The 33rd Annual Conference of the Japanese Society for Artificial Intelligence, 2019

Batch Normalization つき 3 層ニューラルネットワークの学習ダイナミクスの統計力学的定式化

Statistical Mechanical Formulation of Learning Dynamics of Two-Layered Neural Networks with Batch Normalization

高木志郎 Shiro Takagi　吉田雄紀 Yuki Yoshida　岡田真人 Masato Okada

東京大学大学院新領域創成科学研究科
Graduate School of Frontier Sciences, The University of Tokyo

Batch Normalization is known as a method to shorten training time, stabilize training and improve the performance of neural networks. Despite its wide use, the impact of Batch Normalization on the learning dynamics of neural networks is yet to be clarified. Though some recent studies tried to tackle this problem, few of them derived the exact learning dynamics of neural networks with Batch Normalization. Because deriving the learning dynamics is helpful for understanding what Batch Normalization is doing during training, we derived an exact learning dynamics of two-layered neural networks with Batch Normalization by drawing on the previous work about a statistical mechanical method of neural network analysis. Specifically, for neural networks with Batch Normalization, we derived differential equations of order parameters, which represent a macroscopic behavior of neural networks.

1. はじめに

ニューラルネットワークの学習を高速化，安定化させる手法として，Ioffe と Szegedy が提案した Batch Normalization という手法がある [Ioffe 15]．これは中間層への入力を正規化することで学習の高速化を行う手法であるが，学習の高速化だけではなく，学習率や重みの初期値などのパラメータの設定を容易にしたり，正則化の効果を持っていたりと，ニューラルネットワークの学習を容易にする様々な効果が経験的に知られている．しかし，Batch Normalization がニューラルネットワークの学習に与える影響についての理解は依然として不十分である．Ioffe と Szegedy は，各層への入力がそれ以前までの重みの変更に依存するために生じる「内的共変量シフト」を Batch Normalization が低減できるため，学習を容易にすると主張した [Ioffe 15]．一方 Santurkar らは，Batch Normalization は内的共変量シフトとは関係なく，むしろ誤差曲面を滑らかにすることで予測しやすい安定な勾配が計算できるようにし，学習効率を向上させると主張した [Santurkar 18]．Bjorck らは，Batch Normalization が可能にする大きな学習率が，正則化の効果を持つことによって，学習の高速化だけでなく汎化性能の向上をもたらすことを示した [Bjorck 18]．Kohler らは，Batch Normalization は重みベクトルの最適化を長さの最適化と方向の最適化に切り分けることによって最適化を容易にすると主張した [Kohler 18]．また Arora らは，Batch Normalization が学習率の自動調節をもたらすことによって最適な収束率を実現することを，滑らかな誤差関数を用いた学習に対して示した [Arora 18]．これらはいずれも理論的な解析により Batch Normalization の効果について示唆を与えるものだが，ニューラルネットワークの学習中に重みや誤差が具体的にどのように振る舞うかについては議論ができていない．ニューラルネットワークのパラメータや誤差のダイナミクスを解析的に導出するのは一般に困難である．そのため，理想化された単純な系であっても学習のダイナミクスを求めることは重要である．そこで，私たちは 90 年代に考案された統計力学的手法を用いて Batch Normalization を適用した 3 層ソフトコミティの学習ダイナミクスを導出した．ソフトコミティとは中間層から出力層への重みを定数に固定した場合のニューラルネットワークであり，解析の簡単のため用いられることがある．統計力学的手法とは大規模ネットワークを仮定することで，系の大域的な挙動を記述するパラメータであるオーダーパラメータと訓練誤差の期待値として定義される汎化誤差のダイナミクスを解析的に導出する手法である [Schwarze 93, Seung 92, Saad 95, Biehl 95, Riegler 95]．統計力学的手法を用いて Batch Normalization のダイナミクスを解析した研究としては，Luo らの研究がある [Luo 18]．しかしこれは単層パーセプトロンに議論を限定しており，中間層がある場合のダイナミクスは扱っていない．我々は中間層がある 3 層ニューラルネットワークについてダイナミクスを導出した．

