F-037

行列分解を用いた深層残差 H_∞ 学習の計算量の削減

Reduction of a Computational Complexity of the Deep Residual H_∞-Learning Using Matrix Factorization

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、ある深さから学習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接続（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。我々の先行研究では、モデル集合の包含関係と誤差曲面の高次鞍点化の観点 [3] から深層残差ネットワークの H_∞ 学習が優れた学習特性をもつことを明らかにした。しかし、H_∞ 学習を大規模なネットワークに適用するためには計算量の問題を解決する必要があった。

本研究では、H_∞ 学習の計算量を削減するため、まず N_L 出力のネットワークの出力層から入力層に向けて各ニューロンに対する結合重みとしきい値（部分重みベクトル）を順次更新する。次に、各ニューロンの部分観測行列が $N_L \times 1$ 行列と $1 \times N_n$ 行列の積に分解できることに着目し、部分重みベクトルの更新に必要な $N_L \times N_L$ 行列の逆行列の計算を回避する。最後に、これらの削減法を導入した二つの深層残差 H_∞ 学習アルゴリズムを提案する。

2 H_∞ 学習

H_∞ 学習問題とは、ある値 γ_f が与えられたとき、

$$\sup_{w_0,\{\boldsymbol{v}_p\}} \frac{\sum_{p=0}^{k} \|\boldsymbol{e}_{f,p}\|^2_{(\sigma_v^2 \boldsymbol{I})^{-1}}}{\|\boldsymbol{w}-\check{\boldsymbol{w}}_0\|^2_{(\epsilon_0 \boldsymbol{I})^{-1}} + \sum_{p=0}^{k} \|\boldsymbol{v}_p\|^2_{(\sigma_v^2 \boldsymbol{I})^{-1}}} < \gamma_f^2 \quad (1)$$

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、$\boldsymbol{v}_p$ は線形化誤差を含む観測雑音である。 ■

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) [4] 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 [5]。次に、出力層のニューロン数が一つの場合を示す。

$$\hat{\boldsymbol{w}}_k = \hat{\boldsymbol{w}}_{k-1} + \boldsymbol{K}_{s,k}(y_k - h_k(\hat{\boldsymbol{w}}_{k-1})) \quad (2)$$

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

ただし、

$$\boldsymbol{H}_k = \left.\frac{\partial h_k(\boldsymbol{w})}{\partial \boldsymbol{w}}\right|_{\boldsymbol{w}=\hat{\boldsymbol{w}}_{k-1}}, \hat{\boldsymbol{P}}_{0|-1} = \frac{\epsilon_0}{\sigma_v^2}\boldsymbol{I},\ \epsilon_0 > 0 \quad (4)$$

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

$$\boldsymbol{R} = \begin{bmatrix} 1 & 0 \\ 0 & -\gamma_f^2 \end{bmatrix}, \quad \gamma_f > 1 \quad (5)$$

3 残差ネットワーク (ResNet)

ResNet は、ある 2 つの層間の出力をスキップ接続 (skip 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]。スキップ接続とは、ある NN における n 層の出力 $\boldsymbol{x} \in \mathcal{R}^M$ と $n+m$ 層の出力 $\boldsymbol{y}(\boldsymbol{x}) \in \mathcal{R}^M$ を加算することである。その和を $\boldsymbol{z}(\boldsymbol{x}) = \boldsymbol{y}(\boldsymbol{x}) + \boldsymbol{x}$ とする。なお、$\mathcal{R}$ は実数全体の集合、n, m, M は自然数である。このスキップ接続により、$\boldsymbol{z}(\boldsymbol{x})$ を学習する問題は残差 $\boldsymbol{y}(\boldsymbol{x}) = \boldsymbol{z}(\boldsymbol{x}) - \boldsymbol{x}$ を学習する問題に帰着できる。このことから、このスキップ接続を含む 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 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（多出力の場合も扱う）。この ResNet の層数 L はスキップ接続の数 l で決定される ($L = 2l + 3$, $l = 0, 1, \ldots$)。図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)。図中の弧線はスキップ接続を表す。

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

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

$$\boldsymbol{z}_p^n \;=\; \boldsymbol{s}_p^{n-1},\;\; \boldsymbol{s}_p^{n-1} = \boldsymbol{W}^{n-1}\boldsymbol{z}_p^{n-1} + \boldsymbol{z}_p^{n-2} \quad (n = 4, 6, \ldots, L-1) \tag{7}$$

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

$$\boldsymbol{f}(\boldsymbol{s}_p^{n-1}) = \left[f(s_{1,p}^{n-1}), \ldots, f(s_{N_n,p}^{n-1})\right]^T \tag{8}$$

は n 層の膜電位 $\boldsymbol{s}_p^{n-1}$ の各成分に対するニューロンの応答から成るベクトル値関数である。また、

$$\boldsymbol{W}^n = \begin{bmatrix} w_{1,1}^n & \cdots & w_{1,N_n}^n \\ \vdots & \ddots & \vdots \\ w_{N_{n+1},1}^n & \cdots & w_{N_{n+1},N_n}^n \end{bmatrix} \in \mathcal{R}^{N_{n+1} \times N_n} \tag{9}$$

は n 層から $n+1$ 層への重み行列、$\boldsymbol{b}^n = [w_{1,0}^n, \ldots, w_{N_{n+1},0}^n]^T$ は $n+1$ 層のしきい値ベクトルである。

