F-017

深層ニューラルネットワークにおける鞍点問題を攻略するための深層残差 H_∞ 学習

The Deep Residual H_∞-Learning for Attacking the Saddle Point Problem in Deep Neural Networks

西山 清
Kiyoshi NISHIYAMA

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

多層ニューラルネットワーク [1] において層を深くして行くと、ある深さから精度が著しく劣化することが知られている。この対策としてスキップコネクション（ショートカット接続）が導入された残差ネットワーク（Residual Network）が考案された [2]。残差ネットワークでは層を深くすればするほど、学習時の精度が向上することが報告されている。スキップコネクションをもつ残差ネットワークでは、目標関数そのものを学習するのではなく、目標関数と入力の残差を学習する。よって、目標関数が恒等関数に近いときでも非線形ネットワークで容易に学習できると考えられている。

果して劣化問題をスキップコネクションで攻略できる理由はそれだけであろうか？

この疑問に答えるため、本研究では全く異なった視点から深層化に対する劣化問題を捉え、新たな攻略法について考察したい。

2　スキップコネクションと誤差曲面の形状

深層ニューラルネットワークの学習における誤差曲面の停留点の多くは鞍点（saddle point）であると考えられている。特に、図 1 のような monkey 鞍点（monkey saddle）の場合、$x = 0$ の直線上の任意の点 $(0, y)$ で y 軸方向の傾き（1 次微分）が 0 となる。2 次微分（ヘッセ行列）まで考慮した準ニュートン法はこの鞍点にトラップされる [3]。

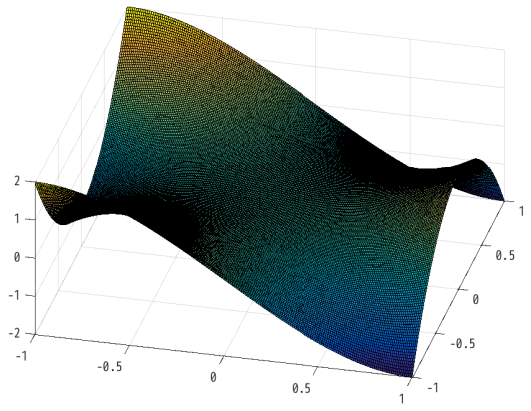


図 1　monkey 鞍点： $f(x, y) = x^3 - 3xy^2$.

図 2 の残差ネットワークで第 2 層から第 4 層が恒等関数であるときを考える。このとき、$\boldsymbol{W}^2 = \boldsymbol{0}$ で $\boldsymbol{W}^3$ は任意、あるいは $\boldsymbol{W}^3 = \boldsymbol{0}$ で $\boldsymbol{W}^2$ は任意となり、残差ネットワークの誤差曲面には図 1 のような monkey 鞍点が現れることが予想される。よって、図 2 のような深層残差ネットワークでは、高次元の重み空間上に monkey 鞍点が幾重にも重なる誤差曲面となることが推測される。

このような誤差曲面を攻略する最適化法（optimizer）として、最近急速に普及しているのが、Adam（Adaptive moment estimation）である [4]。Adam は 1 次最適化法であり、1 回の更新当たりの計算量が少ない。一方、2 次最適化法としてはサドルフリーニュートン法が知られている [3]。この方法は固有値の計算が必要であり、計算量に難点がある。

3　スキップコネクションとモデル集合

深層ニューラルネットワークにスキップコネクションを導入すれば、層を深くすればするほど精度を向上することが可能となった。これはなぜであろうか？この理由についてモデル集合の観点から考えてみたい。まず、図 2 の残差ネットワークが 3 層の場合を考え、入力を $\boldsymbol{x}$、隠れ層のニューロン数を N、重みパラメータを $\boldsymbol{W}_{2-N-1}$ とすれば、ネットワークで表せる関数は $f(\boldsymbol{x}; \boldsymbol{W}_{2-N-1})$ となる。重み空間内の一つの点に対してある関数が対応し、これより 3 層ネットワークで表現できる関数の集合が得られる。この集合を $\{f(\boldsymbol{x}; \boldsymbol{W}_{2-N-1})\}$ で表す。層数を 5、7、…と増やすとスキップコネクションの存在からそれぞれの関数集合の間には次の包含関係が成り立つ。

$$\begin{aligned} \{f(\boldsymbol{x}; \boldsymbol{W}_{2-N-1})\} &\subseteq \{f(\boldsymbol{x}; \boldsymbol{W}_{2-N-N-N-1})\} \\ &\subseteq \{f(\boldsymbol{x}; \boldsymbol{W}_{2-N-N-N-N-N-1})\} \\ &\vdots \end{aligned} \tag{1}$$