2. Batch Normalization つき 3 層ニューラルネットワークの統計力学的定式化

2.1 統計力学定式化

統計力学的定式化では一般に教師生徒型ニューラルネットワークのオンライン学習を考える [Saad 95, Biehl 95]．ここで教師生徒型学習とは，図 1 に示すように，学習器と同じ構造を持つニューラルネットワークを教師データの生成モデルと仮定する教師あり学習を指し，オンライン学習とは各更新毎に新しく生成されるサンプルサイズ 1 のデータを用いた確率的勾配降下法による学習のことを指す．この時生成モデルを教師ネットワーク，学習器を生徒ネットワークと呼ぶ．

入力素子数 N，生徒の中間素子数が K，教師の中間素子数が M，出力素子数が O の 3 層ニューラルネットワークを考える．入力 $\boldsymbol{\xi} \in \mathbb{R}^{\mathrm{N}}$ の各成分は期待値 0 分散 σ^2 の分布から i.i.d. にサンプリングされるとする．生徒ネットワークの第 1 層の重み行列を $[\mathbf{J}_1, \ldots, \mathbf{J}_K]^T \in \mathbb{R}^{K \times N}$，第二層の重み行列を $[\mathbf{w}_1, \ldots, \mathbf{w}_K]^T \in \mathbb{R}^{O \times K}$，教師ネットワークの第 1 層の重み行列を $[\mathbf{B}_1, \ldots, \mathbf{B}_M]^T \in \mathbb{R}^{M \times N}$，第二層の重み行列を $[\mathbf{v}_1, \ldots, \mathbf{v}_M]^T \in \mathbb{R}^{O \times M}$ と表記する．生徒と教師の第 1 層の重みベクトルは $\mathbf{J}_i \in \mathbb{R}^N \overset{\text{iid}}{\sim} \mathcal{N}(0, 1/N)$，$\mathbf{B}_n \in \mathbb{R}^N \overset{\text{iid}}{\sim} \mathcal{N}(0, 1/N)$ と初期化をする．今ソフトコミティを考えているので $\mathbf{w}_i \in \mathbb{R}^O$，$\mathbf{v}_n \in \mathbb{R}^O$ は要素が定数の O 次元のベクトルで，値が不変である．ただし i, n はそれぞれ生徒と教師の中間層の素子のインデックスである．中間層の活性化関数を ϕ と

し，出力層の活性化関数は恒等写像とする．このとき，生徒と教師のネットワークの出力はそれぞれ，

$$\mathbf{s} \in \mathbb{R}^O = \sum_i^K \mathbf{w}_i \phi\left(\mathbf{J}_i \cdot \boldsymbol{\xi}\right), \tag{1}$$

$$\mathbf{t} \in \mathbb{R}^O = \sum_n^M \mathbf{v}_n \phi\left(\mathbf{B}_n \cdot \boldsymbol{\xi}\right), \tag{2}$$

と書ける．損失関数としては二乗損失 $\varepsilon = \frac{1}{2}||\mathbf{t}-\mathbf{s}||^2$ を用いて，生徒ネットワークの重みを教師ネットワークの重みに近づけていく．ここで，系の大域的な挙動を記述するパラメータであるオーダーパラメータを次のように定義する：$Q_{ij} = \mathbf{J}_i \cdot \mathbf{J}_j$，$R_{in} = \mathbf{J}_i \cdot \mathbf{B}_n$，$T_{nm} = \mathbf{B}_n \cdot \mathbf{B}_m$，$D_{ij} = \mathbf{w}_i \cdot \mathbf{w}_j$，$E_{in} = \mathbf{w}_i \cdot \mathbf{v}_n$，$F_{nm} = \mathbf{v}_n \cdot \mathbf{v}_m$ [Saad 95, Biehl 95, Yoshida 18]．入力素子数 N が十分に大きい時，活性化関数によってはいくつかの理想化のもとでこれらのパラメータの微分方程式を導出することができる [Saad 95, Biehl 95]．また，訓練誤差の $\boldsymbol{\xi}$ についての期待値として定義される汎化誤差 ε_g はオーダーパラメータの関数となるので，汎化誤差のダイナミクスも導出することができる．Saad と Solla は 3 層ソフトコミティの入力層から中間層への重みについて，Biehl と Schwarze は一般の 3 層を対象として入力層から中間層への重みについて，Yoshida らは 3 層の全ての重みについて，オーダーパラメータのダイナミクスを導出している [Saad 95, Biehl 95, Yoshida 18]．我々はこのうちソフトコミティについてのダイナミクスを Batch Normalization ありの場合に拡張した．