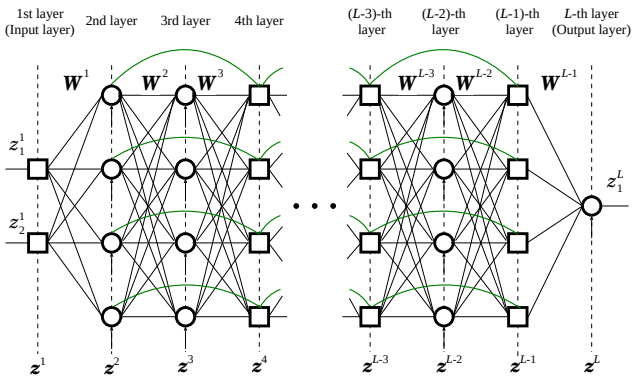

図 1　L 層 ResNet；L は層数であり（$L = 2l+3,\ l = 0, 1, \ldots$）、隠れ層のニューロン数 $N_n,\ n = 2, \ldots, L-1$ は等しい。$\boldsymbol{z}^n$ は n 層の出力、$\boldsymbol{W}^n$ は n 層から $n+1$ 層への重み行列である。四角のニューロンは、応答関数が恒等写像であり、しきい値を持たないことを表す。一方、丸のニューロンは応答関数がシグモイド関数であり、しきい値を持つことを表す。n を 0 ではない偶数とするとき、n 層と $n+2$ 層はスキップ接続されている。

4　スキップ接続を考慮した H_∞ 学習

L 層 ResNet における H_∞ 学習と文献 [5] で述べた 3 層 NN の H_∞ 学習の違いは、NN の線形状態空間モデル

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

における観測行列

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

の計算方法だけである。ここで、

$$\boldsymbol{m}_k \;=\; \boldsymbol{y}_k - \boldsymbol{h}_k(\hat{\boldsymbol{w}}_{k-1}) + \boldsymbol{H}_k\hat{\boldsymbol{w}}_{k-1} \tag{12}$$

$$\boldsymbol{w} \;=\; [w_{1,0}^1, w_{1,1}^1, \ldots, w_{1,N_1}^1, \ldots \ldots, w_{N_L,0}^{L-1}, w_{N_L,1}^{L-1}, \ldots, w_{N_L,N_{L-1}}^{L-1}]^T \tag{13}$$

であり、$\boldsymbol{w} = [vec([\boldsymbol{b}^1, \boldsymbol{W}^1]^T)^T, \cdots,]^T$ はすべてのしきい値と結合重みからなる重みベクトル、$\boldsymbol{h}_k(\boldsymbol{w}) = [h_{1,k}(\boldsymbol{w}), \ldots, h_{N_L,k}(\boldsymbol{w})]^T$ は更新ステップ k の L 層 ResNet の出力 $\boldsymbol{z}_k^L$ を表す $\boldsymbol{w}$ の関数（$\boldsymbol{y}_k = \boldsymbol{z}_k^L$）、$\hat{\boldsymbol{w}}_k$ は更新ステップ k 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 $\frac{\partial \boldsymbol{h}_k}{\partial w_{j,i}^n}$ は次式により計算される [6]。

$$\frac{\partial \boldsymbol{h}_k}{\partial w_{j,i}^1} \;=\; \boldsymbol{\Phi}_k^2 \cdot \frac{\partial \boldsymbol{s}_k^1}{\partial w_{j,i}^1} \tag{14}$$

$$\frac{\partial \boldsymbol{h}_k}{\partial w_{j,i}^2} \;=\; \boldsymbol{\Phi}_k^3 \cdot \frac{\partial \boldsymbol{s}_k^2}{\partial w_{j,i}^2},\ \cdots,\ \frac{\partial \boldsymbol{h}_k}{\partial w_{j,i}^n} = \boldsymbol{\Phi}_k^{n+1} \cdot \frac{\partial \boldsymbol{s}_k^n}{\partial w_{j,i}^n} \tag{15}$$

ここで、行列 $\boldsymbol{\Phi}_k^n$ は次の逆方向の再帰式で計算できる。

$$\boldsymbol{\Phi}_k^n \;=\; \boldsymbol{\Phi}_k^{n+1}\frac{\partial \boldsymbol{s}_k^n}{\partial \boldsymbol{z}_k^n}\frac{\partial \boldsymbol{z}_k^n}{\partial \boldsymbol{s}_k^{n-1}} \;\in \mathcal{R}^{N_L \times N_n} \tag{16}$$

$$\;=\; \boldsymbol{\Phi}_k^{n+1}\boldsymbol{W}^n \begin{bmatrix} \frac{\partial z_{1,k}^n}{\partial s_{1,k}^{n-1}} & & \boldsymbol{O} \\ & \ddots & \\ \boldsymbol{O} & & \frac{\partial z_{N_n,k}^n}{\partial s_{N_n,k}^{n-1}} \end{bmatrix} \tag{17}$$

