多層パーセプトロンにおける自然勾配学習

理化学研究所 脳科学総合研究センター　甘利 俊一

[要　旨]

多層パーセプトロンは、多数のパラメータを含む非線形の関数を用いてその入出力関係を表現する。学習とは、実現したい入出力関係が外部より与えられたときに、その例題のみに基づいてパーセプトロンの可変パラメータを調整し、これによって例題の含む“一般的”な仕組みを得てこの関数を近似的に実現することである。特に、オンライン学習は、例題が一つづつ逐次的に与えられたときに、これを記憶することなくパラメータ学習を行なう。

学習アルゴリズムの典型的なものは、誤差関数の確率降下法に基づくもので、誤差逆伝播法の名前で知られている。ところが、これは学習の収束がきわめて遅いところに問題がある。

近年の統計物理学的手法によるオンライン学習の研究は、この収束の遅さがプラトーと呼ばれる対称性に起因する領域に状態が捉わるからであること、さらにプラトーは普遍的に存在することを明らかにした。

ではプラトーから脱出する方策として何がよいであろうか。本研究は、パーセプトロンのパラメータ空間はユークリッド空間ではなくてリーマン空間の構造を持つことを明らかにする。リーマン空間における勾配（グラディエント）は反変ベクトルで表現されるべきであり、これが真の最急方向を与える。この自然のリーマン構造を用いた学習法は、局所的に漸近最適であるのみならず、プラトーに捉われないものであることを示す。

Natural Gradient Eliminates Plateaus in Multilayer Perceptrons Learning

Shun-ichi Amari
RIKEN Brain Science Institute
amari@brain.riken.go.jp

Abstract

The present memo studies the topological and metrical structures of the parameter space of neural networks, in particular multilayer perceptrons. It is important to study such geometrical structures for evaluating learning abilities of various algorithm. We further propose the natural Riemannian gradient learning algorithm which is powerful not only in the asymptotic regime but also in the intermediate regime.

1 Multilayer perceptrons

Let us consider the following model of multilayer perceptrons,

$$y = \sum_{\alpha=1}^{m} v_\alpha \varphi(\boldsymbol{w}_\alpha \cdot \boldsymbol{x}) + \xi, \tag{1}$$

where $\boldsymbol{x}$ is an n-dimensional input subject to the Gaussian distribution $N(O, I)$, I being the $n \times n$ identity matrix; y is an output from a linear output unit; $\boldsymbol{w}_\alpha$ is an n-dimensional connection weight vector from the input to the α-th hidden unit, $\alpha = 1, \cdots, m$, m being the number of hidden units; v_α is the connection weight from the α-th hidden unit to the output unit; and ξ is a random noise subject to $N(0, \sigma^2)$. The function φ is a sigmoidal function, and we use

$$\varphi(u) = \sqrt{\frac{2}{\pi}} \int_o^u \exp\left\{-\frac{z^2}{2}\right\} dz \tag{2}$$

in the present memo in order to obtain an explicit analytical form of the Fisher information matrix. When bias terms exist, the output of the α-th unit is written as

$$\varphi(\boldsymbol{w}_\alpha \cdot \boldsymbol{x} + \boldsymbol{b}_\alpha), \tag{3}$$

but we neglect $\boldsymbol{b}$ in this memo for simplicity's sake. The terms $\boldsymbol{b}_\alpha$ can be analyzed in a similar way without difficulty. When all v_α are fixed to be equal to 1, we have a committee machine,

$$y = \sum \varphi(\boldsymbol{w}_\alpha \cdot \boldsymbol{x}) + \xi, \tag{4}$$

as a special case.

The parameters $\{\boldsymbol{w}_1, \cdots, \boldsymbol{w}_\alpha; \boldsymbol{v}\}$ can be summarized into a single $m(n+1)$-dimensional vector $\boldsymbol{\theta}$, which plays the role of a (local) coordinate system in the space S of the multilayer perceptrons.