2.2 Batch Normalization の統計力学定式化

従来の統計力学的定式化ではサンプルサイズ 1 の学習を取り扱っていたため，これを Batch Normalization を取り扱えるように拡張した．各更新毎に新しく b 個の入力を i.i.d. にサンプリングし，それを用いて学習を行うものとする．サンプルサイズ b のデータの中の 2 つのサンプル $\boldsymbol{\xi}^u$，$\boldsymbol{\xi}^v$ が互いに無相関だと仮定する．すると，Q_{ij} のダイナミクスに一部修正を加えるだけで，Batch Normalization を取り扱えるように統計力学的手法を自然に拡張できることを確認した．

Batch Normalization では，各中間素子への入力 $x_i^u = \mathbf{J}_i \boldsymbol{\xi}^u$ それぞれに対して，ミニバッチデータについての算術平均と標準偏差で正規化したものに学習可能パラメータ g_i をかけて β_i を足したものを活性化関数への入力とする．ここでは解析の簡単のため，算術平均を引く操作と β を足す操作を行わず，標準偏差は定数とする．今，入力に期待値 0 の分布を仮定しているので，サンプルサイズ b が十分大きいとき，中間層の各素子の b 個の入力についての標準偏差 $\sigma_{x_i} = \sqrt{\frac{1}{b}\sum_{u=1}^b (x_i^u - \mu_i)^2}$ は $\sigma_{x_i} \approx \sqrt{\frac{b}{b-1}\langle x_i^2 \rangle} \approx \sqrt{\mathbf{J}_i^T \langle \boldsymbol{\xi}^u \boldsymbol{\xi}^{uT} \rangle \mathbf{J}_i} = \sqrt{\sigma^2 ||\mathbf{J}_i||^2} = \sigma\sqrt{Q_{ii}}$ となり，$\boldsymbol{\xi}$ に依存しなくなる．ただし $\langle\cdot\rangle$ は入力 $\boldsymbol{\xi}$ についての期待値をとる操作である．この時，生徒の出力は，

$$\mathbf{s}^u = \sum_i^K \mathbf{w}_i \phi\left(\frac{g_i}{\sigma\sqrt{Q_{ii}}} \mathbf{J}_i \boldsymbol{\xi}^u\right) = \sum_i^K \mathbf{w}_i \phi\left(\frac{g_i}{\sigma\sqrt{Q_{ii}}} x_i^u\right), \tag{3}$$

となる．そして生徒の第 1 層の重みと学習可能パラメータ g_i の更新式はそれぞれ，

$$\Delta \mathbf{J}_i = \frac{\eta}{Nb} \sum_{u=1}^b [(\mathbf{t}^u - \mathbf{s}^u) \cdot \mathbf{w}_i] \, \phi'\left(\frac{g_i}{\sigma\sqrt{Q_{ii}}} x_i^u\right) \frac{g_i}{\sigma\sqrt{Q_{ii}}} \boldsymbol{\xi}^u, \tag{4}$$

$$\Delta g_i = \frac{\eta}{Nb} \sum_{u=1}^b [(\mathbf{t}^u - \mathbf{s}^u) \cdot \mathbf{w}_i] \, \phi'\left(\frac{g_i}{\sigma\sqrt{Q_{ii}}} x_i^u\right) \frac{\mathbf{J}_i \boldsymbol{\xi}^u}{\sigma\sqrt{Q_{ii}}}, \tag{5}$$