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

$$\boldsymbol{\Phi}_k^n \;=\; (\boldsymbol{\Phi}_k^{n+1}\frac{\partial \boldsymbol{s}_k^n}{\partial \boldsymbol{z}_k^n} + \boldsymbol{\Phi}_k^{n+2})\frac{\partial \boldsymbol{z}_k^n}{\partial \boldsymbol{s}_k^{n-1}} \tag{18}$$

$$\;=\; \left(\boldsymbol{\Phi}_k^{n+1}\boldsymbol{W}^n + \boldsymbol{\Phi}_k^{n+2}\right) \begin{bmatrix} \frac{\partial z_{1,k}^n}{\partial s_{1,k}^{n-1}} & & \boldsymbol{O} \\ & \ddots & \\ \boldsymbol{O} & & \frac{\partial z_{N_n,k}^n}{\partial s_{N_n,k}^{n-1}} \end{bmatrix} \tag{19}$$

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

ただし、

$$\boldsymbol{\Phi}_k^L \;=\; \frac{\partial \boldsymbol{z}_k^L}{\partial \boldsymbol{s}_k^{L-1}} \;\in \mathcal{R}^{N_L \times N_L} \tag{20}$$

$$\frac{\partial \boldsymbol{z}_k^n}{\partial \boldsymbol{s}_k^{n-1}} \;=\; \begin{bmatrix} \frac{\partial z_{1,k}^n}{\partial s_{1,k}^{n-1}} & & O \\ & \ddots & \\ O & & \frac{\partial z_{N_n,k}^n}{\partial s_{N_n,k}^{n-1}} \end{bmatrix} \tag{21}$$

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

$$\frac{\partial \boldsymbol{s}_k^n}{\partial \boldsymbol{z}_k^n} = \boldsymbol{W}^n \in \mathcal{R}^{N_{n+1} \times N_n} \tag{22}$$

である。

5 深層残差 H_∞ 学習の計算量の削減

先行研究 [6] では、1 出力の場合に限定した H_∞ 学習に関して述べた。本章では、多出力 NN に対する深層残差 H_∞ 学習の計算量の削減方法を導出する [5],[7]。

まず、第 $n+1$ 層の i ニューロンに着目した場合の H_∞ 学習のリカッチ方程式を考える。

$$\hat{\boldsymbol{\Sigma}}^n_{i,k+1|k} = \hat{\boldsymbol{\Sigma}}^n_{i,k|k-1} - \hat{\boldsymbol{\Sigma}}^n_{i,k|k-1} \begin{bmatrix} \boldsymbol{H}^n_{i,k} \\ \boldsymbol{H}^n_{i,k} \end{bmatrix}^T \boldsymbol{R}^{n,i\,-1}_{\boldsymbol{e},k} \begin{bmatrix} \boldsymbol{H}^n_{i,k} \\ \boldsymbol{H}^n_{i,k} \end{bmatrix} \hat{\boldsymbol{\Sigma}}^n_{i,k|k-1} \tag{23}$$

ただし、

$$\boldsymbol{R}^{n,i}_{\boldsymbol{e},k} = \boldsymbol{R}^n_k + \begin{bmatrix} \boldsymbol{H}^n_{i,k} \\ \boldsymbol{H}^n_{i,k} \end{bmatrix} \hat{\boldsymbol{\Sigma}}^n_{i,k|k-1} \begin{bmatrix} \boldsymbol{H}^n_{i,k} \\ \boldsymbol{H}^n_{i,k} \end{bmatrix}^T \tag{24}$$

$$\boldsymbol{R}^n_k = \begin{bmatrix} \hat{\sigma}^2_{v_k}\boldsymbol{I} & 0 \\ 0 & -\gamma_f^2\boldsymbol{I} \end{bmatrix} \tag{25}$$

$$\boldsymbol{H}_k = [\boldsymbol{H}^1_{1,k}, \cdots, \boldsymbol{H}^{n+1}_{N_n,k}, \cdots, \boldsymbol{H}^{L-1}_{N_L,k}] \tag{26}$$

ここで、$\boldsymbol{R}^{n,i\,-1}_{\boldsymbol{e},k}$ を 2×2 ブロック行列

$$\boldsymbol{R}^{n,i\,-1}_{\boldsymbol{e},k} = \begin{bmatrix} \boldsymbol{L}^n_{i,k} & \boldsymbol{M}^n_{i,k} \\ \boldsymbol{N}^n_{i,k} & \boldsymbol{O}^n_{i,k} \end{bmatrix} \tag{27}$$

で表し、

$$\boldsymbol{S}^n_{i,k} = \boldsymbol{L}^n_{i,k} + \boldsymbol{M}^n_{i,k} + \boldsymbol{N}^n_{i,k} + \boldsymbol{O}^n_{i,k} \tag{28}$$

を定義すれば、式 (23) を次のように書き換えることができる。

$$\hat{\boldsymbol{\Sigma}}^n_{i,k+1|k} = \hat{\boldsymbol{\Sigma}}^n_{i,k|k-1} - \hat{\boldsymbol{\Sigma}}^n_{i,k|k-1}\boldsymbol{H}^{n\,T}_{i,k}\boldsymbol{S}^n_{i,k}\boldsymbol{H}^n_{i,k}\hat{\boldsymbol{\Sigma}}^n_{i,k|k-1} \tag{29}$$