A perceptron having parameter θ gives the conditional probability of output y conditioned on input $\boldsymbol{x}$,

$$p(y|\boldsymbol{x},\theta) = \frac{1}{\sqrt{2\pi}\sigma} \exp\left[-\frac{1}{2\sigma^2}\{y - f(\boldsymbol{x},\theta)\}^2\right], \tag{5}$$

$$f(\boldsymbol{x},\theta) = \sum v_\alpha \varphi(\boldsymbol{w}_\alpha \cdot \boldsymbol{x}) \tag{6}$$

is the mean value of y given input $\boldsymbol{x}$. Its logarithm is

$$l(z|\boldsymbol{x},\theta) = -\frac{1}{2\sigma^2}\{y - f(\boldsymbol{x},\theta)\}^2 - \log(\sqrt{2\pi}\sigma). \tag{7}$$

This can be regarded as the negative of the square of an error when y is a target value, except for a scale and a constant term. The space S of perceptrons is identified with the set of all the conditional probability distributions of the form (5).

2 Global Structure of S

When

$$p(y|\boldsymbol{x};\theta_1) = p(y|\boldsymbol{x};\theta_2) \tag{8}$$

holds, two perceptrons with parameters θ_1 and θ_2 have the same input-output relation. We define that two perceptrons (or two points) θ_1 and θ_2 are equivalent, $\theta_1 \approx \theta_2$ in this case. When θ has no equivalent points other than itself, it is called a proper point. Otherwise, it is a singular point. From the behavioral point of view, we need to consider the quotient space $\tilde{S} = S/R$, where R is the equivalence relation just introduced.

In order to study the topological structure of $\tilde{S}$, we study when two points are equivalent. To this end, we introduce the following group acting on S (Chen, Liu and Hecht-Nielsen). Let $\prod$ be a permutation of indices $\{1,2,\cdots,m\}$ which maps i to $\prod(i)$. Let $\prod$ act on $\theta = \{\boldsymbol{w}_1,\cdots,\boldsymbol{w}_m;v_1,\cdots,v_m\}$ as

$$\prod\{\boldsymbol{w}_1,\cdots,\boldsymbol{w}_m;\boldsymbol{v}\} = \left\{\boldsymbol{w}_{\Pi(1)}\cdots,\boldsymbol{w}_{\Pi(m)};v_{\Pi(1)},\cdots v_{\Pi(m)}\right\}. \tag{9}$$

Let $C_\alpha(\alpha = 1,\cdots,m)$ be the change of signs of $\boldsymbol{w}_\alpha$ and v_α:

$$C_\alpha\{\boldsymbol{w}_1,\cdots,\boldsymbol{w}_m;v_1,\cdots,v_m\} = \{\boldsymbol{w}_1,\cdots,-\boldsymbol{w}_\alpha,\cdots,\boldsymbol{w}_m;v_1,\cdots,-v_\alpha,\cdots,v_m\}. \tag{10}$$

Theorem 1 (Chen, Liu and Hecht-Nielsen). Two points are equivalent when they are transformed to each other by an element of the group composed of $Gr = \{\prod, C_\alpha\}$.

The proof is trivial since the form (1) is invariant under $\prod$ and C_α because of $\varphi(-u) = -\varphi(u)$.

Therefore, we need to study the residual set S/Gr. However, when θ is invariant under some elements of Gr, further singular points are found. We have the following three types:

1. When $v_\alpha = 0$, two points are equivalent when they differ only by $\boldsymbol{w}_\alpha$.

2. When $\boldsymbol{w}_\alpha = 0$, two points are equivalent when they differ only by v_α.

3. When $\boldsymbol{w}_\alpha = \pm\boldsymbol{w}_\beta$ holds, two points are equivalent when the values of $v_\alpha \pm v_\beta$ are equal.

Theorem 2 (Sussman). Two points are equivalent when, and only when, they are transformed by an element of Gr or they are connected by one of the above relations.

