートカット接続 (shortcut connection) で結合した構造を含んだニューラルネットワーク（NN）である [4]。ショートカット接続とは、ある NN における l 層の出力 $\boldsymbol{x} \in \mathcal{R}^M$ と $l+m$ 層の出力 $\boldsymbol{y} \in \mathcal{R}^M$ を加算することである。その和を $\boldsymbol{z} = \boldsymbol{y}+\boldsymbol{x}$ とする。なお、$\mathcal{R}$ は実数全体の集合、l, m, M は自然数である。このショートカット接続により、$\boldsymbol{z}$ を学習する問題は残差 $\boldsymbol{y} = \boldsymbol{z} - \boldsymbol{x}$ を学習する問題に帰着できる。このことから、このショートカット接続を含む NN は残差ネットワーク (ResNet) と呼ばれる。

本研究では、図 1 のような L 個の層を持ち、各隠れ層のニューロン数が同じである、ResNet について考える。この ResNet の層の数 L はショートカット接続の数 n で決定される ($L = 2n+3,\ n = 1,2,\ldots$)。例えば、ショートカット接続が $n = 3$ であるとき層の数は $L = 9$ となる。図 1 中の四角のニューロンは、応答関数が恒等写像であり、しきい値を持たないことを表す。一方、丸のニューロンは応答関数がシグモイド関数 $f(x) = 1/(1+\exp(-\eta_0 x))$ であり、しきい値をもつことを表す ($\eta_0 > 0$ はシグモイド関数の傾き)。図 1 中の弧線はショートカット接続を表す。

この L 層 ResNet に p 番目の入力 $\boldsymbol{z}^1[p] = \left[z_1^1[p],\ldots,z_{N_1}^1[p]\right]^T \in \mathcal{R}^{N_1\times 1}$ が与えられたとき、l 層の出力 $\boldsymbol{z}^l[p] = \left[z_1^l[p],\ldots,z_{N_l}^l[p]\right]^T \in \mathcal{R}^{N_l\times 1}$ を以下のように定める。

$$\begin{aligned}
\boldsymbol{z}^l[p] &= \boldsymbol{f}(\boldsymbol{s}^l),\ \ \boldsymbol{s}^l = \boldsymbol{W}^{l-1}\boldsymbol{z}^{l-1}[p] + \boldsymbol{b}^l\\
&\qquad (l = 2 \text{ または } l = 3,5,\ldots,L) \quad (6)\\
\boldsymbol{z}^l[p] &= \boldsymbol{s}^l,\ \ \boldsymbol{s}^l = \boldsymbol{W}^{l-1}\boldsymbol{z}^{l-1}[p] + \boldsymbol{z}^{l-2}[p]\\
&\qquad (l = 4,6,\ldots,L-1) \quad (7)
\end{aligned}$$

ここで、N_l は l 層のニューロン数、$\boldsymbol{s}^l = \left[s_1^l,\ldots,s_{N_l}^l\right]^T \in \mathcal{R}^{N_l\times 1}$ は l 層の膜電位、

$$\boldsymbol{f}(\boldsymbol{s}^l) = \left[f(s_1^l),\ldots,f(s_{N_l}^l)\right]^T \quad (8)$$

は l 層の膜電位 $\boldsymbol{s}^l$ の各成分に対するニューロンの出力から成るベクトル値関数である。また、

$$\boldsymbol{W}^l = \begin{bmatrix} w_{1,1}^l & \cdots & w_{1,N_l}^l\\ \vdots & \ddots & \vdots\\ w_{N_{l+1},1}^l & \cdots & w_{N_{l+1},N_l}^l\end{bmatrix} \in \mathcal{R}^{N_{l+1}\times N_l} \quad (9)$$

は $l+1$ 層、l 層間の重み行列、$\boldsymbol{b}^l = \left[w_{1,0}^{l-1}, \ldots, w_{N_l,0}^{l-1}\right]^T \in \mathcal{R}^{N_l}$ は l 層のしきい値ベクトルである。