一方、$\boldsymbol{R}^{n,i}_{\boldsymbol{e},k}$ の逆行列は次のように分解・整理できる。

$$\begin{aligned} \boldsymbol{R}^{n,i\,-1}_{\boldsymbol{e},k} &= \begin{bmatrix} \boldsymbol{I} & \boldsymbol{A}^n_{i,k} \\ 0 & \boldsymbol{I} \end{bmatrix} \begin{bmatrix} \boldsymbol{B}^n_{i,k} & 0 \\ 0 & \boldsymbol{C}^n_{i,k} \end{bmatrix} \begin{bmatrix} \boldsymbol{I} & 0 \\ \boldsymbol{A}^n_{i,k} & \boldsymbol{I} \end{bmatrix} \\ &= \begin{bmatrix} \boldsymbol{B}^n_{i,k} + \boldsymbol{A}^n_{i,k}\boldsymbol{C}^n_{i,k}\boldsymbol{A}^n_{i,k} & \boldsymbol{A}^n_{i,k}\boldsymbol{C}^n_{i,k} \\ \boldsymbol{C}^n_{i,k}\boldsymbol{A}^n_{i,k} & \boldsymbol{C}^n_{i,k} \end{bmatrix} \end{aligned} \tag{30}$$

また、$n+1$ 層のニューロン i に対する部分観測行列は $N_L \times 1$ 行列 $\boldsymbol{u}^n_{i,k}$ を用いて次のように分解できる。

$$\boldsymbol{H}^n_{i,k} = \frac{\partial \boldsymbol{z}^L_k}{\partial s^n_{i,k}} \frac{\partial s^n_{i,k}}{\partial \boldsymbol{w}^n_{i,k}} = \boldsymbol{u}^n_{i,k}\hat{\boldsymbol{z}}^{nT}_k \tag{31}$$

さらに、ブロック行列 $\boldsymbol{R}^{n,i}_{\boldsymbol{e},k}$ の逆行列の公式と行列分解 $\boldsymbol{H}^n_{i,k} = \boldsymbol{u}^n_{i,k}\hat{\boldsymbol{z}}^{n\,T}_k$ を用いれば、行列 $\boldsymbol{A}^n_{i,k}$、$\boldsymbol{B}^n_{i,k}$ および $\boldsymbol{C}^n_{i,k}$ は次のように整理できる。

$$\boldsymbol{A}^n_{i,k} = -\frac{\alpha^n_{i,k}}{\hat{\sigma}^2_{v_k} + \alpha^n_{i,k}\beta^n_{i,k}} \boldsymbol{u}^n_{i,k}\boldsymbol{u}^{n\,T}_{i,k} \tag{32}$$

$$\begin{aligned} \boldsymbol{B}^n_{i,k} &= (\hat{\sigma}^2_{v_k}\boldsymbol{I} + \boldsymbol{H}^n_{i,k}\hat{\boldsymbol{\Sigma}}^n_{i,k|k-1}\boldsymbol{H}^{n\,T}_{i,k})^{-1} \qquad (33) \\ &= (\hat{\sigma}^2_{v_k}\boldsymbol{I} + \boldsymbol{u}^n_{i,k}\alpha^n_{i,k}\boldsymbol{u}^{n\,T}_{i,k})^{-1} \\ &= \frac{1}{\hat{\sigma}^2_{v_k}} \left[\boldsymbol{I} - \frac{\alpha^n_{i,k}}{\hat{\sigma}^2_{v_k} + \alpha^n_{i,k}\beta^n_{i,k}} \boldsymbol{u}^n_{i,k}\boldsymbol{u}^{n\,T}_{i,k} \right] \qquad (34) \\ &= \frac{1}{\hat{\sigma}^2_{v_k}} \left[\boldsymbol{I} + \boldsymbol{A}^n_{i,k} \right] \qquad (35) \end{aligned}$$

$$\begin{aligned} \boldsymbol{C}^n_{i,k} = \frac{1}{\gamma_f^2} \Bigg[& \frac{\hat{\sigma}^2_{v_k}\alpha^n_{i,k}}{\hat{\sigma}^2_{v_k}\alpha^n_{i,k}\beta^n_{i,k} - \gamma_f^2(\hat{\sigma}^2_{v_k} + \alpha^n_{i,k}\beta^n_{i,k})} \\ & \boldsymbol{u}^n_{i,k}\boldsymbol{u}^{n\,T}_{i,k} - \boldsymbol{I} \Bigg] \end{aligned} \tag{36}$$

ただし、

$$\alpha^n_{i,k} = \hat{\boldsymbol{z}}^{n\,T}_k \hat{\boldsymbol{\Sigma}}^n_{i,k|k-1}\hat{\boldsymbol{z}}^n_k, \quad \beta^n_{i,k} = \boldsymbol{u}^{n\,T}_{i,k}\boldsymbol{u}^n_{i,k} \tag{37}$$

