4M1-4

Weight Normalizationを用いた2層パーセプトロンのオンライン学習の統計力学的解析

Analysis of On-line Learning with Weight Normalization in Single Layer Perceptron Using Statistical Mechanical Approach

吉田雄紀[*1] Yuki Yoshida　唐木田亮[*2] Ryo Karakida　岡田真人[*1*2*3] Masato Okada　甘利俊一[*3] Shun-ichi Amari

[*1]東京大学大学院 新領域創成科学研究科 複雑理工学専攻
Department of Complexity Science and Engineering, Graduate School of Frontier Sciences, The University of Tokyo

[*2]産業技術総合研究所 人工知能研究センター
AIST The Artificial Intelligence Research Center

[*3]理化学研究所 脳科学総合研究センター
RIKEN Brain Science Institute

Weight normalization (WN), a newly developed optimization algorithm for neural networks by Salimans & Kingma(2016), factorizes the weight vector of a neural network into a radial length and a direction vector, and the factorized parameters follow their steepest gradient descent update. They report that WN realizes faster learning speed than standard stochastic gradient descent (SGD) does in various tasks. However, it is theoretically obscure why this method works well. In this research, we formulate on-line learning with WN in a statistical mechanical fashion, and derive order parameters of the dynamics of learning. We show quantitatively that WN achieves fast learning speed by automatically tuning the effective learning rate, and discuss its parameter dependency.

1. はじめに

近年，深層学習技術 [LeCun 15] のめざましい発展により，大規模なニューラルネットワークが様々なタスクに対して用いられるようになった．その成功は，勾配法の修正 [Duchi 11,Zeiler 12,Kingma 14,Tieleman 12,Amari 98] や，種々の正規化 [Ioffe 15,Salimans 16,Ba 16] など，アルゴリズムの様々な改良に依って実現してきたといえる．とりわけ，Salimans & Kingma によって 2016 年に提案された weight normalization [Salimans 16] は，結合加重の値を以下のように再パラメータ化する手法であり，様々な種類のニューラルネットワークに自然に組み込めること，および，実装が容易いことから，最近注目を浴びている．一般的なニューラルネットワークにおいて，その各素子（ニューロン）は，入力ベクトル $\boldsymbol{x}$ に対して $y=g(\boldsymbol{W}\cdot\boldsymbol{x}+b)$（$g$ は活性化関数と呼ばれ，一般には非線形）という値を出力するが，このニューロンの結合加重ベクトル $\boldsymbol{W}$ は，通常の最急勾配法では，ε を誤差（損失関数）として $\Delta\boldsymbol{W}=-\eta\dfrac{\partial\varepsilon}{\partial\boldsymbol{W}}$ に従って更新されていく（$\eta>0$ は学習係数）．一方，weight normalization では，$\boldsymbol{W}=r\dfrac{\boldsymbol{V}}{|\boldsymbol{V}|}$ と分解した上で，r と $\boldsymbol{V}$ を新たなパラメータとみなして勾配法を行う．すなわち，$\Delta r=-\eta\dfrac{\partial\varepsilon}{\partial r}, \Delta\boldsymbol{V}=-\eta\dfrac{\partial\varepsilon}{\partial\boldsymbol{V}}$ に従って r と $\boldsymbol{V}$ が更新されていく．この weight normalization を用いると，画像認識や強化学習タスクにおいて高速な学習の収束が実現する [Salimans 16]．しかし，そのメカニズムは十分に解明されていない．

オンライン学習のダイナミクスの解析手法として，結合加重ベクトルの挙動を巨視的に捉えるオーダパラメータのダイナミクスを学習則から導出して解析する統計力学的手法が，[Biehl 95], [Saad 95] らによって確立されている．彼らは，学習を行うニューラルネットワーク（生徒ネットワーク）と同構造をした「教師ネットワーク」の入出力関係を学習するという問題設定の下で，2 層のパーセプトロン，および 3 層の soft-committee machine のオンライン教師あり学習におけるオーダパラメータのダイナミクスを解析的に導出し，その性質を論じた．

