*ARTICLE* **Communicated by Akio Fujiwara**

# Dynamics of Learning in MLP: Natural Gradient and Singularity Revisited

**Shun-ichi Amari**
*amari@brain.riken.jp*
*RIKEN Brain Science Institute, Wako-shi, Saitama 351-0198, Japan*

**Tomoko Ozeki**
*tozeki@tokai.ac.jp*
*Tokai University, Hiratsuka-shi, Kanagawa 259-1292, Japan*

**Ryo Karakida**
*karakida.ryo@aist.go.jp*
*National Institute of Advanced Industrial Science and Technology, Koto-ku, Tokyo 135-0064, Japan*

**Yuki Yoshida**
*yoshida@mns.k.u-tokyo.ac.jp*
**Masato Okada**
*okada@k.u-tokyo.ac.jp*
*University of Tokyo, Kashiwashi, Chiba 277-8561, Japan*

**The dynamics of supervised learning play a main role in deep learning, which takes place in the parameter space of a multilayer perceptron (MLP). We review the history of supervised stochastic gradient learning, focusing on its singular structure and natural gradient. The parameter space includes singular regions in which parameters are not identifiable. One of our results is a full exploration of the dynamical behaviors of stochastic gradient learning in an elementary singular network. The bad news is its pathological nature, in which part of the singular region becomes an attractor and another part a repulser at the same time, forming a Milnor attractor. A learning trajectory is attracted by the attractor region, staying in it for a long time, before it escapes the singular region through the repulser region. This is typical of plateau phenomena in learning. We demonstrate the strange topology of a singular region by introducing blow-down coordinates, which are useful for analyzing the natural gradient dynamics. We confirm that the natural gradient dynamics are free of critical slowdown. The second main result is the good news: the interactions of elementary singular networks eliminate the attractor part and the Milnor-type attractors disappear. This explains why large-scale networks do not suffer from serious critical slowdowns due to singularities. We**

*Neural Computation* **30, 1–33** (2018)
doi:10.1162/NECO_a_01029

**finally show that the unit-wise natural gradient is effective for learning in spite of its low computational cost.**

## 1 Introduction

The multilayer perceptron (MLP) was first proposed by Rosenblatt (1962) as a learning machine consisting of neuron-like binary units. Rosenblatt considered various types of general deep networks with feedback connections, but most studies at that time focused on the simple perceptron, a three-layer feedforward network with input, hidden, and output layers. Its capability of learning was proved by the perceptron convergence theorem, although only output binary neurons are modifiable. It is necessary to introduce analog neurons to train hidden units (Amari, 1967; Werbos, 1974; Rumelhart, Hinton, & Williams, 1986).

Deep learning has achieved astonishing results in various applications in spite of the fact that the network architecture is almost the same as that of the original perceptron of Rosenblatt. It has benefited from a number of remarkable ideas. One is the introduction of analog neurons, which makes the stochastic gradient descent learning method applicable. Another is the convolutional structure, which realizes shift invariance (Fukushima, 1980; LeCun, Bottou, Bengio, & Haffner, 1998). The third is the introduction of restricted Boltzmann machines and autoencoders, which makes it possible to implement self-organization as pre-training. We may add the dropout technique and long- and short-term memory, among many others. However, theoretical foundations of deep learning have not yet been established.

This article includes a review of the dynamics of supervised learning of MLP, paying attention to the singular structure of the parameter space of MLP. Because of singularities, stochastic gradient dynamics are often trapped on plateaus, resulting in very slow convergence. The natural gradient method was proposed to overcome this difficulty (Amari, 1998). (See the detailed studies by Ollivier, 2015a, 2015b.)

We begin by analyzing the behavior of an elementary MLP, consisting of two hidden neurons and one output neuron, which is included in a general MLP as a subnetwork. We recapitulate the analysis of learning dynamics in a neighborhood of a singular region (Wei, Zhang, Cousseau, Ozeki, & Amari, 2008; Cousseau, Ozeki, & Amari, 2008), fortified by new results. When a singular region is not the optimal solution, it may form a strange attractor of the Milnor type, causing retardation in learning. Next, we analyze the behaviors of natural gradient dynamics, showing that the natural gradient eliminates Milnor attractors. We also elucidate the topology of the singular region by introducing "blow-down" coordinates.