ここで、式 (27) と式 (30) を比較すれば式 (28) は次のように整理できる（$\hat{\sigma}^2_{v_k} + \alpha^n_{i,k}\beta^n_{i,k}$ による約分に注意）。

$$\begin{aligned} \boldsymbol{S}^n_{i,k} &= (\boldsymbol{B}^n_{i,k} + \boldsymbol{A}^n_{i,k}\boldsymbol{C}^n_{i,k}\boldsymbol{A}^n_{i,k}) \\ &\quad + \boldsymbol{A}^n_{i,k}\boldsymbol{C}^n_{i,k} + \boldsymbol{C}^n_{i,k}\boldsymbol{A}^n_{i,k} + \boldsymbol{C}^n_{i,k} \\ &= (\boldsymbol{A}^n_{i,k} + \boldsymbol{I}) \left\{ \frac{1}{\hat{\sigma}^2_{v_k}}\boldsymbol{I} + \boldsymbol{C}^n_{i,k}(\boldsymbol{A}^n_{i,k} + \boldsymbol{I}) \right\} \\ &= \frac{(1 - \gamma_f^{-2}\hat{\sigma}^2_{v_k})}{\hat{\sigma}^2_{v_k}} \\ &\quad \times \left[\boldsymbol{I} - \frac{(1 - \gamma_f^{-2}\hat{\sigma}^2_{v_k})\alpha^n_{i,k}}{\hat{\sigma}^2_{v_k} + (1 - \gamma_f^{-2}\hat{\sigma}^2_{v_k})\alpha^n_{i,k}\beta^n_{i,k}} \boldsymbol{u}^n_{i,k}\boldsymbol{u}^{n\,T}_{i,k} \right] \end{aligned} \tag{38}$$

これより、次式を得る。

$$\boldsymbol{u}^{n\,T}_{i,k}\boldsymbol{S}^n_{i,k}\boldsymbol{u}^n_{i,k} = \frac{(1 - \gamma_f^{-2}\hat{\sigma}^2_{v_k})\beta^n_{i,k}}{\hat{\sigma}^2_{v_k} + (1 - \gamma_f^{-2}\hat{\sigma}^2_{v_k})\alpha^n_{i,k}\beta^n_{i,k}}. \tag{39}$$

これを次のように整理された式 (29) の右辺第 2 項

$$\begin{aligned} & \hat{\boldsymbol{\Sigma}}^n_{i,k|k-1}\boldsymbol{H}^{n\,T}_{i,k}\boldsymbol{S}^n_{i,k}\boldsymbol{H}^n_{i,k}\hat{\boldsymbol{\Sigma}}^n_{i,k|k-1} \\ &= \hat{\boldsymbol{\Sigma}}^n_{i,k|k-1}\hat{\boldsymbol{z}}^n_k \left(\boldsymbol{u}^{n\,T}_{i,k}\boldsymbol{S}^n_{i,k}\boldsymbol{u}^n_{i,k} \right) \hat{\boldsymbol{z}}^{n\,T}_k \hat{\boldsymbol{\Sigma}}^n_{i,k|k-1} \end{aligned}$$

に代入すれば逆行列を含まないリカッチ方程式が得られる。

$$\hat{\boldsymbol{\Sigma}}^n_{i,k+1|k} = \hat{\boldsymbol{\Sigma}}^n_{i,k|k-1} - \frac{(1 - \gamma_f^{-2}\hat{\sigma}^2_{v_k})\beta^n_{i,k}}{\hat{\sigma}^2_{v_k} + (1 - \gamma_f^{-2}\hat{\sigma}^2_{v_k})\alpha^n_{i,k}\beta^n_{i,k}} \boldsymbol{\psi}^n_{i,k}\boldsymbol{\psi}^{n\,T}_{i,k} \tag{40}$$

ただし、

$$\boldsymbol{\psi}^n_{i,k} = \hat{\boldsymbol{\Sigma}}^n_{i,k|k-1}\hat{\boldsymbol{z}}^n_k \tag{41}$$

同様に、フィルタゲインも式 (34) を用いれば次のように整理できる（$\boldsymbol{H}^n_{i,k}\hat{\boldsymbol{\Sigma}}^n_{i,k|k-1}\boldsymbol{H}^{n\,T}_{i,k} = \boldsymbol{u}^n_{i,k}\alpha^n_{i,k}\boldsymbol{u}^{n\,T}_{i,k}$）。

$$\begin{aligned} \boldsymbol{K}^n_{i,k} &= \hat{\boldsymbol{\Sigma}}^n_{i,k|k-1}\boldsymbol{H}^{n\,T}_{i,k}(\hat{\sigma}^2_{v_k}\boldsymbol{I} + \boldsymbol{H}^n_{i,k}\hat{\boldsymbol{\Sigma}}^n_{i,k|k-1}\boldsymbol{H}^{n\,T}_{i,k})^{-1} \\ &= \frac{1}{\hat{\sigma}^2_{v_k} + \alpha^n_{i,k}\beta^n_{i,k}} \boldsymbol{\psi}^n_{i,k}\boldsymbol{u}^{n\,T}_{i,k} \end{aligned} \tag{42}$$

以上より，図 4 の l-EHF アルゴリズムが得られる。このアルゴリズムでは出力の観測雑音の分散 $\hat{\sigma}^2_{v_k}$ が推定されている。一方、ハイパー H_∞ フィルタ [8] を用いて同様に求めた図 5 の l-EHHF アルゴリズムでは忘却係数 ρ が効果的に利用されている。

6 シミュレーション

本章では英文字パターン認識問題を用いて深層残差 H_∞ 学習を評価する。ここで、A〜Z までの合計 26 パターンを用いた（$N_p = 26$）。これらは 0 または 1 の値をとる 7×7 ドットで表されている。図 2 はスキップ接続がないプレーンな 49-12-26 ネットワークの目的とする入出力関係を示す。これより、あるパターンが入力された場合に、出力層内の対応する 1 つのニューロンのみが値 1 をとり、その他のニューロンは値 0 をとる。

