P-16

線形多層ニューラルネットワークにおける特異値分解と次元削減

Singular Value Decomposition and Gradient Descent in Linear Multi-Layer Neural Networks

○清水恭介 [1], 青柳美輝 [2]
Kyosuke Shimizu[1], Miki Aoyagi[2]

Abstract: Multilayer linear neural networks, in which each layer consists solely of linear transformations, are mathematically equivalent to a single linear transformation. Nevertheless, when trained with gradient descent, the learning dynamics of multilayer and single-layer formulations differ, leading to distinct solutions. This study examines the effects of introducing multiple linear layers on the learning process and explores the underlying relationships between these formulations.

1. 導入

多層線形ニューラルネットワークにおいて、各層が線形変換のみで構成される場合、その全体の写像は単一の線形写像に帰着可能である。例えば、入力 $x \in \mathbb{R}^n$ に対して 2 層の線形ネットワークを考えると、出力 $y \in \mathbb{R}^m$ は、W_1,W_2 を行列として

$$y = W_2 W_1 x$$

と表される。ここで $W = W_2 W_1$ と置けば、単層の線形写像 $y = Wx$ と同値となる。しかし、勾配降下法で学習させると、両者の学習ダイナミクスは異なるものになり、得られる解も異なってくる。本講演では多層の線形層を挿入することによる学習の差異について考察し、さらにその関係性を探る。

2. 勾配降下法

入力ベクトルを $x^i \in \mathbb{R}^d$、教師信号を $y^i \in \mathbb{R}^m$ とする。二層線形モデルは次式で与えられる。

$$\hat{y}_i = W_2 W_1 x_i, \tag{1}$$

ここで $W_1 \in \mathbb{R}^{h\times d}$、$W_2 \in \mathbb{R}^{m\times h}$ である。損失関数として二乗誤差

$$SSE(W_1, W_2) = \frac{1}{2}\|y_i - \hat{y}_i\|^2 \tag{2}$$

を考える。勾配降下法は、目的関数を最小化するためにパラメータを勾配の負の方向に更新する手法である。更新則は

$$\begin{aligned} \Delta W_1 &= -\lambda \frac{\partial}{\partial W_1} SSE(W_1, W_2), \\ \Delta W_2 &= -\lambda \frac{\partial}{\partial W_2} SSE(W_1, W_2). \end{aligned} \tag{3}$$

となる。ここで $\lambda > 0$ は学習率である。

誤差ベクトルを $e_i = y_i - \hat{y}_i$ とすると、勾配は

$$\frac{\partial}{\partial W_1} SSE(W_1, W_2) = -W_2^\top e_i \, x_i^\top, \tag{4}$$

$$\frac{\partial}{\partial W_2} SSE(W_1, W_2) = -e_i \, h_i^\top, \qquad h_i = W_1 x_i \tag{5}$$

となる。したがって更新則は次のように与えられる。

$$\Delta W_1 = \lambda \mathbf{W}_2^\top (\mathbf{y}_i - \hat{\mathbf{y}}_i)\mathbf{x}_i^\top, \tag{6}$$

$$\Delta W_2 = \lambda (\mathbf{y}_i - \hat{\mathbf{y}}_i)\mathbf{h}_i^\top. \tag{7}$$

以上より、この逐次的な更新則は確率的勾配降下法（SGD）の一ステップに対応し、標準的な誤差逆伝播法の規則に一致する。

3. 特異値分解の定義

3.1　データ行列の特異値分解

入力データ $x_i \in \mathbb{R}^d$ と出力データ $y_i \in \mathbb{R}^m$ を集めて作る相関行列を

$$\Sigma^{yx} = \sum_i y_i x_i^\top \in \mathbb{R}^{m\times d} \tag{8}$$

とする。この行列に対する特異値分解（Singular Value Decomposition; SVD）は次式で表される：

$$\Sigma^{yx} = USV^\top = \sum_{\alpha=1}^{\min(m,d)} s_\alpha \, \mathbf{u}^\alpha (\mathbf{v}^\alpha)^\top, \tag{9}$$

ここで、U と V は直交行列、S は対角成分に非負の特異値 s_α を並べた行列、$\mathbf{u}^\alpha$ と $\mathbf{v}^\alpha$ はそれぞれ U と V の α 番目の列である。

4. 特異値の軌跡

細かな仮定は省くが、勾配降下法を微分方程式で表し、ある条件のもと時間 t による特異値の軌跡を求めると、2 層の場合には

$$a_\alpha(t) = \frac{s_\alpha e^{2s_\alpha t/\tau}}{e^{2s_\alpha t/\tau} - 1 + s_\alpha/a_\alpha^0} \tag{10}$$