本稿では，weight normalization に対してオンライン学習の統計力学的解析を適用し，通常の勾配法および weight normalization を用いた場合のオンライン学習のダイナミクスを定量的に比較する．具体的には，大域解周囲での安定性解析から，両者の学習則におけるオーダパラメータの収束速度を求め，自動調整される実効的な学習係数が導出されること，および，収束を高速化するオーダパラメータの初期値が存在することを示す．また，その他のパラメータへの学習の依存性も論じる．

2. モデル

2.1 生徒ネットワークと教師ネットワークによる定式化

本論文では，2 層パーセプトロンによる学習を扱う．すなわち，入力データ $\boldsymbol{x}\in\mathbb{R}^N$ に対して $s=g(\boldsymbol{J}\cdot\boldsymbol{x})$（ただし活性化関数 $g:\mathbb{R}\to\mathbb{R}$ は，定数関数ではない広義単調関数であるものとする）を出力するニューラルネットワークの学習を考える．理想的な状況として，教師データ t が $t=g(\boldsymbol{B}\cdot\boldsymbol{x})$ により定められる状況を考える．すなわち，学習を行うニューラルネットワーク（生徒ネットワーク）は，生徒ネットワークと構造が同じで結合加重の異なる「教師ネットワーク」の入出力関係を学習する（図 1 (a)）．誤差 ε としては，ここでは最も典型的な二乗損失 $\varepsilon=\dfrac{1}{2}(t-s)^2$ を考えるが，ソフトマックス損失など他の損失に対しても，本稿と同様の議論を行うことが可能である．

2.2 統計力学的定式化：通常の勾配法の場合

オンライン学習を統計力学的に定式化するために，さらにいくつかの理想化を行う．まず，入力素子数 N は十分大きいものとし，入力データ $\boldsymbol{x}$ は，その各要素が独立な正規分布 $\mathcal{N}(x_i|0,1/N)$ から生成されるとする（$|\boldsymbol{x}|\approx 1$ に注意されたい）．そして，$|\boldsymbol{B}|=\sqrt{N}$ と仮定し，また $|\boldsymbol{J}|=\sqrt{N}\,l(\alpha)$, $\boldsymbol{B}\cdot\boldsymbol{J}=N\,l(\alpha)R(\alpha)$ とおく（α は時間を表す）[Biehl 95,Saad 95]．このとき，$l(\alpha)$ と $R(\alpha)$ がオーダパラメータであり，前者は生徒の結合加重ベクトルの長さ（ノルム）を，後者は生徒と教師の結合加重ベクトルがなす方向余弦を表す（図 1 (b)）．l, R の初期値は，結合加重ベクトル $\boldsymbol{J}$ の初期化の方法によっ