すなわち、深い層の残差ネットワークは必ず浅い層の残差ネットワークを含み、層数を増やせば増やすほど表現できる関数のクラスが広がることがわかる。

次の課題は、深層化によって関数空間の包含関係を内包しつつ構築された特殊な構造をもった部分重み空間の中で最適な重みベクトルを効率的に探索することである。

本研究では、この課題解決のために H_∞ 学習を取り上げる。

4 H_∞ 学習

H_∞ 学習問題とは、$\gamma_f > 0$ が与えられたとき、

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

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、$\boldsymbol{e}_{f,p} = \boldsymbol{y}_p - \boldsymbol{h}_p(\boldsymbol{w})$ 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) [5] を適用して得られる。この学習アルゴリズムは 1 次微分のみを用いて重み空間全体を探索して H_∞ 準最適な解を求めることから、g-EHF 学習アルゴリズムと呼ばれる [6]。次に、出力層のニューロン数が一つの場合を示す。

$$\hat{\boldsymbol{w}}_{k+1} = \hat{\boldsymbol{w}}_k + \boldsymbol{K}_{s,k+1}(y_{k+1} - h_{k+1}(\hat{\boldsymbol{w}}_k)) \quad (3)$$

$$\boldsymbol{K}_{s,k+1} = \hat{\boldsymbol{P}}_{k+1|k}\boldsymbol{H}_{k+1}^T(\boldsymbol{H}_{k+1}\hat{\boldsymbol{P}}_{k+1|k}\boldsymbol{H}_{k+1}^T+1)^{-1}$$

$$\hat{\boldsymbol{P}}_{k+1|k} = \hat{\boldsymbol{P}}_{k|k-1} - \hat{\boldsymbol{P}}_{k|k-1} \times \begin{bmatrix} \boldsymbol{H}_k^T & \boldsymbol{H}_k^T \end{bmatrix} \boldsymbol{R}_{\boldsymbol{e},k}^{-1} \begin{bmatrix} \boldsymbol{H}_k \\ \boldsymbol{H}_k \end{bmatrix} \hat{\boldsymbol{P}}_{k|k-1} \quad (4)$$

ただし、

$$\boldsymbol{H}_k = \left.\frac{\partial h_k(\boldsymbol{w})}{\partial \boldsymbol{w}}\right|_{\boldsymbol{w}=\hat{\boldsymbol{w}}_{k-1}}, \hat{\boldsymbol{P}}_{0|-1} = \epsilon_0\boldsymbol{I},\ \epsilon_0 > 0 \quad (5)$$

$$\boldsymbol{R}_{\boldsymbol{e},k} = \boldsymbol{R} + \begin{bmatrix} \boldsymbol{H}_k \\ \boldsymbol{H}_k \end{bmatrix} \hat{\boldsymbol{P}}_{k|k-1} \begin{bmatrix} \boldsymbol{H}_k^T & \boldsymbol{H}_k^T \end{bmatrix}$$

$$\boldsymbol{R} = \begin{bmatrix} 1 & 0 \\ 0 & -\gamma_f^2 \end{bmatrix}, \quad \gamma_f > 1 \quad (6)$$

5 H_∞ 学習の挙動解析

リカッチ方程式の解 $\hat{\boldsymbol{P}}_{k+1|k}$ の逆行列は逆行列の補助定理を用いれば次のように表すことができる。

$$\begin{aligned}\hat{\boldsymbol{P}}_{k+1|k}^{-1} &= \hat{\boldsymbol{P}}_{k|k-1}^{-1} + (1-\gamma_f^{-2})\boldsymbol{H}_k^T\boldsymbol{H}_k \\ &= \epsilon_0^{-1}\boldsymbol{I} + (1-\gamma_f^{-2})\sum_{i=0}^{k}\boldsymbol{H}_i^T\boldsymbol{H}_i \\ &= \epsilon_0^{-1}\boldsymbol{I} + (1-\gamma_f^{-2})\boldsymbol{U}_k\boldsymbol{\Lambda}_k\boldsymbol{U}_k^T \\ &= \boldsymbol{U}_k(\epsilon_0^{-1}\boldsymbol{I} + (1-\gamma_f^{-2})\boldsymbol{\Lambda}_k)\boldsymbol{U}_k^T \quad (7)\end{aligned}$$