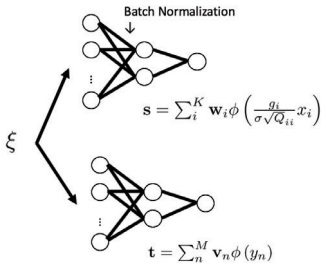


図 1: Batch Normalization ありの場合の 3 層ソフトコミティの教師生徒型学習の図．入力素子数が N，中間素子数が $K = 2, M = 2$，出力素子数が $O = 1$ の場合．$\boldsymbol{\xi}$ を共通の入力として出力された $\mathbf{s}$ と $\mathbf{t}$ の間の誤差を小さくするように生徒が重みを調節する．

と書ける．ただし $\frac{\eta}{N}$ は学習率を表す．$\frac{g_i}{\sigma\sqrt{Q_{ii}}} x_i = \hat{x}_i$ と書くと，オーダーパラメータと g_i の更新式は，

$$\begin{aligned}
&\Delta Q_{ij} \\
=&\frac{\eta}{bN} \sum_{u=1}^b \left[\sum_{p=1}^M E_{ip} \phi'(\hat{x}_i^u) \hat{x}_j^u \phi(y_p^u) - \sum_{p=1}^K D_{ip} \phi'(\hat{x}_i^u) \hat{x}_j^u \phi(\hat{x}_p^u) \right. \\
&\left. + \sum_{p=1}^M E_{jp} \phi'(\hat{x}_j^u) \hat{x}_i^u \phi(y_p^u) - \sum_{p=1}^K D_{jp} \phi'(\hat{x}_j^u) \hat{x}_i^u \phi(\hat{x}_p^u) \right] \\
&+ \frac{\eta^2}{b^2 N^2} \sum_{u,v}^{b,b} \boldsymbol{\xi}^u \boldsymbol{\xi}^v \left[\sum_{p,q}^{K,K} D_{ip} D_{jq} \phi'(\hat{x}_i^u) \phi'(\hat{x}_j^v) \phi(\hat{x}_p^u) \phi(\hat{x}_q^v) \right. \\
&+ \sum_{p,q}^{M,M} E_{ip} E_{jq} \phi'(\hat{x}_i^u) \phi'(\hat{x}_j^v) \phi(y_p^u) \phi(y_q^v) \\
&- \sum_{p,q}^{K,M} D_{ip} E_{jq} \phi'(\hat{x}_i^u) \phi'(\hat{x}_j^v) \phi(\hat{x}_p^u) \phi(y_q^v) \\
&\left. - \sum_{p,q}^{M,K} E_{ip} D_{jq} \phi'(\hat{x}_i^u) \phi'(\hat{x}_j^v) \phi(y_p^u) \phi(\hat{x}_q^v) \right],
\end{aligned} \tag{6}$$

$$\begin{aligned}
&\Delta R_{in} \\
=&\frac{\eta}{bN} \sum_{u=1}^b \left[\sum_{p=1}^M E_{ip} \phi'(\hat{x}_i^u) y_n^u \phi(y_p^u) - \sum_{p=1}^K D_{ip} \phi'(\hat{x}_i^u) y_n^u \phi(\hat{x}_p^u) \right],
\end{aligned} \tag{7}$$

$$\begin{aligned}
&\Delta g_i \\
=&\frac{\eta}{bN g_i} \sum_{u=1}^b \left[\sum_{p=1}^M E_{ip} \phi'(\hat{x}_i^u) \hat{x}_i^u \phi(y_p^u) - \sum_{p=1}^K D_{ip} \phi'(\hat{x}_i^u) \hat{x}_i^u \phi(\hat{x}_p^u) \right],
\end{aligned} \tag{8}$$

と書ける．また，汎化誤差は，

$$\begin{aligned}
\varepsilon_g =& \frac{1}{2} \left[\sum_{p,q}^{M,M} F_{pq} \phi(y_p^u) \phi(y_q^u) + \sum_{p,q}^{K,K} D_{pq} \phi(\hat{x}_p^u) \phi(\hat{x}_q^u) \right. \\
&\left. - 2 \sum_{p,q}^{K,M} E_{pq} \phi(\hat{x}_p^u) \phi(y_q^u) \right],
\end{aligned} \tag{9}$$

となる．ただし $y_n^u = \mathbf{B}_n \boldsymbol{\xi}^u$ である．この更新式を $\boldsymbol{\xi}$ について期待値を取ったものは，活性化関数によっては厳密に計算すること