We call the subspaces defined by

$$\begin{aligned} &1)\quad v_\alpha = 0, \\ &2)\quad \boldsymbol{w}_\alpha = 0, \\ &3)\quad \boldsymbol{w}_\alpha = \pm \boldsymbol{w}_\beta \end{aligned} \tag{11}$$

critical subspaces. A critical subspace is decomposed into a family of lower dimensional subspaces in which all the points are equivalent. For example, $v_\alpha = 0$ for a fixed α defines a hyperplane V_α in S. The set of the points which differ only by $\boldsymbol{w}_\alpha$ is an n-dimenensional subspace in it, and V_α is composed of such n-dimensional subspaces in which all the points are equivalent. Similarly, $\boldsymbol{w}_\alpha = \boldsymbol{w}_\beta$ defines an nm-dimensional subspace in S. It is decomposed into a family of lines defined by $v_\alpha + v_\beta = c$, on which all the points are equivalent.

It should be noted that critical subspaces may intersect. The space of multilayer perceptrons is the residue class by these equivalence relations. It is an $m(n+1)$-dimensional manifold except for the critical submanifolds in which $\tilde{S}$ has lower-dimensional structures. The following simple example shows such a structure: Let S be a Euclidean 3-space with coordinates (x, y, z). Let $x = 0$ be a critical subspace, and define an equivalence relation that $(0, y, z)$ is equivalent to $(0, y', z)$. An equivalence class is a line defined by $x = 0$ and $z = c$, and such lines form the critical surface $x = 0$. In this example, S is divided into two parts by $x = 0$, and the degeneration of dimension reduction takes place on this surface.

3 Structural Degeneration

In the critical subspaces, a multilayer perceptron degenerates into a simpler structure. When $v_\alpha = 0$ or $\boldsymbol{w}_\alpha = 0$ holds, the α-th hidden unit is ineffective and we can remove it, having a perceptron with $m - 1$ hidden units. Similarly, when $\boldsymbol{w}_\alpha = \boldsymbol{w}_\beta$ $(\boldsymbol{w}_\alpha = -\boldsymbol{w}_\beta)$ holds, the outputs of the α-th and β-th elements are always the same (the same with opposite signs). Therefore, the two units can be merged into one unit where $v_\alpha + v_\beta$ $(v_\alpha - v_\beta)$ is the connection weight of the new unit to the output unit.

Further structural degenerations take place at the intersections of critical subspaces.

4 Learning of Multilayer Perceptrons

A multilayer perceptron is trained by examples $\{\boldsymbol{x}_1, y_1^*\}, \cdots, \{\boldsymbol{x}_t, y_t^*\}$, where y_t^* is the output of the teacher when $\boldsymbol{x}_t$ is given. The teacher signal y_t^* is generated by

$$y_t^* = f^*(\boldsymbol{x}_t) + \xi_t. \tag{12}$$

The conventional on-line (or off-line) learning method modifies the current parameter $\boldsymbol{\theta}_t$ by using the gradient of the loss function such that

$$\boldsymbol{\theta}_{t+1} = \boldsymbol{\theta}_t - \eta \frac{\partial l^*(\boldsymbol{x}_t, y_t^*; \boldsymbol{\theta}_t)}{\partial \boldsymbol{\theta}}. \tag{13}$$

Here y_t^* is the teacher output for $\boldsymbol{x}_t$ and

$$l^*(\boldsymbol{x}_t, y_t^*; \boldsymbol{\theta}_t) = \frac{1}{2}\{y_t^* - f(\boldsymbol{x}_t; \boldsymbol{\theta}_t)\}^2. \tag{14}$$

This is the conventional backpropagation method. Such a stochastic descent method for analog multilayer perceptron was proposed by Amari (1967), where he suggested to use a positive definite matrix C to obtain

$$\boldsymbol{\theta}_{t+1} = \boldsymbol{\theta}_t - \eta C \frac{\partial l(\boldsymbol{x}_t, y_t^*; \boldsymbol{\theta}_t)}{\partial \boldsymbol{\theta}}. \tag{15}$$

After the monumental work of Rumelhart et al., this method is called the backpropagation learning algorithm, and a lot of acceleration methods have been studied. They are, for example, the momentum method, conjugate gradient, Newton's method, etc. Adaptive learning rate η has also been studied (Amari, 1967 ; Barkai and Sompolinski, Murata et al).