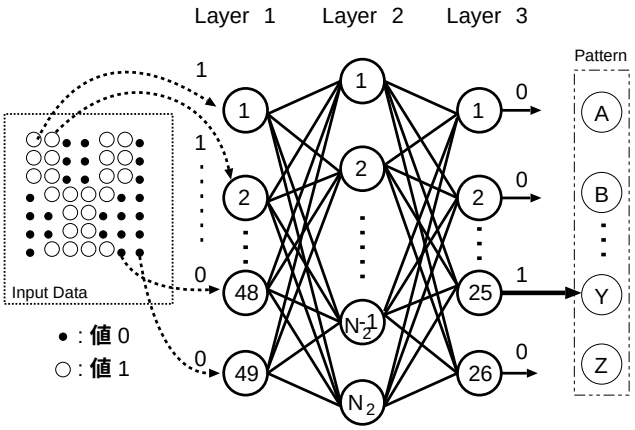

図 2　ニューラルネットによるパターン'Y' の認識

図 3 には、層数 L が $5, 21, 71$ の ResNet に対する l-EHHF の 100 試行の学習曲線、表 1 には学習回数（epoch 数=更新回数/パターン数）の平均、最大、最小を示す。一方、l-EHF（$\gamma_f = 1.7$）は 21 層で一つの初期値で更新回数が上限に達した。ただし、重みベクトル $\boldsymbol{w}$ の学習は各試行ごとに $(-0.2 \sim 0.2)$ の範囲の乱数により初期化され、10^2 試行を行った。学習は出力 2 乗誤差 J が 0.05 以下となったとき終了し、更新回数は 10^4 回を上限とした。また、シグモイド関数の傾き η_0 は 2.5 とし、l-EHF、l-EHHF において $\epsilon_0 = 0.1$ に固定した。

7 まとめ

本研究では、各ニューロンに対する部分重みベクトルに着目してリカッチ方程式の共分散行列をブロック対角化した上で、多出力系に起因する逆行列の計算を部分観測行列の行列分解を用いて回避することにより深層残差 H_∞ 学習の計算量を大幅に削減することに成功した。

今後は、大規模な畳み込みネットワーク（CNN）等の H_∞ 学習に挑戦したい。

表 1　L 層 ResNet における学習終了時の学習回数 (epoch 数) (区間 $[-0.20, 0.20]$ の一様分布により重みを初期化した 100 試行)；学習法は l-EHHF 法 ($\gamma_f = 32.0, \rho = 1 - \gamma_f^{-2}$)、入力層のニューロン数は $N_1 = 49$、隠れ層のニューロン数は $N_n = 5$、出力層のニューロン数は $N_L = 26$、シグモイド関数の傾きは $\eta_0 = 2.5$、打ち切り誤差は 5.0×10^{-2}。

層数 L	学習回数の平均	学習回数の最大/最小
5	15.0	16.3/14.0
11	11.7	13.2/10.8
21	9.89	11.2/9.04
51	8.48	10.8/6.81
71	8.46	10.2/6.54

(a)$L = 5$　(b)$L = 21$　(c)$L = 71$

図 3　L 層 ResNet における 100 試行の学習曲線 (重みは区間 $[-0.20, 0.20]$ の一様分布により初期化)；学習法は l-EHHF 法 ($\gamma_f = 32.0, \rho = 1 - \gamma_f^{-2}$)、入力層のニューロン数は $N_1 = 49$、隠れ層のニューロン数は $N_n = 5$、出力層のニューロン数は $N_L = 26$、シグモイド関数の傾きは $\eta_0 = 2.5$、打ち切り誤差は 5.0×10^{-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] 西山 清, "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," 第 20 回情報科学技術フォーラム, F-017, pp.1-4, 2021.

[4] 西山 清, 最適フィルタリング, 培風館, 2001.

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

[6] 西山 清, 菅原 康滉, "深層残差ネットワークの H_∞ 学習," 第 19 回情報科学技術フォーラム, G-007, pp.1-3, 2020.

[7] 西山 清, 鈴木清彦, "H_∞ 学習 ー 局所的最適化アプローチ," 電子情報通信学会 技術研究報告, NC2001-224, pp.223-230, 2002.

[8] K. Nishiyama, "A Unified View of Adaptive Algorithms for Finite Impulse Response Filters using the H_∞ Framework," Signal Processing (Elsevier), 97, pp.55-64, April 2014.