ができる [Saad 95, Biehl 95, Yoshida 18]．$\phi(x) = \mathrm{erf}(x/\sqrt{2})$ のとき，これらのオーダーパラメータと g_i，そして汎化誤差のダイナミクスは以下のように求まる *1：

$$N\frac{\mathrm{d}Q_{ij}}{\mathrm{d}t} = \frac{2\eta}{\pi}\left[\mathcal{Q}_1 - \mathcal{Q}_2\right] + \frac{4\eta^2 g_i g_j}{b\pi^2\sqrt{\Lambda Q_{ii}Q_{jj}}}\left[\mathcal{Q}_3 + \mathcal{Q}_4 - \mathcal{Q}_5 - \mathcal{Q}_6\right], \tag{24}$$

$$N\frac{\mathrm{d}R_{in}}{\mathrm{d}t} = \frac{2\eta}{\pi}\left[\mathcal{R}\right], \tag{25}$$

$$N\frac{\mathrm{d}g_i}{\mathrm{d}t} = \frac{2\eta}{\pi g_i}\left[\mathcal{G}\right]. \tag{26}$$

ただし $\mathrm{d}t$ は微小な変化量で，$(l,k) = (i,j,n,p,q)$ について $Q'_{lk} = \frac{\sigma^2 g_l g_k}{\sigma_{x_l}\sigma_{x_k}} Q_{lk}, R'_{lk} = \frac{\sigma^2 g_l}{\sigma_{x_l}} R_{lk}, T'_{lk} = \sigma^2 T_{lk}$ である．

*1

$$\mathcal{Q}_1 = \sum_{p=1}^{M}\left[\frac{E_{ip}\left(R'_{jp}(1+Q'_{ii}) - Q'_{ij}R'_{ip}\right)}{(1+Q'_{ii})\sqrt{(1+Q'_{ii})(1+T'_{pp}) - R'^2_{ip}}} + \frac{E_{jp}\left(R'_{ip}(1+Q'_{jj}) - Q'_{ji}R'_{jp}\right)}{(1+Q'_{jj})\sqrt{(1+Q'_{jj})(1+T'_{pp}) - R'^2_{jp}}}\right] \tag{10}$$

$$\mathcal{Q}_2 = \sum_{p=1}^{K}\left[\frac{D_{ip}\left(Q'_{jp}(1+Q'_{ii}) - Q'_{ij}Q'_{ip}\right)}{(1+Q'_{ii})\sqrt{(1+Q'_{ii})(1+Q'_{pp}) - Q'^2_{ip}}} + \frac{D_{jp}\left(Q'_{ip}(1+Q'_{jj}) - Q'_{ji}Q'_{jp}\right)}{(1+Q'_{jj})\sqrt{(1+Q'_{jj})(1+Q'_{pp}) - Q'^2_{jp}}}\right] \tag{11}$$

$$\mathcal{Q}_3 = \sum_{p,q}^{K,K} D_{ip}D_{jq} \times \mathrm{asin}\left(\frac{\Lambda Q'_{pq} - Q'_{jp}Q'_{jq}(1+Q'_{ii}) - Q'_{ip}Q'_{iq}(1+Q'_{jj}) + Q'_{ij}Q'_{ip}Q'_{jq} + Q'_{ij}Q'_{iq}Q'_{jp}}{\sqrt{\Lambda_1\Lambda_2}}\right) \tag{12}$$

$$\mathcal{Q}_4 = \sum_{p,q}^{M,M} E_{ip}E_{jq} \times \mathrm{asin}\left(\frac{\Lambda T'_{pq} - R'_{jp}R'_{jq}(1+Q'_{ii}) - R'_{ip}R'_{iq}(1+Q'_{jj}) + Q'_{ij}R'_{ip}R'_{jq} + Q'_{ij}R'_{iq}R'_{jp}}{\sqrt{\Lambda_3\Lambda_4}}\right) \tag{13}$$