It is known that the backpropagation method is extremely slow because of the existence of plateaus. A plateau is a saddle point of the learning dynamics, and trajectories of dynamics are first attracted by such plateaus. It takes long learning time to get rid of a plateau. Therefore, a typical learning curve takes the form shown in Fig. 2. Where do plateaus exist in S? We show that plateaus exist in critical subspaces.

Let us consider a critical subspace $W_{\alpha\beta}$ defined by $\boldsymbol{w}_\alpha = \boldsymbol{w}_\beta$ $(\alpha \neq \beta)$. This corresponds to networks of $m-1$ hidden units where the $\alpha-$ and $\beta-$th units are merged. Given a target function $f^*(\boldsymbol{x})$ provided by the teacher, let us consider its optimal approximator in $W_{\alpha\beta}$, that is , the best approximator of $f^*(\boldsymbol{x})$ by a network with $m-1$ hidden units. Let it be $\boldsymbol{\theta}^*$. From

$$\frac{\partial E\left[l(\boldsymbol{x}, \boldsymbol{y}^*; \boldsymbol{\theta}^*)\right]}{\partial \boldsymbol{\theta}^*} = 0, \tag{16}$$

where

$$l^* = \frac{1}{2}\left\{y^* - f(\boldsymbol{x}, \boldsymbol{\theta})\right\}^2. \tag{17}$$

We then have

$$\frac{\partial E[l]}{\partial \boldsymbol{w}_\gamma} = -E\left[\left\{y^* - f(\boldsymbol{x}, \boldsymbol{\theta}^*)\right\}\frac{\partial f(\boldsymbol{x}, \boldsymbol{\theta}^*)}{\partial \boldsymbol{w}_\gamma}\right] = 0. \tag{18}$$

This gives

$$E\left[\left\{y^* - f(\boldsymbol{x}, \boldsymbol{\theta}^*)\right\}\varphi'(\boldsymbol{w}_\gamma^* \cdot \boldsymbol{x})\boldsymbol{x}\right] = 0 \tag{19}$$

for all γ.

Now we show the following theorem.

Theorem 3. The best approximation $\boldsymbol{\theta}^* \in W_{\alpha\beta}$ is a critical point forming a plateau.

Proof. Let us calculate Δl^* corresponding to a change of $\boldsymbol{w}_\alpha = \boldsymbol{w}_\beta$ to $\boldsymbol{w}_\alpha + \boldsymbol{\epsilon}$ and $\boldsymbol{w}_\beta - \boldsymbol{\epsilon}$ for a small vector $\boldsymbol{\epsilon}$ by which $\boldsymbol{\theta}$ escapes from $W_{\alpha\beta}$. It is easily shown that

$$\begin{aligned} \Delta l^* &= E\left[\left\{y^* - f(\boldsymbol{x}, \boldsymbol{\theta}^*)\right\}\varphi'(\boldsymbol{w}_\alpha \cdot \boldsymbol{x})(v_\alpha - v_\beta)\boldsymbol{\epsilon} \cdot \boldsymbol{x}\right] \\ &= 0. \end{aligned} \tag{20}$$

Therefore, the gradient of l^* in directions from $\boldsymbol{\theta}^*$ toward outside $W_{\alpha\beta}$ is 0 as well as those inside $W_{\alpha\beta}$. Hence, $\boldsymbol{\theta}^*$ is a critical point. We next show that there exist trajectories staying in $W_{\alpha\beta}$ and converging to $\boldsymbol{\theta}^*$. To show this, we calculate Δl^* at any point $\boldsymbol{\theta} \in W_{\alpha\beta}$ corresponding to a change of $\boldsymbol{w}_\alpha$ by $v_\beta \boldsymbol{\epsilon}$ and a change of $\boldsymbol{w}_\beta$ $(= \boldsymbol{w}_\alpha)$ by $v_\alpha \boldsymbol{\epsilon}$. The corresponding changes $\Delta \boldsymbol{w}_\alpha$ and $\Delta \boldsymbol{w}_\beta$ by learning are