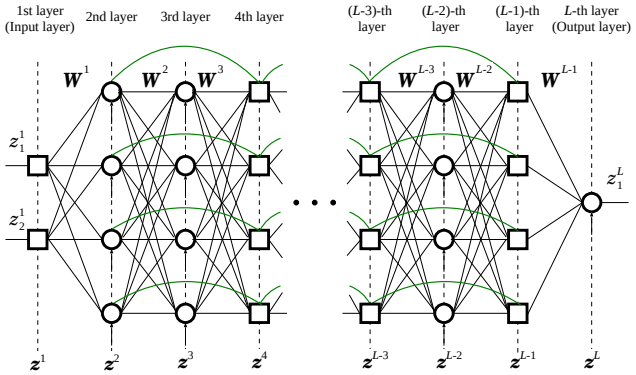


図 1　L 層 ResNet；L は層の数であり（$L = 2n+3,\ n = 1, 2, \ldots$）、隠れ層のニューロン数 $N_l,\ l = 2, \ldots, L-1$ は等しい。$\boldsymbol{z}^l$ は l 層の出力、$\boldsymbol{W}^l$ は $l+1$ 層、l 層間の重み行列である。四角のニューロンは、応答関数が恒等写像であり、しきい値を持たないことを表す。一方、丸のニューロンは応答関数がシグモイド関数であり、しきい値を持つことを表す。l を 0 ではない偶数とするとき、l 層と $l+2$ 層はショートカット接続されている。

4　ショートカットを考慮した H_∞ 学習

L 層 ResNet における H_∞ 学習は、文献 [5] で述べた深層 NN の H_∞ 学習とほとんど同じである。異なるのは、NN の線形状態空間モデル

$$\boldsymbol{w}_{k+1} = \boldsymbol{w}_k,\ \boldsymbol{m}_k = \boldsymbol{H}_k \boldsymbol{w}_k + \boldsymbol{v}_k \tag{10}$$

における観測行列

$$\boldsymbol{H}_k = \left.\frac{\partial \boldsymbol{h}_k(\boldsymbol{w})}{\partial \boldsymbol{w}}\right|_{\boldsymbol{w}=\hat{\boldsymbol{w}}_{k-1}} \in \mathcal{R}^{N_L \times N_w} \tag{11}$$

$$= \left.\left[\frac{\partial \boldsymbol{h}_k}{\partial w_{1,0}^1}, \cdots, \frac{\partial \boldsymbol{h}_k}{\partial w_{N_L,N_{L-1}}^{L-1}}\right]\right|_{\boldsymbol{w}=\hat{\boldsymbol{w}}_{k-1}} \tag{12}$$

$$= \left.\begin{bmatrix} \frac{\partial h_{k,1}}{\partial w_{1,0}^1} & \cdots & \frac{\partial h_{k,1}}{\partial w_{N_L,N_{L-1}}^{L-1}} \\ \frac{\partial h_{k,2}}{\partial w_{1,0}^1} & \cdots & \frac{\partial h_{k,2}}{\partial w_{N_L,N_{L-1}}^{L-1}} \\ \vdots & \ddots & \vdots \\ \frac{\partial h_{k,N_L}}{\partial w_{1,0}^1} & \cdots & \frac{\partial h_{k,N_L}}{\partial w_{N_L,N_{L-1}}^{L-1}} \end{bmatrix}\right|_{\boldsymbol{w}=\hat{\boldsymbol{w}}_{k-1}} \tag{13}$$

の計算方法だけである。ここで、

$$\boldsymbol{w} = [w_{1,0}^1, w_{1,1}^1, \ldots, w_{N_2,N_1}^1, \ldots \\ \ldots, w_{1,0}^{L-1}, w_{1,1}^{L-1}, \ldots, w_{N_L,N_{L-1}}^{L-1}]^T \in \mathcal{R}^{N_w} \tag{14}$$