How do elementary singular networks behave in a larger network? It is our finding that mutual coupling of singular subnetworks eliminates singular attracting regions, so that singular regions are no longer serious

problems. This implies that the Milnor-type attractor vanishes because of interactions of backpropagated error signals.

Although singular attracting regions disappear, there are lots of saddle points having eigenvalues close to zero, and the vanilla gradient dynamics are slow. The natural gradient is effective, but its computational cost is high. Lots of ideas have been proposed to overcome this deficit, including efficient approximations of natural gradients (see, e.g., Pascanu & Bengio, 2013). We confirm here that the unit-wise natural gradient method (Kurita, 1994; Ollivier, 2015a, 2015b) is effective in spite of its low computational cost.

This article is organized as follows. Section 2 recounts not-so-well-known historical episodes concerning MLP stochastic gradient descent learning and remarks on new trends related to the theoretical foundations of deep learning. Section 3 is a preliminary study devoted to the Fisher information matrix of a single neural unit. Section 4, the main part, elucidates the singular structure of the parameter space of MLP and summarizes the dynamical behaviors near singular regions. The blow-down coordinates are introduced to show the topology of the singular regions and analyze the dynamics of natural gradient learning. Section 5 proves a striking fact that attracting singular regions (Milnor attractors) disappear by connecting elementary singular networks, when the optimal solution is outside the singular regions. This fact confirms that a large network is free of pathological plateau phenomena due to singularities. In section 6, after a survey of various invariant Riemannian metrics (versions of Fisher information), we show that the unit-wise natural gradient works well in singular regions. Section 7 summarizes our conclusions.

## 2 Brief Episodes on MLP Learning

We begin by giving a short recount of historical developments. The original MLP (Rosenblatt, 1962) is a binary machine, where learning takes place only in the output units. One needs to use hidden analog neural units to modify them. A differentiable cost function can naturally be defined, and the stochastic gradient descent method is applicable. These ideas were proposed by Amari (1967) in order to facilitate learning of hidden units in MLP. The dynamical behavior of learning was also analyzed in the vicinity of the optimal solution in this early paper.

The stochastic gradient learning method has a root in the Robbins-Monro stochastic approximation method (Robbins & Monro, 1951). Bryson (1962) and Tsypkin (1966) proposed a similar idea for optimizing nonlinear systems, but it cannot be directly applied to binary MLP. Amari (1967) proposed a stochastic gradient method applicable to MLP. Furthermore, the dynamical behavior of this method was studied near the optimal solution, and a trade-off was found between the speed of convergence and accuracy of learning. The first computer-simulated results of online stochastic

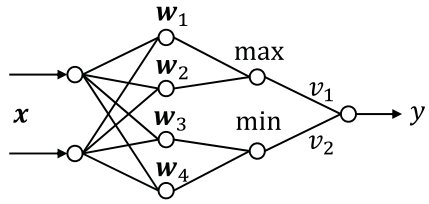


Figure 1: Network of piecewise linear classifier.

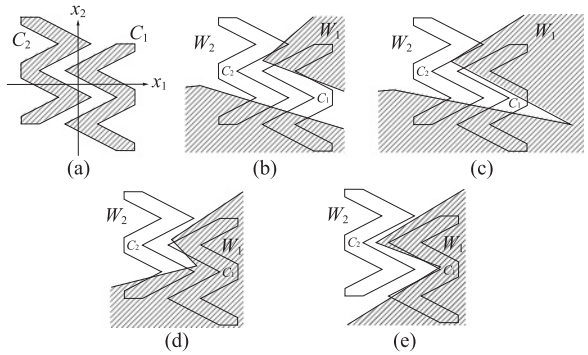


Figure 2: Results of learning. Two categories $C_1$ and $C_2$ of patterns $\boldsymbol{\xi} = (\xi_1, \xi_2)$. $W_1$ is the decision region for $C_1$, having piecewise linear boundary. As learning proceeds, the region changes from initial (panel b) to panels c and d, and, finally, panel e.