$for(k=1,\cdots,N_p,1+N_p,\cdots,2N_p,1+2N_p,\cdots)$
{
$\hat{\boldsymbol{z}}_k^n,\ \ n=1,2,\cdots,L$ (forward propagation with $\hat{\boldsymbol{w}}_{k-1}$)
$\hat{\sigma}_{v_k}^2=\hat{\sigma}_{v_{k-1}}^2+\mu_k\left[\dfrac{(\boldsymbol{y}_k-\hat{\boldsymbol{z}}_k^L)^T(\boldsymbol{y}_k-\hat{\boldsymbol{z}}_k^L)}{N_L}-\hat{\sigma}_{v_{k-1}}^2\right]$
$\hat{\boldsymbol{h}}_{1,k}^{L-1}=\hat{\boldsymbol{z}}_k^L$

$for(n=L-1;\ n\ \geq\ 1;\ n\leftarrow n-1)$
{
$for(i=1;\ i\ \leq\ N_{n+1};\ i\leftarrow i+1)$
{

$$\boldsymbol{u}_{i,k}^n=\begin{cases}\eta_0\hat{z}_{i,k}^{n+1}(1-\hat{z}_{i,k}^{n+1})[0,\cdots,0,\overset{i}{1},0,\cdots,0]^T, & n=L-1\\ \sum_{j=1}^{N_{n+2}}\hat{w}_{j,i,k}^{n+1}\boldsymbol{u}_{j,k}^{n+1}\ , & n=L-2\\ \sum_{j=1}^{N_{n+2}}\hat{w}_{j,i,k}^{n+1}\boldsymbol{u}_{j,k}^{n+1}\ +\boldsymbol{u}_{i,k}^{n+2}\ , & n\%2\neq 0\\ \eta_0\hat{z}_{i,k}^{n+1}(1-\hat{z}_{i,k}^{n+1})\sum_{j=1}^{N_{n+2}}\hat{w}_{j,i,k}^{n+1}\boldsymbol{u}_{j,k}^{n+1}\ , & n\%2=0\\ \eta_0\hat{z}_{i,k}^{n+1}(1-\hat{z}_{i,k}^{n+1})(\sum_{j=1}^{N_{n+2}}\hat{w}_{j,i,k}^{n+1}\boldsymbol{u}_{j,k}^{n+1}\ +\boldsymbol{u}_{i,k}^{n+2})\ , & n=1\end{cases}$$

$\boldsymbol{\psi}_{i,k}^n=\hat{\boldsymbol{\Sigma}}_{i,k|k-1}^n\hat{\boldsymbol{z}}_k^n$
$\alpha_{i,k}^n=\hat{\boldsymbol{z}}_k^{nT}\boldsymbol{\psi}_{i,k}^n,\qquad \beta_{i,k}^n=\boldsymbol{u}_{i,k}^{n\ T}\boldsymbol{u}_{i,k}^n$
$\hat{\boldsymbol{w}}_{i,k}^n=\hat{\boldsymbol{w}}_{i,k-1}^n+\dfrac{\boldsymbol{u}_{i,k}^{n\ T}[\boldsymbol{y}_k-\hat{\boldsymbol{h}}_{i,k}^n]}{\hat{\sigma}_{v_k}^2+\alpha_{i,k}^n\beta_{i,k}^n}\boldsymbol{\psi}_{i,k}^n$
$\hat{\boldsymbol{\Sigma}}_{i,k+1|k}^n=\hat{\boldsymbol{\Sigma}}_{i,k|k-1}^n-\dfrac{(1-\gamma_f^{-2}\hat{\sigma}_{v_k}^2)\beta_{i,k}^n}{\hat{\sigma}_{v_k}^2+(1-\gamma_f^{-2}\hat{\sigma}_{v_k}^2)\alpha_{i,k}^n\beta_{i,k}^n}\boldsymbol{\psi}_{i,k}^n\boldsymbol{\psi}_{i,k}^{n\ T}$
$\hat{\boldsymbol{h}}_{i+1,k}^n=\hat{\boldsymbol{h}}_{i,k}^n+\boldsymbol{u}_{i,k}^n\hat{\boldsymbol{z}}_k^{nT}(\hat{\boldsymbol{w}}_{i,k}^n-\hat{\boldsymbol{w}}_{i,k-1}^n)$
}
$\hat{\boldsymbol{h}}_{1,k}^{n-1}=\hat{\boldsymbol{h}}_{N_{n+1}+1,k}^n$
}
$\hat{\boldsymbol{w}}_k=\left[\hat{\boldsymbol{w}}_{1,k}^{1\ T},\cdots,\hat{\boldsymbol{w}}_{N_2,k}^{1\ T},\cdots,\hat{\boldsymbol{w}}_{1,k}^{L-1T},\cdots,\hat{\boldsymbol{w}}_{N_L,k}^{L-1\ T}\right]^T$
}

図 4　The l-EHF learning algorithm with skip connections ;
$\hat{\boldsymbol{w}}_{i,k}^n=[\hat{\theta}_{i,k}^n,\hat{w}_{i,1,k}^n,\cdots,\hat{w}_{i,N_n,k}^n]^T,\ \hat{\boldsymbol{z}}_k^n=[1,\hat{z}_{1,k}^n,\cdots,\hat{z}_{N_n,k}^n]^T,\ \hat{z}_{i,k}^1=z_{i,k}^1,$
$\boldsymbol{\Phi}_k^{n+1}=[\boldsymbol{u}_{1,k}^{n+1},\cdots,\boldsymbol{u}_{N_{n+1},k}^{n+1}],\ \ \hat{\boldsymbol{\Sigma}}_{i,1|0}^n=\epsilon_0\boldsymbol{I},\ \ \mu_k=\begin{cases}\frac{1}{k}, & k\leq T_{max}\\ \frac{1}{T_{max}}, & \text{otherwise}\end{cases},\ T_{max}=10N_p$