はすべてのしきい値と結合重みからなる重みベクトル、$\boldsymbol{h}_k(\boldsymbol{w}) = [h_{k,1}(\boldsymbol{w}), \ldots, h_{k,N_L}(\boldsymbol{w})]^T \in \mathcal{R}^{N_L \times 1}$ は時刻 k の L 層 ResNet の出力 $\boldsymbol{z}^L$、$\hat{\boldsymbol{w}}_{k-1}$ は時刻 $k-1$ における重みベクトル $\boldsymbol{w}$ の推定値である。

観測行列 $\boldsymbol{H}_k$ 中の $\frac{\partial \boldsymbol{h}_k}{\partial w_{j,i}^1}$ は次式により計算される。

$$\frac{\partial \boldsymbol{h}_k}{\partial w_{j,i}^1} = \frac{\partial \boldsymbol{z}^L}{\partial \boldsymbol{s}^L} \cdot \frac{\partial \boldsymbol{s}^L}{\partial \boldsymbol{z}^{L-1}} \frac{\partial \boldsymbol{z}^{L-1}}{\partial \boldsymbol{s}^{L-1}} \cdots \\ \cdots \frac{\partial \boldsymbol{s}^4}{\partial \boldsymbol{z}^3} \frac{\partial \boldsymbol{z}^3}{\partial \boldsymbol{s}^3} \cdot \frac{\partial \boldsymbol{s}^3}{\partial \boldsymbol{z}^2} \frac{\partial \boldsymbol{z}^2}{\partial \boldsymbol{s}^2} \cdot \frac{\partial \boldsymbol{s}^2}{\partial w_{j,i}^1} \tag{15}$$

$$= \boldsymbol{\Phi}^L \cdot \frac{\partial \boldsymbol{s}^L}{\partial \boldsymbol{z}^{L-1}} \frac{\partial \boldsymbol{z}^{L-1}}{\partial \boldsymbol{s}^{L-1}} \cdots \\ \cdots \frac{\partial \boldsymbol{s}^4}{\partial \boldsymbol{z}^3} \frac{\partial \boldsymbol{z}^3}{\partial \boldsymbol{s}^3} \cdot \frac{\partial \boldsymbol{s}^3}{\partial \boldsymbol{z}^2} \frac{\partial \boldsymbol{z}^2}{\partial \boldsymbol{s}^2} \cdot \frac{\partial \boldsymbol{s}^2}{\partial w_{j,i}^1} \tag{16}$$

$$\vdots \tag{17}$$

$$= \boldsymbol{\Phi}^2 \cdot \frac{\partial \boldsymbol{s}^2}{\partial w_{j,i}^1} \tag{18}$$

$$\frac{\partial \boldsymbol{h}_k}{\partial w_{j,i}^2} = \boldsymbol{\Phi}^3 \cdot \frac{\partial \boldsymbol{s}^3}{\partial w_{j,i}^2},\ \cdots,\ \frac{\partial \boldsymbol{h}_k}{\partial w_{j,i}^l} = \boldsymbol{\Phi}^{l+1} \cdot \frac{\partial \boldsymbol{s}^{l+1}}{\partial w_{j,i}^l} \tag{19}$$

ここで、変数 $\boldsymbol{\Phi}^l$ は次の逆方向の再帰式で得られる。

$$\left(\boldsymbol{\Phi}^l\right)^T = \left(\boldsymbol{\Phi}^{l+1} \frac{\partial \boldsymbol{s}^{l+1}}{\partial \boldsymbol{z}^l} \frac{\partial \boldsymbol{z}^l}{\partial \boldsymbol{s}^l}\right)^T \in \mathcal{R}^{N_l \times N_L} \tag{20}$$

$$= \left(\boldsymbol{\Phi}^{l+1} \boldsymbol{W}^l \begin{bmatrix} \frac{\partial z_1^l}{\partial s_1^l} & & \boldsymbol{O} \\ & \ddots & \\ \boldsymbol{O} & & \frac{\partial z_{N_l}^l}{\partial s_{N_l}^l} \end{bmatrix}\right)^T \tag{21}$$

$$= \begin{bmatrix} \frac{\partial z_1^l}{\partial s_1^l} & & \boldsymbol{O} \\ & \ddots & \\ \boldsymbol{O} & & \frac{\partial z_{N_l}^l}{\partial s_{N_l}^l} \end{bmatrix} \left(\boldsymbol{\Phi}^{l+1} \boldsymbol{W}^l\right)^T \tag{22}$$