$$\mathcal{Q}_5 = \sum_{p,q}^{K,M} D_{ip}E_{jq} \times \mathrm{asin}\left(\frac{\Lambda R'_{pq} - Q'_{jp}R'_{jq}(1+Q'_{ii}) - Q'_{ip}R'_{iq}(1+Q'_{jj}) + Q'_{ij}Q'_{ip}R'_{jq} + Q'_{ij}R'_{iq}Q'_{jp}}{\sqrt{\Lambda_1\Lambda_4}}\right) \tag{14}$$

$$\mathcal{Q}_6 = \sum_{p,q}^{M,K} E_{ip}D_{jq} \times \mathrm{asin}\left(\frac{\Lambda R'_{pq} - R'_{jp}Q'_{jq}(1+Q'_{ii}) - R'_{ip}Q'_{iq}(1+Q'_{jj}) + Q'_{ij}R'_{ip}Q'_{jq} + Q'_{ij}Q'_{iq}Q'_{jp}}{\sqrt{\Lambda_2\Lambda_3}}\right) \tag{15}$$

$$\mathcal{R} = \sum_{p=1}^{M}\frac{E_{ip}\left(T'_{np}(1+Q'_{ii}) - R'_{in}R'_{ip}\right)}{(1+Q'_{ii})\sqrt{(1+Q'_{ii})(1+T'_{pp}) - R'^2_{ip}}} - \sum_{p=1}^{K}\frac{D_{ip}\left(R'_{pn}(1+Q'_{ii}) - R'_{in}Q'_{ip}\right)}{(1+Q'_{ii})\sqrt{(1+Q'_{ii})(1+Q'_{pp}) - Q'^2_{ip}}} \tag{16}$$

$$\mathcal{G} = \sum_{p=1}^{M}\frac{E_{ip}\left(R'_{ip}(1+Q'_{ii}) - Q'_{ii}R'_{ip}\right)}{(1+Q'_{ii})\sqrt{(1+Q'_{ii})(1+T'_{pp}) - R'^2_{ip}}} - \sum_{p=1}^{K}\frac{D_{ip}\left(Q'_{ip}(1+Q'_{ii}) - Q'_{ii}Q'_{ip}\right)}{(1+Q'_{ii})\sqrt{(1+Q'_{ii})(1+Q'_{pp}) - Q'^2_{ip}}} \tag{17}$$

$$\Lambda = (1+Q'_{ii})(1+Q'_{jj}) - (1+Q'^2_{ij}) \tag{18}$$

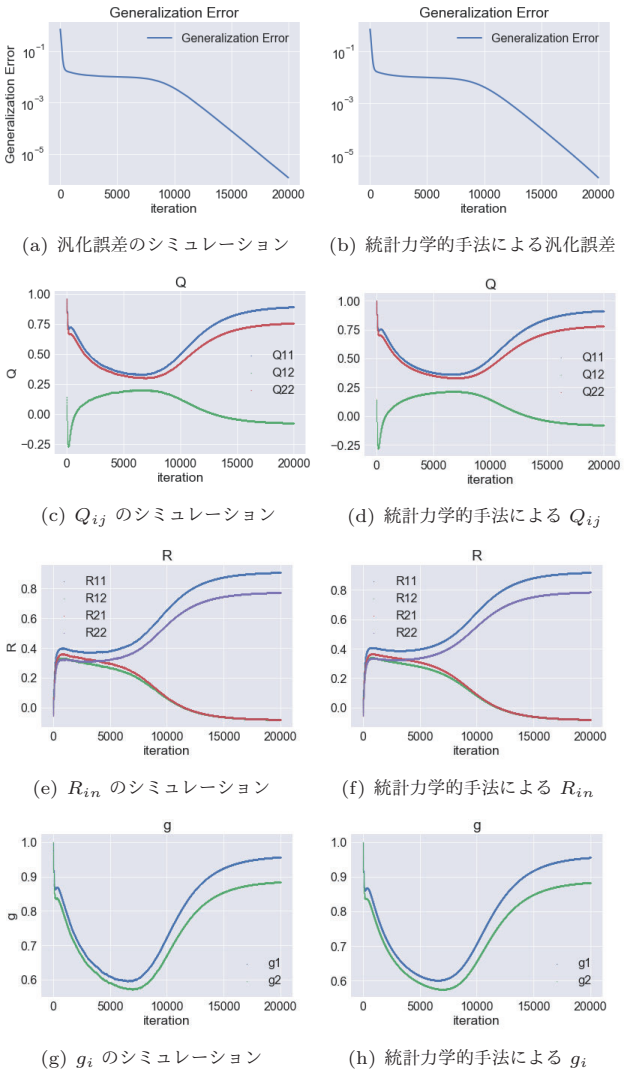
$$\Lambda_1 = \Lambda(1+Q'_{pp}) - Q'^2_{jp}(1+Q'_{ii}) - Q'^2_{ip}(1+Q'_{jj}) + 2Q'_{ij}Q'_{ip}Q'_{jp} \tag{19}$$