$for(k=1,\cdots,N_p,1+N_p,\cdots,2N_p,1+2N_p,\cdots)$
{
$\hat{\boldsymbol{z}}_k^n,\ \ n=1,2,\cdots,L$ (forward propagation with $\hat{\boldsymbol{w}}_{k-1}$)
$\hat{\boldsymbol{h}}_{1,k}^{L-1}=\hat{\boldsymbol{z}}_k^L$

$for(n=L-1;\ n\ \geq\ 1;\ n\leftarrow n-1)$
{
$for(i=1;\ i\ \leq\ N_{n+1};\ i\leftarrow i+1)$
{

$$\boldsymbol{u}_{i,k}^n=\begin{cases}\eta_0\hat{z}_{i,k}^{n+1}(1-\hat{z}_{i,k}^{n+1})[0,\cdots,0,\overset{i}{1},0,\cdots,0]^T, & n=L-1\\ \sum_{j=1}^{N_{n+2}}\hat{w}_{j,i,k}^{n+1}\boldsymbol{u}_{j,k}^{n+1}\ , & n=L-2\\ \sum_{j=1}^{N_{n+2}}\hat{w}_{j,i,k}^{n+1}\boldsymbol{u}_{j,k}^{n+1}\ +\boldsymbol{u}_{i,k}^{n+2}\ , & n\%2\neq 0\\ \eta_0\hat{z}_{i,k}^{n+1}(1-\hat{z}_{i,k}^{n+1})\sum_{j=1}^{N_{n+2}}\hat{w}_{j,i,k}^{n+1}\boldsymbol{u}_{j,k}^{n+1}\ , & n\%2=0\\ \eta_0\hat{z}_{i,k}^{n+1}(1-\hat{z}_{i,k}^{n+1})(\sum_{j=1}^{N_{n+2}}\hat{w}_{j,i,k}^{n+1}\boldsymbol{u}_{j,k}^{n+1}\ +\boldsymbol{u}_{i,k}^{n+2})\ , & n=1\end{cases}$$

$\boldsymbol{\psi}_{i,k}^n=\hat{\boldsymbol{\Sigma}}_{i,k|k-1}^n\hat{\boldsymbol{z}}_k^n$
$\alpha_{i,k}^n=\hat{\boldsymbol{z}}_k^{nT}\boldsymbol{\psi}_{i,k}^n,\qquad \beta_{i,k}^n=\boldsymbol{u}_{i,k}^{n\ T}\boldsymbol{u}_{i,k}^n$
$\hat{\boldsymbol{w}}_{i,k}^n=\hat{\boldsymbol{w}}_{i,k-1}^n+\dfrac{\boldsymbol{u}_{i,k}^{n\ T}[\boldsymbol{y}_k-\hat{\boldsymbol{h}}_{i,k}^n]}{\rho+\alpha_{i,k}^n\beta_{i,k}^n}\boldsymbol{\psi}_{i,k}^n$
$\hat{\boldsymbol{\Sigma}}_{i,k+1|k}^n=\left(\hat{\boldsymbol{\Sigma}}_{i,k|k-1}^n-\dfrac{(1-\gamma_f^{-2})\beta_{i,k}^n}{\rho+(1-\gamma_f^{-2})\alpha_{i,k}^n\beta_{i,k}^n}\boldsymbol{\psi}_{i,k}^n\boldsymbol{\psi}_{i,k}^{n\ T}\right)/\rho$
$\hat{\boldsymbol{h}}_{i+1,k}^n=\hat{\boldsymbol{h}}_{i,k}^n+\boldsymbol{u}_{i,k}^n\hat{\boldsymbol{z}}_k^{nT}(\hat{\boldsymbol{w}}_{i,k}^n-\hat{\boldsymbol{w}}_{i,k-1}^n)$
}
$\hat{\boldsymbol{h}}_{1,k}^{n-1}=\hat{\boldsymbol{h}}_{N_{n+1}+1,k}^n$
}
$\hat{\boldsymbol{w}}_k=\left[\hat{\boldsymbol{w}}_{1,k}^{1\ T},\cdots,\hat{\boldsymbol{w}}_{N_2,k}^{1\ T},\cdots,\hat{\boldsymbol{w}}_{1,k}^{L-1T},\cdots,\hat{\boldsymbol{w}}_{N_L,k}^{L-1\ T}\right]^T$
}

図 5　The l-EHHF learning algorithm with skip connections ;
$\hat{\boldsymbol{w}}_{i,k}^n=[\hat{\theta}_{i,k}^n,\hat{w}_{i,1,k}^n,\cdots,\hat{w}_{i,N_n,k}^n]^T,\ \hat{\boldsymbol{z}}_k^n=[1,\hat{z}_{1,k}^n,\cdots,\hat{z}_{N_n,k}^n]^T,\ \hat{z}_{i,k}^1=z_{i,k}^1,$
$\boldsymbol{\Phi}_k^{n+1}=[\boldsymbol{u}_{1,k}^{n+1},\cdots,\boldsymbol{u}_{N_{n+1},k}^{n+1}],\ \ \boldsymbol{R}_k^n=\begin{bmatrix}\rho\boldsymbol{I} & 0\\ 0 & -\gamma_f^2\rho\boldsymbol{I}\end{bmatrix},\ \ \hat{\boldsymbol{\Sigma}}_{i,1|0}^n=\epsilon_0\boldsymbol{I}$