gradient descent learning applied to MLP were reported in a Japanese book (Amari, 1968). It is a very simple model of a pattern classifier with two inputs, one output, and six hidden units, arranged in four layers. We show the model in Figure 1 as a historical episode. Its input-output functions is

$$y = v_1 \max\{\boldsymbol{w}_1 \cdot \boldsymbol{x}, \boldsymbol{w}_2 \cdot \boldsymbol{x}\} + v_2 \min\{\boldsymbol{w}_3 \cdot \boldsymbol{x}, \boldsymbol{w}_4 \cdot \boldsymbol{x}\}, \tag{2.1}$$

where $\boldsymbol{x} = (x_1, x_2)$ is the input, $y$ is the output, and $\boldsymbol{w}_1, \ldots, \boldsymbol{w}_4$ are modifiable synaptic weight vectors. $v_1$ and $v_2$ are also modifiable weights of the output neuron, but they are set equal to 1 in this example. The task is to find a classifier that separates two categories $C_1$ and $C_2$. They are not linearly separable, so an MLP was used, which realizes a piecewise linear function, equation 2.1. As is shown in Figure 2, the classifier successfully converges after 25 steps.

The stochastic gradient descent of MLP was followed by similar ideas (e.g., Werbos, 1974). It became very popular after the proposal of the

backpropagation idea and algorithm for calculating the gradient of a function by the PDP group (Rumelhart et al., 1986).

A multilayer convolutional network, neocognitron, was proposed by Fukushima (1980). LeCun et al. (1998) used the convolutional architecture together with stochastic gradient learning, obtaining excellent results in pattern recognition. Although the above two ideas were first proposed in Japan, to our regret, researchers in other countries were the first to integrate the convolutional structure with stochastic gradient learning, obtaining fruitful results.

Hochreiter and Schumdhuber (1997) proposed LSTM (long- and short-term memory) to overcome the difficulty of vanishing gradients. This idea is particularly useful for recurrent neural networks, although Ollivier (2015b) demonstrated that the unit-wise natural gradient method is more efficient for some examples on learning of symbol sequences in recurrent networks.

A new era of deep learning was opened by the groups of G. E. Hinton (see, e.g., Hinton, Osindero, & Teh, 2006; Hinton & Salakhutdinov, 2006) and Bengio (2009), where self-organization is introduced as pretraining before stochastic gradient supervised learning. (See the detailed historical remarks on deep learning by Schmidhuber, 2015.)

Although deep learning opened the door to a new world of neural-network-based AI, many theoretical questions remain unanswered. One important question is how input signals are represented in a deep architecture by self-organization. It is hoped that higher-order features are formed in higher layers, but we do not have convincing theories in this regard. (See Saxe, McClelland, and Ganguli, 2013, for one such approach that uses linear networks.) Poole, Lahiri, Raghu, Sohl-Dickstein, and Ganguli (2016) used a statistical neurodynamics approach involving random networks to show how signals are entangled in deep layers. To understand the problem, we need to use adequate models of probability distributions of signals in the real world in order to show what higher-order features are and how they arise in higher layers.

Deep learning uses a huge number of units, so the underlying parameter space has extremely many dimensions. It has gradually been revealed that high-dimensional models have properties that are remarkably different from those of low-dimensional ones. For instance, Dauphin et al. (2014) used random matrix theory to show that most critical points are saddles in high dimensions. Choromanska, Henaff, Mathieu, Aous, and LeCun (2015) used a spherical spin-glass model to show that the values of most local minima are concentrated around the level of the global minimum. This might suggest that deep learning is free of the local minimum problem.