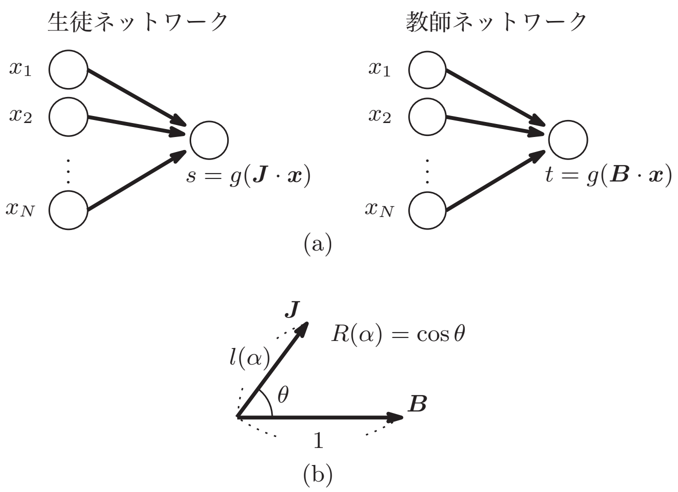

図 1: (a) 生徒ネットワークと教師ネットワーク．(b) オーダパラメータ $l(\alpha)$, $R(\alpha)$ の幾何学的解釈．

て様々な値となりうるが，原点について球対称な分布からサンプリングする場合は，$N \to \infty$ の下で R の初期値は 0 に確率収束する．

次節にて，系の状態を巨視的に捉えるオーダパラメータ l, R が従うダイナミクスを，系の微視的な変数（結合加重ベクトル）のダイナミクスから導出する．

2.3 統計力学的定式化：weight normalization の場合

weight normalization では，結合加重ベクトル $\boldsymbol{J}$ を，動径長 r と方向ベクトル $\boldsymbol{V}$ に分解し（$\boldsymbol{J} = r\dfrac{\boldsymbol{V}}{|\boldsymbol{V}|}$），この r と $\boldsymbol{V}$ について勾配法を行う．前小節にて $|\boldsymbol{J}| = \sqrt{N}\,l(\alpha)$ と定めたため $r = \sqrt{N}\,l(\alpha)$ であり，以降は $l(\alpha)$ のことを動径長と呼ぶことにする．また，$\boldsymbol{V}$ のノルムを $|\boldsymbol{V}| = \sqrt{N}\,z(\alpha)$ と定める．weight normalization の場合は，$l(\alpha)$, $R(\alpha)$ に $z(\alpha)$ を加えた 3 つのオーダパラメータが従うダイナミクスを次節にて導出する．

3. 理論

本節では，前節で述べたオーダパラメータらのダイナミクスを導出する．

3.1 通常の勾配法の場合のオーダパラメータのダイナミクス

通常の確率的最急勾配法によるオンライン学習における，結合加重 $\boldsymbol{J}$ の更新則は

$$\Delta\boldsymbol{J} = -\eta\frac{d\varepsilon}{d\boldsymbol{J}} = \eta g'(\boldsymbol{J}\cdot\boldsymbol{x})(t-s)\boldsymbol{x} \tag{1}$$

と書けるが，式 (1) より，オーダパラメータ l と R に関する差分方程式を導出でき，さらに $N \to \infty$ により，以下の微分方程式が得られる [Biehl 95, Saad 95]：

$$\begin{aligned} N\frac{d}{d\alpha}l^2 &= 2\eta A_1 + 2\eta^2 A_2, \\ N\frac{d}{d\alpha}lR &= \eta A_3 \end{aligned} \tag{2}$$

ただし　$\boldsymbol{J}\cdot\boldsymbol{x} = lu,\ \boldsymbol{B}\cdot\boldsymbol{x} = v$　として

$$\begin{aligned} A_1(l,R) &= \langle g'(lu)(g(v)-g(lu))lu\rangle, \\ A_2(l,R) &= \frac{1}{2}\langle g'(lu)^2(g(v)-g(lu))^2\rangle, \\ A_3(l,R) &= \langle g'(lu)(g(v)-g(lu))v\rangle. \end{aligned} \tag{3}$$

ここで山括弧 $\langle\cdot\rangle$ は，$\boldsymbol{x}$ が $\mathcal{N}(x_i|0,1/N)$ に従って動く（このとき u, v は確率分布 $\mathcal{N}(\boldsymbol{0}, \begin{pmatrix} 1 & R \\ R & 1 \end{pmatrix})$ に従う）下での期待値を表す．また，このとき，汎化誤差 ε_g は

$$\varepsilon_g = \frac{1}{2}\langle(t-s)^2\rangle = \frac{1}{2}\langle(g(v)-g(lu))^2\rangle \tag{4}$$

と表される [Biehl 95, Saad 95]．これらの式に登場する期待値は，活性化関数 g の具体形によっては，解析的に求めることが可能である（例えば，$g(x) = x$ および $g(x) = \mathrm{erf}(x/\sqrt{2})$ の場合 [Biehl 95, Saad 95]，および $g(x) = \mathrm{ReLU}(x)$ の場合には可能である）．