ただし、

$$\sum_{i=0}^{k}\boldsymbol{H}_i^T\boldsymbol{H}_i = \boldsymbol{U}_k\boldsymbol{\Lambda}_k\boldsymbol{U}_k^T,\ \ \boldsymbol{\Lambda}_k \geq 0 \quad (8)$$

ここで、$\boldsymbol{\Lambda}_k$ は固有値からなる非負対角行列 $\mathrm{diag}\{\lambda_0,\cdots,\lambda_k,0,\cdots,0\}$、$\boldsymbol{U}_k$ は固有ベクトルからなる直交行列である。よって、式 (7) より $\hat{\boldsymbol{P}}_{k+1|k}$ は次のように表すことができる。

$$\hat{\boldsymbol{P}}_{k+1|k} = \boldsymbol{U}_k\mathrm{diag}\{\frac{\epsilon_0}{1+\epsilon_0(1-\gamma_f^{-2})\lambda_0},\cdots, \frac{\epsilon_0}{1+\epsilon_0(1-\gamma_f^{-2})\lambda_k},\epsilon_0,\cdots,\epsilon_0\}\boldsymbol{U}_k^T \quad (9)$$

これより、式 (3) による重みベクトルの更新方向は、固有値 λ_i が十分に大きければ、$\boldsymbol{H}_0,\cdots,\boldsymbol{H}_k$ が張る部分重み空間とほぼ直交する。その結果、H_∞ 学習は鞍点を回避して効果的に解を探索できる。

6 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 [2]。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 2 のような L 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 $n = 3$ であるとき層数は $L = 9$ となる。図 2 中の四角のニューロンは、応答関数が恒等関数であり、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 2 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。

$$\boldsymbol{z}^l[p] = \boldsymbol{f}(\boldsymbol{s}^l),\ \ \boldsymbol{s}^l = \boldsymbol{W}^{l-1}\boldsymbol{z}^{l-1}[p] + \boldsymbol{b}^l \quad (l = 2 \text{ または } l = 3,5,\ldots,L) \quad (10)$$

$$\boldsymbol{z}^l[p] = \boldsymbol{s}^l,\ \ \boldsymbol{s}^l = \boldsymbol{W}^{l-1}\boldsymbol{z}^{l-1}[p] + \boldsymbol{z}^{l-2}[p] \quad (l = 4,6,\ldots,L-1) \quad (11)$$

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、

$$\boldsymbol{f}(\boldsymbol{s}^l) = \left[f(s_1^l),\ldots,f(s_{N_l}^l)\right]^T \quad (12)$$

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、

$$\boldsymbol{W}^l = \begin{bmatrix} w^l_{1,1} & \cdots & w^l_{1,N_l} \\ \vdots & \ddots & \vdots \\ w^l_{N_{l+1},1} & \cdots & w^l_{N_{l+1},N_l} \end{bmatrix} \in \mathcal{R}^{N_{l+1}\times N_l} \quad (13)$$

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

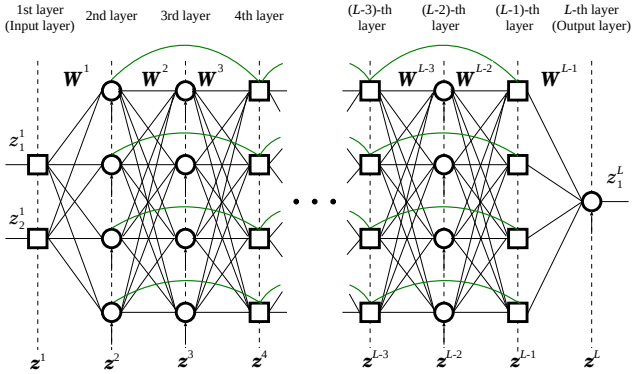


図 2　L 層 ResNet；L は層数であり（$L = 2n+3,\ n = 1, 2, \ldots$）、隠れ層のニューロン数 $N_l,\ l = 2, \ldots, L-1$ は等しい。$\boldsymbol{z}^l$ は l 層の出力、$\boldsymbol{W}^l$ は $l+1$ 層、l 層間の重み行列である。四角のニューロンは、応答関数が恒等写像であり、しきい値を持たないことを表す。一方、丸のニューロンは応答関数がシグモイド関数であり、しきい値を持つことを表す。l を 0 ではない偶数とするとき、l 層と $l+2$ 層はショートカット接続されている。