Singularities have been believed to be a serious problem for supervised learning in the parameter space of a multilayer network. A series of research articles attempted to tackle this problem (Fukumizu & Amari, 2000; Amari, Park, & Fukumizu, 2000; Park, Amari, & Fukumizu, 2000; Amari, Park, & Ozeki, 2006; Wei et al., 2008; Cousseau et al., 2008). A Bayesian theory of

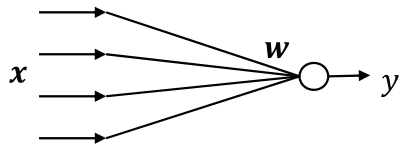


Figure 3: Simple perceptron.

singularities in a learning system has been developed by Watanabe (2009), who used algebraic geometry for this purpose. In our article, we show a new theoretical result as to why singularities are not so serious in a large-scale network.

Recently, the focus of attention has been widened to include symbol processing by using recurrent neural networks. Ollivier (2015b) demonstrated that the unit-wise natural gradient method is very effective for some examples—better than the LSTM method. Neural network parallel dynamics and symbol manipulation had been two different directions of research on AI. But now their unification has begun. Humans are capable of both quick parallel dynamical processes and slow logical reasoning under consciousness. It is intriguing that future neural-networks-based AI can behave as if it has consciousness. (See Oizumi, Tsuchiya, & Amari, 2016, for a new measure of information integration from information-geometric point of view; see also Amari, 2016.)

## 3 Fisher Information Matrix of a Single Analog Neuron

Here, we examine the Fisher information matrix of a single analog neuron as a preliminary. Let us consider a unit that processes an input vector $\boldsymbol{x}$ to give an output $y$,

$$y = \varphi(\boldsymbol{w} \cdot \boldsymbol{x}) + \varepsilon, \tag{3.1}$$

where $\varepsilon$ is gaussian noise subject to $N(0, 1)$ and $\boldsymbol{w}$ is a weight vector (see Figure 3). Here, we may assume that the variance of the noise is $\sigma^2$, without changing the essence of the following analysis. The activation function $\varphi$ is assumed to be the error function,

$$\varphi(u) = \sqrt{\frac{2}{\pi}} \int_0^u \exp\left\{-\frac{z^2}{2}\right\} dz, \tag{3.2}$$

because it is convenient for obtaining explicit analytical formulas. The results obtained in this article hold similarly for other differentiable sigmoidal functions. However, they cannot be applied to the ReLU activation function.

The input vector is

$$\boldsymbol{x} = (x_1, \ldots, x_n, x_0)\,, \tag{3.3}$$

where $x_0 = 1$ and

$$\boldsymbol{w} = (w_1, \ldots, w_n, w_0)\,, \tag{3.4}$$

where $w_0$ is the bias term equal to the negative of the threshold.

The behavior of a neuron is characterized by the joint probability distribution of $\boldsymbol{x}$ and $y$ parameterized by $\boldsymbol{w}$,

$$p(y, \boldsymbol{x}; \boldsymbol{w}) = \frac{1}{\sqrt{2\pi}} q(\boldsymbol{x}) \exp\left[-\frac{1}{2}\left\{y - \varphi(\boldsymbol{w}\cdot\boldsymbol{x})\right\}^2\right], \tag{3.5}$$

where $q(\boldsymbol{x})$ is the probability distribution of the input signals. The log likelihood is given by

$$\log p(y, \boldsymbol{x}; \boldsymbol{w}) = \log q(\boldsymbol{x}) - \frac{1}{2}\left\{y - \varphi(\boldsymbol{w}\cdot\boldsymbol{x})\right\}^2 - \log\sqrt{2\pi}. \tag{3.6}$$

The score function of the probability model, equation 3.6, is written as

$$\nabla_{\boldsymbol{w}} \log p(y, \boldsymbol{x}; \boldsymbol{w}) = \varepsilon \nabla_{\boldsymbol{w}} \varphi(\boldsymbol{w}\cdot\boldsymbol{x}) = \sqrt{\frac{2}{\pi}}\varepsilon \boldsymbol{x} \exp\left\{-\frac{(\boldsymbol{w}\cdot\boldsymbol{x})^2}{2}\right\}, \tag{3.7}$$

where $\nabla_{\boldsymbol{w}}$ is the gradient operator $\partial/\partial w_i$.