$$\Lambda_2 = \Lambda(1+Q'_{qq}) - Q'^2_{jq}(1+Q'_{ii}) - Q'^2_{iq}(1+Q'_{jj}) + 2Q'_{ij}Q'_{iq}Q'_{jq} \tag{20}$$

$$\Lambda_3 = \Lambda(1+T'_{pp}) - R'^2_{jp}(1+Q'_{ii}) - R'^2_{ip}(1+Q'_{jj}) + 2Q'_{ij}R'_{ip}R'_{jp} \tag{21}$$

$$\Lambda_4 = \Lambda(1+T'_{qq}) - R'^2_{jq}(1+Q'_{ii}) - R'^2_{iq}(1+Q'_{jj}) + 2Q'_{ij}R'_{iq}R'_{jq} \tag{22}$$

$$\varepsilon_g = \frac{1}{\pi}\left[\sum_{p,q}^{M,M}\mathrm{asin}\left(\frac{T'_{pq}}{\sqrt{(1+T'_{pp})(1+T'_{qq})}}\right) - \sum_{p,q}^{K,K}\mathrm{asin}\left(\frac{Q'_{pq}}{\sqrt{(1+Q'_{pp})(1+Q'_{qq})}}\right) - 2\sum_{p,q}^{K,M}\mathrm{asin}\left(\frac{R'_{pq}}{\sqrt{(1+Q'_{pp})(1+T'_{qq})}}\right)\right] \tag{23}$$



(a) 汎化誤差のシミュレーション　(b) 統計力学的手法による汎化誤差

(c) Q_{ij} のシミュレーション　(d) 統計力学的手法による Q_{ij}

(e) R_{in} のシミュレーション　(f) 統計力学的手法による R_{in}

(g) g_i のシミュレーション　(h) 統計力学的手法による g_i

図 2: (a), (c), (e), (g) が数値シミュレーションの結果で (b), (d), (f), (h) が統計力学的手法を用いて導出したダイナミクス．$\eta = 1$, $N = 100$，$b = 100$，$\sigma = 1$，$\mathbf{w}_i = 1, \mathbf{v}_n = 1$ とし，総イテレーション数は 20000 とした

2.3 数値シミュレーションと統計力学的定式化により導出したダイナミクスの一致

統計力学的手法を用いたダイナミクスの導出では近似を用いている．そのため，統計力学的手法によって導出されたダイナミクスと実際の重みの更新式を用いた数値シミュレーションの結果が一致することを確認する必要がある．そこで $K = 2, M = 2, O = 1$ の場合について Q_{ij}, R_{in}, g_i および ε_g の時間発展を比較した．図 2 が数値シミュレーションの結果と統計力学的手法によって導出されたダイナミクスの比較である．図より，これらの二つの結果はよく一致しており，統計力学的手法の近似は妥当であることがわかる．

3. まとめ

Saad らが発展させた統計力学的手法を用いて Batch Normalization がある場合の 3 層ニューラルネットワークのオーダーパラメータおよび汎化誤差のダイナミクスを導出した．こ

れを用いれば Batch Normalization がニューラルネットワークの学習挙動にどのような影響を与えるかを解析することができる．

例えば，ニューラルネットワークの学習では学習初期と終期ではデータから学習する構造が異なると考えられており，それがニューラルネットワークが高い表現能力を持ちながら良い汎化性能を示す原因としてあげられることがある [Saxe 18, Xu 18,Krueger 17,Rahaman 18,Arpit 18]．Batch Normalization がそれぞれの時期の学習にどのような影響を与えるのかを分析することは Batch Normalization がなぜうまくいくのかを理解する上で重要であり，本稿で導出したダイナミクスを解析することでそのような分析が可能となることが期待できる．