$(l = L-1$ または $l = L-2, \ldots, 5, 3)$

$$\left(\boldsymbol{\Phi}^l\right)^T = \left(\left(\boldsymbol{\Phi}^{l+1} \frac{\partial \boldsymbol{s}^{l+1}}{\partial \boldsymbol{z}^l} + \boldsymbol{\Phi}^{l+2}\right) \frac{\partial \boldsymbol{z}^l}{\partial \boldsymbol{s}^l}\right)^T \tag{23}$$

$$= \left(\left(\boldsymbol{\Phi}^{l+1} \boldsymbol{W}^l + \boldsymbol{\Phi}^{l+2}\right) \begin{bmatrix} \frac{\partial z_1^l}{\partial s_1^l} & & \boldsymbol{O} \\ & \ddots & \\ \boldsymbol{O} & & \frac{\partial z_{N_l}^l}{\partial s_{N_l}^l} \end{bmatrix}\right)^T$$

$$= \begin{bmatrix} \frac{\partial z_1^l}{\partial s_1^l} & & \boldsymbol{O} \\ & \ddots & \\ \boldsymbol{O} & & \frac{\partial z_{N_l}^l}{\partial s_{N_l}^l} \end{bmatrix} \left(\boldsymbol{\Phi}^{l+1} \boldsymbol{W}^l + \boldsymbol{\Phi}^{l+2}\right)^T \tag{24}$$

$(l = L-3, L-5, \ldots, 4, 2)$

ただし、

$$\left(\boldsymbol{\Phi}^L\right)^T = \left(\frac{\partial \boldsymbol{z}^L}{\partial \boldsymbol{s}^L}\right)^T \in \mathcal{R}^{N_L \times N_L} \quad (25)$$

$$\frac{\partial \boldsymbol{z}^l}{\partial \boldsymbol{s}^l} = \begin{bmatrix} \frac{\partial z_1^l}{\partial s_1^l} & & O \\ & \ddots & \\ O & & \frac{\partial z_{N_l}^l}{\partial s_{N_l}^l} \end{bmatrix} \in \mathcal{R}^{N_l \times N_l} \quad (26)$$

ここで、$\partial \boldsymbol{z}^l / \partial \boldsymbol{s}^l$ は対角行列であり、$l = L-1, L-3, \ldots, 6, 4$ であるとき単位行列となる。また、

$$\frac{\partial \boldsymbol{s}^{l+1}}{\partial \boldsymbol{z}^l} = \boldsymbol{W}^l \in \mathcal{R}^{N_{l+1} \times N_l} \quad (27)$$

である。

特に、L 層 ResNet の出力の次元数が $N_L = 1$ であるとき、$\left(\boldsymbol{\Phi}^L\right)^T$ はスカラー $\boldsymbol{\phi}^L$ となり、$\left(\boldsymbol{\Phi}^l\right)^T$ は N_l 次元の列ベクトル $\left(\boldsymbol{\phi}^l\right)^T$ となるため、アマダール積 $\odot$ を用いて次のように書き換えられる。

$$\left(\boldsymbol{\phi}^l\right)^T = \left(\boldsymbol{\phi}^{l+1}\boldsymbol{W}^l\right)^T \odot \begin{bmatrix} \frac{\partial z_1^l}{\partial s_1^l} \\ \vdots \\ \frac{\partial z_{N_l}^l}{\partial s_{N_l}^l} \end{bmatrix} \quad (28)$$

$$(l = L-1 \text{ または } l = L-2, \ldots, 5, 3)$$

$$\left(\boldsymbol{\phi}^l\right)^T = \left(\boldsymbol{\phi}^{l+1}\boldsymbol{W}^l + \boldsymbol{\phi}^{l+2}\right)^T \odot \begin{bmatrix} \frac{\partial z_1^l}{\partial s_1^l} \\ \vdots \\ \frac{\partial z_{N_l}^l}{\partial s_{N_l}^l} \end{bmatrix} \quad (29)$$

$$(l = L-3, L-5, \ldots, 6, 4, 2)$$

ただし、

$$\boldsymbol{\phi}^L = \frac{\partial z_1^L}{\partial s_1^L} \in \mathcal{R} \quad (30)$$