$$\Delta \boldsymbol{w}_\alpha = -\eta \frac{\partial l^*}{\partial \boldsymbol{w}_\alpha} = -\eta e \varphi'(\boldsymbol{w}_\alpha \cdot \boldsymbol{x}) v_\alpha \boldsymbol{x}, \tag{21}$$

$$\Delta \boldsymbol{w}_\beta = -\eta \frac{\partial l^*}{\partial \boldsymbol{w}_\beta} = -\eta e \varphi'(\boldsymbol{w}_\alpha \cdot \boldsymbol{x}) v_\beta \boldsymbol{x}. \tag{22}$$

This shows that

$$\Delta \boldsymbol{w}_\alpha \cdot (v_\beta \boldsymbol{\epsilon}) - \Delta \boldsymbol{w}_\beta \cdot (v_\alpha \boldsymbol{\epsilon}) = 0 \tag{23}$$

Therefore, at positions where $v_\alpha = v_\beta$ hold, the gradient flow is alway in $W_{\alpha\beta}$. This proves that $\boldsymbol{\theta}^*$ is a plateau.

5 Fisher Information Metric and Riemannian Structure

The Kullback-Leibler divergence

$$\begin{aligned} D[\boldsymbol{\theta} : \boldsymbol{\theta}'] &= E\left[\log \frac{p(y|\boldsymbol{x};\boldsymbol{\theta})}{p(y|\boldsymbol{x};\boldsymbol{\theta}')}\right] \\ &= \int p(y|\boldsymbol{x};\boldsymbol{\theta}) \log \frac{p(y|\boldsymbol{x};\boldsymbol{\theta})}{p(y|\boldsymbol{x};\boldsymbol{\theta}')} p(\boldsymbol{x}) d\boldsymbol{x} dy \end{aligned} \tag{24}$$

is a divergence measure between two points $\boldsymbol{\theta}$ and $\boldsymbol{\theta}'$ in S. This measure is based on the stochastic behaviors of two networks, so that it is equal to 0 when the two networks $\boldsymbol{\theta}$ and $\boldsymbol{\theta}'$ are equivalent.

When $\boldsymbol{\theta}$ and $\boldsymbol{\theta}' = \boldsymbol{\theta} + d\boldsymbol{\theta}$ are infinitesimally close to each other, we have

$$\begin{aligned} D(\boldsymbol{\theta} : \boldsymbol{\theta} + d\boldsymbol{\theta}) &= \frac{1}{2} d\boldsymbol{\theta}^T G(\boldsymbol{\theta}) d\boldsymbol{\theta}) \\ &= \frac{1}{2} \sum g_{ij}(\boldsymbol{\theta}) d\theta_i d\theta_j, \end{aligned} \tag{25}$$

where $G(\boldsymbol{\theta}) = (g_{ij}(\boldsymbol{\theta}))$ is a matrix defined by

$$\begin{aligned} g_{ij}(\boldsymbol{\theta}) &= E\left[\frac{\partial l(y|\boldsymbol{x};\boldsymbol{\theta})}{\partial \theta_i} \frac{\partial l(y|\boldsymbol{x};\boldsymbol{\theta})}{\partial \theta_j}\right] \\ &= -E\left[\frac{\partial^2}{\partial \theta_i \partial \theta_j} l(y|\boldsymbol{x};\boldsymbol{\theta})\right]. \end{aligned} \tag{26}$$

This is the Fisher information matrix.