参考文献

[Arora 18] Arora, S., Li, Z., and Lyu, K.: Theoretical Analysis of Auto Rate-Tuning by Batch Normalization, *arXiv preprint arXiv:1812.03981* (2018)

[Arpit 18] Arpit, D., Jastrzebski, S., Ballas, N., Krueger, D., Bengio, E., Kanwal, M. S., Maharaj, T., Fischer, A., Courville, A., Bengio, Y., and Lacoste-Julien, S.: A Closer Look at Memorization in Deep Networks, *ICML* (2018)

[Biehl 95] Biehl, M. and Schwarze, H.: Learning by on-line gradient descent, *Journal of Physics A: Mathematical and General*, Vol. 28, No. 3, p. 643 (1995)

[Bjorck 18] Bjorck, J., Gomes, G., Selman, B., and Weinberger, K. Q.: Understanding Batch Normalization, *NeurIPS 2018* (2018)

[Ioffe 15] Ioffe, S. and Szegedy, C.: Batch Normalization: Accelerating Deep Network Training by Reducing Internal Covariate Shift, in *ICML*, pp. 448–456 (2015)

[Kohler 18] Kohler, J., Daneshmand, H., Lucchi, A., Zhou, M., Neymeyr, K., and Hofmann, T.: Exponential convergence rates for Batch Normalization: The power of length-direction decoupling in non-convex optimization, *arXiv preprint arXiv:1805.10694* (2018)

[Krueger 17] Krueger, D., Ballas, N., Jastrzebski, S., Arpit, D., Kanwal, M. S., Maharaj, T., Bengio, E., Fischer, A., and Courville, A.: Deep Nets Don't Learn Via Memorization, *ICLR Workshop* (2017)

[Luo 18] Luo, P., Wang, X., Shao, W., and Peng, Z.: Towards Understanding Regularization in Batch Normalization, *arXiv preprint arXiv:1809.00846* (2018)

[Rahaman 18] Rahaman, N., Baratin, A., Arpit, D., Draxler, F., Lin, M., Hamprecht, F. A., Bengio, Y., and Courville, A.: On The Spectral Bias of Neural Networks, *NeurIPS Workshop* (2018)

[Riegler 95] Riegler, P. and Biehl, M.: On-line backpropagation in two-layered neural networks, *Journal of Physics A*, Vol. 28, pp. L507–L513 (1995)

[Saad 95] Saad, D. and Solla, S. A.: Exact Solution for On-Line Learning in Multilayer Neural Networks, *Physical Review Letters*, Vol. 74, No. 41, pp. 4337–4340 (1995)

[Santurkar 18] Santurkar, S., Tsipras, D., Ilyas, A., and Mardy, A.: How Does Batch Normalization Help Optimization?, *arXiv preprint arXiv:1805.11604* (2018)

[Saxe 18] Saxe, A. M., McClelland, J. L., and Ganguli, S.: A mathematical theory of semantic development in deep neural networks, *arXiv preprint arXiv:1810.1053* (2018)

[Schwarze 93] Schwarze, H.: Learning a rule in a multilayer neural network, *Journal of Physics A*, Vol. 26, pp. 5781–5794 (1993)

[Seung 92] Seung, H. S., Somopolinsky, H., and Tishby, N.: Statistical mechanics of learning from examples, *Physical Review A*, Vol. 45, No. 8, pp. 6056–6091 (1992)

[Xu 18] Xu, Z.-Q. J.: Understanding training and generalization in deep learning by Fourier analysis, *arXiv preprint arXiv:1808.04295* (2018)

[Yoshida 18] Yoshida, Y., Karakida, R., Okada, M., and Amari, S.: Statistical Mechanical Analysis of Learning Dynamics of Two-Layer Perceptron with Multiple Output Units, *J. Phys. A (provisionally accepted)* (2018)