3.2 weight normalization の場合のオーダパラメータのダイナミクス

weight normalization によるオンライン学習における，方向ベクトル $\boldsymbol{V}$ および動径パラメータ r の更新則は

$$\begin{aligned} \Delta r &= -\eta\frac{d\varepsilon}{dr} = \eta g'(\boldsymbol{J}\cdot\boldsymbol{x})(t-s)\frac{\boldsymbol{J}\cdot\boldsymbol{x}}{r}, \\ \Delta\boldsymbol{V} &= -\eta\frac{d\varepsilon}{d\boldsymbol{V}} = \eta g'(\boldsymbol{J}\cdot\boldsymbol{x})(t-s)\left(\frac{r}{|\boldsymbol{V}|}\boldsymbol{x} - \frac{\boldsymbol{J}\cdot\boldsymbol{x}}{|\boldsymbol{V}|^2}\boldsymbol{V}\right). \end{aligned} \tag{5}$$

と書けるが，式 (5) から，オーダパラメータ l, R, z に関する差分方程式を導出でき，さらに $N \to \infty$ により，以下の微分方程式が得られる：

$$\begin{aligned} N\frac{d}{d\alpha}l^2 &= 2\eta A_1, \\ N\frac{d}{d\alpha}Rz &= \eta(\frac{l}{z}A_3 - \frac{R}{z}A_1), \\ N\frac{d}{d\alpha}z^2 &= \eta^2\frac{2l^2}{z^2}A_2. \end{aligned} \tag{6}$$

A_i の定義は式 (3) と同じである．ここでも，期待値については前小節で述べた I_3 または I_4 の形の項しかなく，具体的な g によっては計算可能である．なお，$A_2 \geq 0$ のため，z は単調増加であることに注意されたい．

3.3 線形安定性解析

本稿で扱う 2 層パーセプトロンの系では，$\boldsymbol{J} = \boldsymbol{B}$ のときに限り，汎化誤差 ε_g が 0 に一致する．すなわち，$\boldsymbol{J} = \boldsymbol{B}$ は唯一の大域解である．この大域解において，オーダパラメータの値は $(l,R) = (1,1)$ である．通常の勾配法の系 (2) において，$(l,R) = (1,1)$ は平衡点となっており，また weight normalization の系 (6) において，$(l,R,z) = (1,1,z)$（z は任意）は平衡点となっている．大域解に対応するこれらの平衡点において，安定性解析を行うことにより，オーダパラメータ l, R の大域解への収束速度を定量的に評価することができる．その結果を表 1 に示す．

weight normalization と通常の勾配法を比較すると，方向余弦 R の収束速度が異なっており，weight normalization では，その「実効的な学習係数」が η ではなく η/z_∞^2 に置き換わっている．z の初期値は通常小さな値に設定する（ [Salimans 16]

	方向余弦 R	動径長 l
勾配法	$-\dfrac{\partial A_2}{\partial R}\eta(\eta_c-\eta)$	$\min\left\{-\dfrac{\partial A_1}{\partial l}\eta,\ -\dfrac{\partial A_2}{\partial R}\eta(\eta_c-\eta)\right\}$
WN	$-\dfrac{\partial A_2}{\partial R}\dfrac{\eta}{z_\infty^2}\left(\eta_c-\dfrac{\eta}{z_\infty^2}\right)$	$\min\left\{-\dfrac{\partial A_1}{\partial l}\eta,\ -\dfrac{\partial A_2}{\partial R}\dfrac{\eta}{z_\infty^2}\left(\eta_c-\dfrac{\eta}{z_\infty^2}\right)\right\}$

表 1: 通常の座標での勾配法と，weight normalization における，オーダパラメータ l, R の大域解への収束速度．ただし偏微分の項は，すべて大域解 $(l,R)=(1,1)$ における値を考えるものとする．

では 0.05 が推奨されている）ため，η/z^2 の初期値は大きな値で，そこから学習中に単調に減少していく（z の単調増加性より）．通常の勾配法の場合 $-\lambda_2 \propto \eta(\eta_c-\eta)$ であるため，学習係数が η_c よりも大きいと学習が収束せず，逆に小さすぎると学習の収束が遅い．最も $-\lambda_2$ が大きくなるのは学習係数が $\eta=\dfrac{\eta_c}{2}$ のときであるが，η_c の値は一般にはわからない．しかし weight normalization では，学習中にその実効的な学習係数 η/z^2 が減少していき，最適な学習係数 $\dfrac{\eta_c}{2}$ 付近の値に自動的に到達することで，本来の学習係数 η の設定値に依存しない高速な収束が実現しているものと予想される．