The Fisher information defines a Riemannian metric in the tangent space $T_{\boldsymbol{\theta}}$ at each $\boldsymbol{\theta}$. Given a vector $d\boldsymbol{\theta}$ which represents a small deviation of $\boldsymbol{\theta}$ at $\boldsymbol{\theta}$, its squared length is defined by

$$< d\boldsymbol{\theta}, d\boldsymbol{\theta} >_{\boldsymbol{\theta}} = d\boldsymbol{\theta}^T G(\boldsymbol{\theta}) d\boldsymbol{\theta}. \tag{27}$$

Two vectors $d\boldsymbol{\theta}$ and $d\boldsymbol{\theta}'$ are said to be orthogonal when

$$< d\boldsymbol{\theta}, d\boldsymbol{\theta}' >_{\boldsymbol{\theta}} = d\boldsymbol{\theta}^T G(\boldsymbol{\theta}) d\boldsymbol{\theta}' = 0. \tag{28}$$

When $G(\boldsymbol{\theta}) = I$, the identity matrix, at any $\boldsymbol{\theta}$, the space reduces to a Euclidean space and $<\ ,\ >$ is the ordinary inner product in the Euclidean space. A space is said to be Riemannian, when $G(\boldsymbol{\theta})$ is defined at each $\boldsymbol{\theta}$. The space of multilayer perceptrons is intrinsically Riemannian, not reducing to a Euclidean one.

The Riemannian metric represents the topological structure of S very well. In a critical subspace, for example $W_{\alpha\beta}$, two points $(\boldsymbol{w}_1, \cdots, \boldsymbol{w}_m; \boldsymbol{v})$ and $(\boldsymbol{w}_1, \cdots, \boldsymbol{w}_m; \boldsymbol{v}')$ are equivalent

when $\boldsymbol{w}_\alpha = \boldsymbol{w}_\beta$ and $v_\alpha + v_\beta = v'_\alpha + v'_\beta$. A small deviation $d\boldsymbol{\theta}$ in the equivalent direction $d\boldsymbol{\theta}$ in this equivalence class is of length 0,

$$d\boldsymbol{\theta}^T G(\boldsymbol{\theta}) d\boldsymbol{\theta} = 0, \tag{29}$$

implying that $G(\boldsymbol{\theta})$ is degenerate in a critical subspace. Therefore, when $\boldsymbol{\theta}$ is close to a critical subspace, it is ill conditioned. Inside a critical subspace, we can define $G(\boldsymbol{\theta})$ in this subspace which corresponds to the space of a reduced architecture of networks. It is regular in the subspace except for critical subspaces inside it.

Theorem 4 (Fukumizu). The Fisher information is regular except for critical subspaces.

6 Natural Gradient Learning Algorithm

The natural gradient learning algorithm (Amari, 1998) updates the current $\boldsymbol{\theta}_t$ by

$$\boldsymbol{\theta}_{t+1} = \boldsymbol{\theta}_t - \eta_t G^{-1}(\boldsymbol{\theta}_t) \frac{\partial l(y_t, \boldsymbol{x}_t; \boldsymbol{\theta}_t)}{\partial \boldsymbol{\theta}} \tag{30}$$

where η_t is a learning constant which may depend on t. The term

$$\tilde{\nabla} l = G^{-1}(\boldsymbol{\theta}_t) \frac{\partial l}{\partial \boldsymbol{\theta}} \tag{31}$$

gives the true steepest direction of function l defined in a Riemannian manifold. The inverse G^{-1} does not exist in critical subspaces. Therefore, one should use the reduced G in the submanifold.

The natural gradient learning method has the following remarkable property.

Theorem 5 (Amari, Opper). When the learning rate is $\eta_t = 1/t$, the estimator $\boldsymbol{\theta}_t$ obtained by the natural gradient learning algorithm is Fisher efficient,

$$E\left[(\boldsymbol{\theta}_t - \boldsymbol{\theta})^T (\boldsymbol{\theta}_t - \boldsymbol{\theta})\right] = \frac{1}{t} G^{-1}(\boldsymbol{\theta}), \tag{32}$$

that is, its asymptotic covariance is the same as the best batch estimator.

A much more important property was suggested by Amari (1998) that the natural gradient learning algorithm eliminates plateaus. We can elucidate how plateaus are eliminated by natural gradient descent learning by using the committee machine.

References

[1] Amari, S. (1998), "Natural gradient works efficiently in learning", *Neural Computation*, 10, 251-276.

[2] Amari, S. (1998), "Natural gradient for over- and under-complete bases in ICA, submitted.