単層の場合には

$$b_\alpha(t) = s_\alpha(1 - e^{-t/\tau}) + b_\alpha^0 e^{-t/\tau} \tag{11}$$

になる。ここで $\tau \equiv \frac{1}{\lambda}$ とする。

1: 日大理工・院 (前)・数学　2: 日大理工・教員・数学

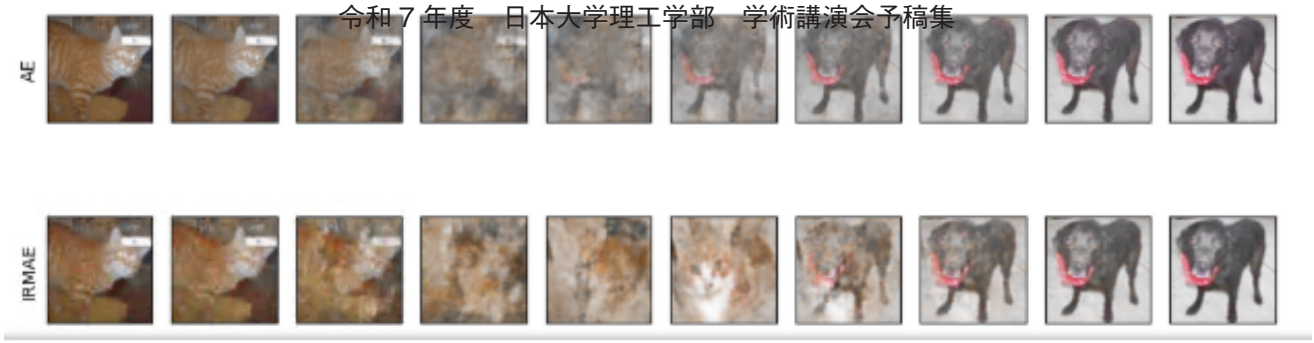

図 1: 潜在空間の可視化

特異値 $a_\alpha(t)$ は、時刻 $t = 0$ で初期値 a_α^0 から始まり、$t \to \infty$ で s_α になる。$b_\alpha(t)$ も同様である。図 2 ではある条件の下、2 層の場合と単層の特異値の時間経過による、学習の軌跡を表している。$a_\alpha(t)$ ではシグモイド軌跡を示しており、$b_\alpha(t)$ では対照的に、単純な指数関数的な挙動を示している。

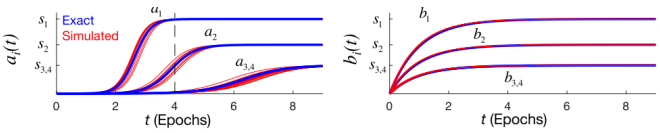

図 2: 各有効特異値の軌跡 [1]

5. 解析方法

数値解析対象は 18 枚ずつの犬と猫の画像を用意し、Encoder と Decoder の class を定義する。Encoder の後に 256 × 256 の重み行列を持つ線形層を 8 層挿入する。即ち潜在空間の次元を 256 としている。次にテストデータに対して潜在変数 z の共分散行列と共分散行列の特異値 s を取得する関数を定義する。今回のモデルでは、犬と猫のテスト画像 18 枚に対する出力 は 18×256 の行列となる。この共分散行列を計算し、結果を特異値分解する。この際にテストデータに対する誤差も計算する。線形層を含まないモデルを AE, 線形層を含んだモデルを IRMAE として, 2 つのモデルを学習させ、それぞれのテストデータに対する特異値を取得する。

図 3: 入力画像の例

6. 結果, 及び考察

6.1　潜在空間の特異値

各モデルの潜在空間の特異値 (最大値 1 に正規化) の結果を図 3 に表示する。

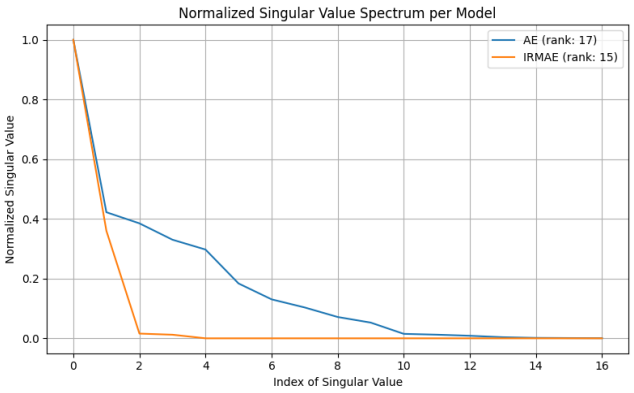

図 4: 有効特異値の比較

6.2　潜在空間の多様体の可視化及びモデル間での比較

最後に 2 つの入力に対する潜在変数を線形補間し、それを Decoder することで潜在空間の多様体を可視化してる。ここでは AE と IRMAE に対して比較する。

その結果、線形層を含めることで、潜在表現の次元をより低く抑えつつ学習が可能であることが示された。これは、線形層が情報圧縮の役割を果たし、特徴量空間を効率的に変換するためであると考えられる。

実際、画像補間の結果を見ると、線形層を含む IRMAE は通常の AE と異なり、間に他の画像が表れている。すなわち、次元削減を効率的に行っていることが確認できる。より密集した空間での識別が行われていることがわかる。以上より、線形層を導入することは、単に次元削減を行うだけでなく、潜在空間の構造を整える効果も持つことがわかる。今後の課題としては、多量の非線形性の強いデータや複雑なタスクにおいても同様の効果が得られるかを実験することが必要である。

7. 参考文献

[1] Saxe, A. et al., PNAS (2019). A mathematical theory of semantic development in deep neural networks.

[2] Python 会 (2020). 線形多層ニューラルネットワークにおける陰的正則化と IRMAE. https://oumpy.github.io/blog/2020/11/irmae.html