一方，動径長 l の収束速度は weight normalization でも $-\dfrac{\partial A_1}{\partial l}\eta$ より大きくなることはないため，通常法に比べて高速になることはない．そのため，動径長の初期値が大域解 ($l=1$) に近い場合には，weight normalization によって通常法よりも大幅に高速な収束が期待されるが，さもなければ，両者の勾配法で収束に要する時間に差があまり生じないことが予想される．また，動径長の初期値が大域解から顕著に外れていると，学習中に η/z^2 が小さくなりすぎて律速となり，通常法よりも weight normalization のほうが収束が遅くなる場合もあると考えられる．

次節では，オーダパラメータの微分方程式 (2) および (6) の数値解が，元のミクロな系での学習の数値シミュレーションの結果と一致することを確認し，さらに，収束速度に関して本節で述べた仮説を検証する．

4.　実験結果

4.1　数値シミュレーションと微分方程式の数値解の一致

通常の勾配法， weight normalization の場合それぞれについて，元のミクロな系（素子数 $N=10000$）の数値シミュレーションを行った．教師の結合加重 $\boldsymbol{B}$，生徒の結合加重の初期値 $\boldsymbol{J}(0)$，方向ベクトルの初期値 $\boldsymbol{V}(0)$ は，その各要素を $\mathcal{N}(B_i|0,1), \mathcal{N}(J_i|0,l_0^2), \mathcal{N}(V_i|0,z_0^2)$ に従ってサンプリングした後に $|\boldsymbol{B}|=\sqrt{N}$, $|\boldsymbol{J}(0)|=\sqrt{N}\,l_0$, $|\boldsymbol{V}(0)|=\sqrt{N}\,z_0$ をみたすように正規化して定めた．シミュレーションにおけるオーダパラメータ l, R, z の経時変化を，オーダパラメータに関する微分方程式（式 (2), (6)）の数値解と比較し（図 2），両者の結果がよく一致することを確認した．

4.2　学習収束速度の，学習係数・動径長初期値への依存性

次に，様々な学習係数 η や動径長初期値 l_0 に対して微分方程式 (2), (6) の数値解を求めることにより，汎化誤差の 0 への収束に要する時間の η, l_0 への依存性を調べ，通常の勾配法と weight normalization で比較し，図 3（$g(x)=\mathrm{erf}(x/\sqrt{2})$ の場合に，η, l_0 のそれぞれを動かした場合），および図 4（$g(x)=x$ の場合に，η, l_0 の両方を動かした場合）に示した．

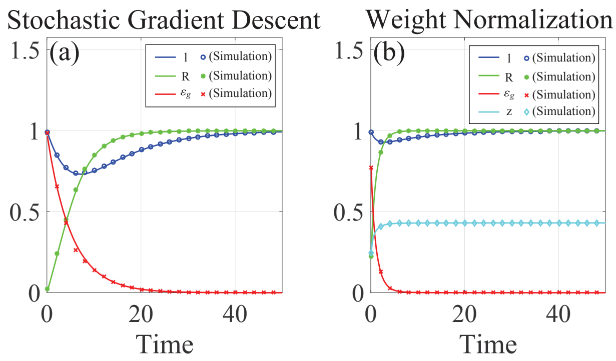

図 2: (a) 通常の勾配法，(b) weight normalization の場合の，オーダパラメータおよび汎化誤差の経時変化．中抜き丸：l，塗りつぶし丸：R，バツ印：ε_g，菱形：z（(b) のみ）．実線は微分方程式 (2), (6) の数値解で，記号は元の系 ($N=10000$) の数値シミュレーション結果である．l の初期値 l_0 は 1.0 である．また (b) では z の初期値 z_0 は 0.05 である．いずれの条件でも $\eta=0.1, g(x)=x$ である．

4.2.1　学習係数への依存性

学習係数 η が大きすぎる場合 ($\eta>\eta_c$ の場合)，通常の勾配法では結合加重が大域解へ収束しないが，weight normalization の場合には，学習係数の大きさに依らず，$\eta \approx \eta_c$ の場合と同程度の速度での大域解への収束が見られた．