5　シミュミレーション

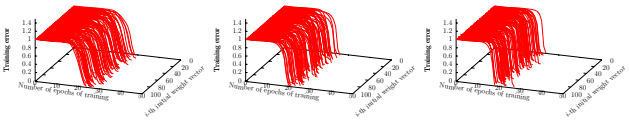
排他的論理和（XOR）問題に対して L 層残差ネットワーク (ResNet) を用いて H_∞ 学習（$\gamma_f = 1.7$）を行ったときの学習過程について考察する。そのため、100 本の学習曲線と、学習終了時における学習回数の統計量を、それぞれ図 2 と表 1 に示した。これより、深層化することによって学習回数が単調に減少することがわかる。この際、深層 H_∞ 学習は、１）初期重みに依存しない；プレトレーニング不要、２）バッチノーマライゼイション不要; 勾配消失問題なし、３）ReLU 不要；シグモイド関数可、４）中間層のニューロン数の調整不要、などの特徴を備えていた。

6　まとめ

論理関数である XOR を深層残差ネットワークで H_∞ 学習した結果、１００１層までパラメータの調整を一切しなくても、ランダムに選んだ１００種類の初期重みに対してすべて２０回程度の更新で学習が終了する画期的な成果を得た。今後は深層 H_∞ 学習のメカニズムを解明したい。

表 1　L 層 ResNet における学習終了時の学習回数に関する統計量 (重みは区間 $[-0.05, 0.05]$ の一様分布により初期化)。目的関数は H_∞ ノルム (最大エネルギーゲイン)、学習法は g-EHF 法 ($\gamma_f = 1.7$)、訓練集合は排他的論理和の真理値表、入力層のニューロン数は $N_1 = 2$、隠れ層のニューロン数は $N_l = 5$ ($l = 2, \ldots, L-1$)、出力層のニューロン数は $N_L = 1$、隠れ層および出力層の応答関数はシグモイド関数、シグモイド関数の傾きは $\eta_0 = 2.5$、打ち切り誤差は 10^{-2}。

層数 L	学習回数の平均	学習回数の標準偏差
5	25.92	2.59
101	23.17	2.36
201	22.58	2.31
401	21.74	1.96
801	20.74	2.39
1001	20.57	2.23



(a)$L = 5$　(b)$L = 201$　(c)$L = 1001$

図 2　L 層 ResNet における 100 本の学習曲線 (重みは区間 $[-0.05, 0.05]$ の一様分布により初期化)。目的関数は H_∞ ノルム、学習法は g-EHF 法 ($\gamma_f = 1.7$)、訓練集合は排他的論理和の真理値表、入力層のニューロン数は $N_1 = 2$、隠れ層のニューロン数は $N_l = 5$ ($l = 2, \ldots, L-1$)、出力層のニューロン数は $N_L = 1$、隠れ層および出力層の応答関数はシグモイド関数、シグモイド関数の傾きは $\eta_0 = 2.5$、打ち切り誤差は 10^{-2}。

参考文献

[1] M. A. Nielsen, “Neural networks and deep learning,” Determination Press, 2015.

[2] 西山 清, 最適フィルタリング, 培風館, 2001.

[3] K. Nishiyama and K. Suzuki, “H_∞-learning of layered neural networks,” IEEE Trans. Neural Networks, 12, 6, pp.1265-1277, 2001.

[4] K. He, X. Zhang, S. Ren, and J. Sun, “Deep residual learning for image recognition,” Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp.770-778, 2016.

[5] 菅原 康滉, 西山 清, “H_∞ 学習の深層ニューラルネットワークへの拡張,” 電子情報通信学会ニューロコンピューティング研究会, NC2019-92, 2020.