7　スキップコネクションを考慮した H_∞ 学習

L 層 ResNet における H_∞ 学習は、文献 [7] 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

$$\boldsymbol{w}_{k+1} = \boldsymbol{w}_k,\ \boldsymbol{m}_k = \boldsymbol{H}_k \boldsymbol{w}_k + \boldsymbol{v}_k \quad (14)$$

における観測行列

$$\boldsymbol{H}_k = \left.\frac{\partial \boldsymbol{h}_k(\boldsymbol{w})}{\partial \boldsymbol{w}}\right|_{\boldsymbol{w}=\hat{\boldsymbol{w}}_{k-1}} \in \mathcal{R}^{N_L \times N_w} \quad (15)$$

の計算方法だけである。ここで、

$$\begin{aligned} \boldsymbol{w} = [&w^1_{1,0}, w^1_{1,1}, \ldots, w^1_{N_2,N_1}, \cdots \\ &\ldots, w^{L-1}_{1,0}, w^{L-1}_{1,1}, \ldots, w^{L-1}_{N_L,N_{L-1}}]^T \in \mathcal{R}^{N_w} \end{aligned} \quad (16)$$

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 $\frac{\partial \boldsymbol{h}_k}{\partial w^1_{j,i}}$ は次式により計算される。

$$\begin{aligned} \frac{\partial \boldsymbol{h}_k}{\partial w^1_{j,i}} &= \frac{\partial \boldsymbol{z}^L}{\partial \boldsymbol{s}^L} \cdot \frac{\partial \boldsymbol{s}^L}{\partial \boldsymbol{z}^{L-1}} \frac{\partial \boldsymbol{z}^{L-1}}{\partial \boldsymbol{s}^{L-1}} \cdots \\ &\quad \cdots \frac{\partial \boldsymbol{s}^4}{\partial \boldsymbol{z}^3} \frac{\partial \boldsymbol{z}^3}{\partial \boldsymbol{s}^3} \cdot \frac{\partial \boldsymbol{s}^3}{\partial \boldsymbol{z}^2} \frac{\partial \boldsymbol{z}^2}{\partial \boldsymbol{s}^2} \cdot \frac{\partial \boldsymbol{s}^2}{\partial w^1_{j,i}} && (17) \\ &= \boldsymbol{\Phi}^L \cdot \frac{\partial \boldsymbol{s}^L}{\partial \boldsymbol{z}^{L-1}} \frac{\partial \boldsymbol{z}^{L-1}}{\partial \boldsymbol{s}^{L-1}} \cdots \\ &\quad \cdots \frac{\partial \boldsymbol{s}^4}{\partial \boldsymbol{z}^3} \frac{\partial \boldsymbol{z}^3}{\partial \boldsymbol{s}^3} \cdot \frac{\partial \boldsymbol{s}^3}{\partial \boldsymbol{z}^2} \frac{\partial \boldsymbol{z}^2}{\partial \boldsymbol{s}^2} \cdot \frac{\partial \boldsymbol{s}^2}{\partial w^1_{j,i}} && (18) \\ &\quad \vdots && (19) \\ &= \boldsymbol{\Phi}^2 \cdot \frac{\partial \boldsymbol{s}^2}{\partial w^1_{j,i}} && (20) \end{aligned}$$

$$\frac{\partial \boldsymbol{h}_k}{\partial w^2_{j,i}} = \boldsymbol{\Phi}^3 \cdot \frac{\partial \boldsymbol{s}^3}{\partial w^2_{j,i}},\ \cdots,\ \frac{\partial \boldsymbol{h}_k}{\partial w^l_{j,i}} = \boldsymbol{\Phi}^{l+1} \cdot \frac{\partial \boldsymbol{s}^{l+1}}{\partial w^l_{j,i}} \quad (21)$$