大域解への収束の際に，方向余弦パラメータ R の「実効的な学習係数」η/z^2 は，多くのケースで，その最適な値である $\eta_c/2$ 前後まで実際に減少してきていた（図 4 (d)）．このことは，「学習係数の自動調整」によって η 非依存的な収束速度が実現していることを裏付けている．

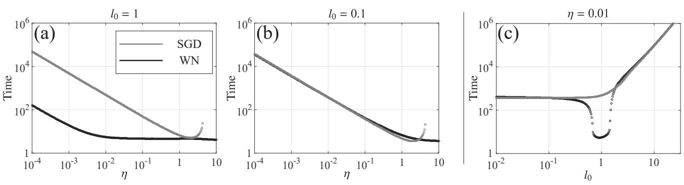

図 3: 汎化誤差 ε_g が 0.01 未満となるまでに要する時間の，(a)(b) 学習係数 η，(c) 動径長初期値 l_0 への依存性．(a) では $l_0=1$（大域解），(b) では $l_0=0.1$，(c) では $\eta=0.01$ に固定している．灰色：通常の勾配法，黒色：weight normalization．活性化関数は $g(x)=\mathrm{erf}(x/\sqrt{2})$ である．

4.2.2　動径長初期値への依存性

動径長の初期値が大域解付近のときに限り，広範囲のオーダ（図 3(a) では $10^{-2}<\eta<10$）の学習係数 η に対して，汎化誤差収束に要する時間が一定であった．通常の勾配法では，学習係数 η が $p(<1)$ 倍になれば汎化誤差収束にはおよそ $1/p$ 倍の時間が必要となるが， weight normalization では収束に要する時間が（上記の範囲内では）変化せず，η のときと同程度の速度で収束した．

一方，大域解よりも非常に大きな動径長初期値から開始した場合，weight normalization が通常法よりも収束に長時間を要するケースが存在した（図 4(c) の黄色い領域）．このよ

うなケースにおいて，方向余弦パラメータ R の「実効的な学習係数」η/z^2 は，l の収束速度を律する η よりも小さな値に到達しており,「実効的な学習係数」が減少しすぎたために収束が通常法よりも遅れたものと考えられる．なお，大域解よりも小さな動径長初期値から開始した場合には，このような現象は見られなかった．したがって，weight normalization を用いる際には，動径長の初期値に留意し，できる限り大域解に近いと考えられる初期値，少なくとも，大域解を大幅には上回らないような初期値を選ぶことが重要と考えられる．

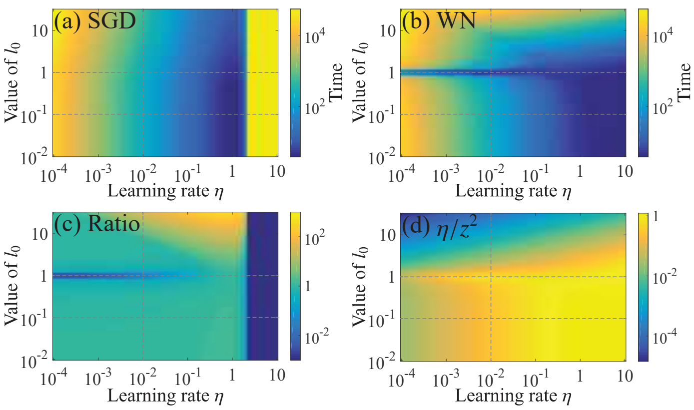

図 4: 汎化誤差 ε_g が 0.01 未満となるまでに要する時間の，学習係数 η，動径長初期値 l_0 への依存性（2 次元プロット）．(a) 通常の勾配法の場合，(b) weight normalization の場合，(c) 両者の比（weight normalization / 通常法）．(d) weight normalization において $\varepsilon_g < 0.01$ となった時点での方向余弦の実効的な学習係数 η/z^2 の値．活性化関数は $g(x) = x$ である．各図の破線は，図 3 に対応している．

5. 結論

我々は，[Biehl 95], [Saad 95] らによって確立されたオンライン学習の統計力学的解析手法を用いて，[Salimans 16] によって提案された weight normalization を，最もシンプルな 2 層パーセプトロンの系で解析した．その結果，方向余弦の実効的な学習係数の「自動調整」によって学習が効率化されていることが，定量的に明らかとなった．本稿で行った理論解析は，活性化関数の形に依存しない．また，詳細は割愛したが，損失関数の形にも依存しない．すなわち，今回明らかとなった weight normalization の利点は，活性化関数や損失関数に依存しない weight normalization 固有の性質と考えられる．

今回は，weight normalization を 2 層パーセプトロンに適用した場合を議論したが，実応用においては，3 層以上のネットワークがしばしば用いられる．2 層パーセプトロンと異なり，多層の非線形パーセプトロンでは，汎化誤差曲面に特異点やプラトーが生じ，これらが汎化誤差の停滞の原因となることが知られているが [Riegler 95]，weight normalization は多層ネットワークに適用した場合にも収束速度の向上を認めており [Salimans 16]，特異点やプラトーの存在下においても weight normalization が高速な収束をもたらすことが示唆される．多層ネットワークのオンライン学習に関する統計力学的解析としては，[Biehl 95] および [Saad 95] によって 3 層ソフトコミティーマシンの学習の解析が行われており，また [Riegler 95] によって 3 層パーセプトロンの学習の解析が行われている．これらの統計力学的解析を応用して，3 層以上のパーセプトロンにおける weight normalization のダイナミクスの解析も行えるものと考えられる．

参考文献

[Amari 98] Amari, S.: Natural gradient works efficiently in learning, *Neural computation*, Vol. 10, No. 2, pp. 251–276 (1998)

[Ba 16] Ba, J. L., Kiros, J. R., and Hinton, G. E.: Layer Normalization, *arXiv preprint arXiv:1607.06450* (2016)

[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)

[Duchi 11] Duchi, J., Hazan, E., and Singer, Y.: Adaptive subgradient methods for online learning and stochastic optimization, *Journal of Machine Learning Research*, Vol. 12, No. Jul, pp. 2121–2159 (2011)

[Hara 15] Hara, K., Saito, D., and Shouno, H.: Analysis of function of rectified linear unit used in deep learning, in *2015 International Joint Conference on Neural Networks (IJCNN)*, pp. 1–8IEEE (2015)

[Ioffe 15] Ioffe, S. and Szegedy, C.: Batch normalization: Accelerating deep network training by reducing internal covariate shift, *arXiv preprint arXiv:1502.03167* (2015)

[Kingma 14] Kingma, D. and Ba, J.: Adam: A method for stochastic optimization, *arXiv preprint arXiv:1412.6980* (2014)

[LeCun 15] LeCun, Y., Bengio, Y., and Hinton, G.: Deep learning, *Nature*, Vol. 521, No. 7553, pp. 436–444 (2015)

[Park 05] Park, H., Inoue, M., and Okada, M.: Slow dynamics due to singularities of hierarchical learning machines, *Progress of Theoretical Physics Supplement*, Vol. 157, pp. 275–279 (2005)

[Riegler 95] Riegler, P. and Biehl, M.: On-line backpropagation in two-layered neural networks, *Journal of Physics A: Mathematical and General*, Vol. 28, No. 20, p. L507 (1995)

[Saad 95] Saad, D. and Solla, S. A.: On-line learning in soft committee machines, *Physical Review E*, Vol. 52, No. 4, p. 4225 (1995)

[Salimans 16] Salimans, T. and Kingma, D. P.: Weight normalization: A simple reparameterization to accelerate training of deep neural networks, in *Advances in Neural Information Processing Systems* (2016)

[Tieleman 12] Tieleman, T. and Hinton, G.: Lecture 6.5-rmsprop: Divide the gradient by a running average of its recent magnitude, *COURSERA: Neural Networks for Machine Learning*, Vol. 4, No. 2 (2012)

[Zeiler 12] Zeiler, M. D.: ADADELTA: an adaptive learning rate method, *arXiv preprint arXiv:1212.5701* (2012)