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 ϕ^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逆方向の再帰式で計算できる。

$$\left(\boldsymbol{\phi}^l\right)^T = \left(\boldsymbol{\phi}^{l+1}\boldsymbol{W}^l\right)^T \odot \begin{bmatrix} \frac{\partial z^l_1}{\partial s^l_1} \\ \vdots \\ \frac{\partial z^l_{N_l}}{\partial s^l_{N_l}} \end{bmatrix} \quad (22)$$

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

$$\left(\boldsymbol{\phi}^l\right)^T = \left(\boldsymbol{\phi}^{l+1}\boldsymbol{W}^l + \boldsymbol{\phi}^{l+2}\right)^T \odot \begin{bmatrix} \frac{\partial z^l_1}{\partial s^l_1} \\ \vdots \\ \frac{\partial z^l_{N_l}}{\partial s^l_{N_l}} \end{bmatrix} \quad (23)$$

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

ただし、

$$\phi^L = \frac{\partial z^L_1}{\partial s^L_1} \in \mathcal{R} \quad (24)$$

8　シミュレーション

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（epoch 数）の統計量を、それぞれ図 3 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、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（スキップコネクションがない場合は文献 [7] を参照されたい）。最急降下方向に更新した場合（$\hat{\boldsymbol{P}}_{k+1|k} = \hat{\boldsymbol{P}}_{0|-1}$）、層数を増やすにつれて平均学習回数が増加し、$L = 801$ で 4833.9 となった。

一方、図 4 と表 2 には Adam [4] を用いたときの結果を示した。学習回数の平均は H_∞ 学習の 100 倍近くになっている。層数を増やすと H_∞ 学習と同様に学習回数が減少する傾向があったが、深い層では学習曲線の後半で激しく振動していることがわかる。これは Adam がモーメント法の一種であることから予測できる（深くて狭い谷で激しく振動）。Adam が性能を発揮するためにはモーメント β_1, β_2 の調整に注意が必要である。

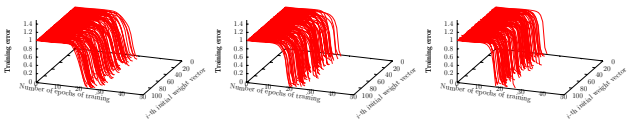
9　まとめ

論理関数 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 epoch 数で学習が終了した。一方、Adam を用いた学習では１更新当たりの計算量は少ないが、学習回数（epoch 数）が初期重みに大きく依存し、その回数は H_∞ 学習の 100 倍近かった。これより、H_∞ 学習と Adam は鞍点の多い誤差曲面における解の探索戦略が本質的に異なることがわかる。

今後は、多出力系のニューラルネットワークに対して l-EHF [6] をベースに実用的な H_∞ 学習アルゴリズムの開発を目指したい。

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

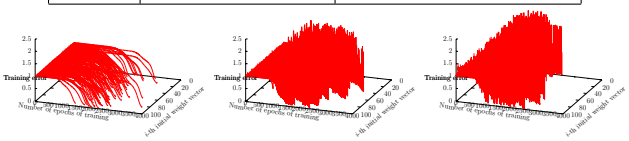


(a)$L = 5$　(b)$L = 201$　(c)$L = 1001$

図 3　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}。

表 2　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)。学習法は Adam($\alpha = 0.01, \beta_1 = 0.79, \beta_2 = 0.78, \epsilon = 10^{-8}$)、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	2160.5	442.5
101	1984.9	447.9
201	2078.1	467.4
401	1950.8	423.6
801	1783.7	346.4
1001	1724.6	366.2



(a)$L = 5$　(b)$L = 201$　(c)$L = 1001$

図 4　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)。学習法は Adam($\alpha = 0.01, \beta_1 = 0.79, \beta_2 = 0.78, \epsilon = 10^{-8}$)、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] 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.

[3] Y.N. Dauphin et al., “Identifying and attacking the saddle point problem in high-dimensional non-convex optimization,” https://arxiv.org/abs/1406.2572, 2014.

[4] S. Bock and M. Weiss, “A Proof of Local Convergence for the Adam Optimizer,” Proceedings of IJCNN, 2019.

[5] 西山 清, 最適フィルタリング, 培風館, 2001.

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

[7] 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.