The Fisher information is a matrix,

$$\mathbf{G}(\boldsymbol{w}) = \mathrm{E}\left[\left(\nabla_{\boldsymbol{w}} \log p\right)\left(\nabla_{\boldsymbol{w}} \log p\right)^T\right], \tag{3.8}$$

where $T$ denotes transposition of column vectors $\nabla_{\boldsymbol{w}} \log p$ and E is the expectation with respect to $p(y, \boldsymbol{x}; \boldsymbol{w})$. Fisher information plays the role of the Riemannian metric in the parameter space $M = \{\boldsymbol{w}\}$ of a single neuron.

Below, we calculate $\mathbf{G}(\boldsymbol{w})$ for two simple input distributions $q(\boldsymbol{x})$.

**3.1 Gaussian Input Distribution.** When a random variable $\boldsymbol{x}$ is subject to $N(0, \mathbf{V})$, which denotes a zero mean gaussian distribution with covariance matrix $\mathbf{V}$, we have

$$\begin{aligned}\mathbf{G}(\boldsymbol{w}) &= \frac{2}{\pi}\mathrm{E}\left[\boldsymbol{x}\boldsymbol{x}^T \exp\left\{-(\boldsymbol{w}\cdot\boldsymbol{x})^2\right\}\right] \\ &= \frac{2}{\pi}\frac{1}{(\sqrt{2\pi})^n\sqrt{|\mathbf{V}|}}\int \boldsymbol{x}\boldsymbol{x}^T \exp\left\{-\frac{1}{2}\boldsymbol{x}^T(\mathbf{V}^{-1} + 2\boldsymbol{w}\boldsymbol{w}^T)\boldsymbol{x}\right\} d\boldsymbol{x}.\end{aligned} \tag{3.9}$$

Through simple calculations, we get

$$\mathbf{G}(w) = \frac{2}{\pi} \frac{1}{\sqrt{|\mathbf{I} + 2\mathbf{V}\boldsymbol{w}\boldsymbol{w}^T|}} \left(\mathbf{V}^{-1} + 2\boldsymbol{w}\boldsymbol{w}^T\right)^{-1}. \tag{3.10}$$

The inverse of the Fisher information matrix is explicitly obtained as

$$\mathbf{G}^{-1}(\boldsymbol{w}) = \frac{\pi}{2} \sqrt{|\mathbf{I} + 2\mathbf{V}\boldsymbol{w}\boldsymbol{w}^T|} \left(\mathbf{V}^{-1} + 2\boldsymbol{w}\boldsymbol{w}^T\right). \tag{3.11}$$

When $\mathbf{V} = \mathbf{I}$, with $\mathbf{I}$ being the identity matrix, by putting $w^2 = \boldsymbol{w} \cdot \boldsymbol{w}$, we further have

$$\mathbf{G} = \frac{2}{\pi} \frac{1}{\sqrt{1 + 2w^2}} \left(\mathbf{I} - \frac{2}{1 + 2w^2} \boldsymbol{w}\boldsymbol{w}^T\right), \tag{3.12}$$

$$\mathbf{G}^{-1} = \frac{\pi}{2} \sqrt{1 + 2w^2} (\mathbf{I} + 2\boldsymbol{w}\boldsymbol{w}^T). \tag{3.13}$$

In this derivation, we have omitted the component $x_0 = 1$, which is not subject to $N(0, 1)$. It is easy to calculate $\mathbf{G}$ in this case too.

**3.2 Degenerate Input Distribution.** Here, we consider the case in which the input signals $\boldsymbol{x}$ are limited to be within a subspace. Let $N$ be an orthogonal subspace such that for $\boldsymbol{u} \in N$, $\boldsymbol{u} \cdot \boldsymbol{x}$ is always equal to 0. Then, from equation 3.9, we have

$$\boldsymbol{u}^T \mathbf{G}(\boldsymbol{w}) \boldsymbol{u} = 0. \tag{3.14}$$

This implies that $\mathbf{G}$ is singular.

More specifically, when $x_1 = x_2$ is always satisfied, we have

$$g_{11} = g_{12} = g_{21} = g_{22}. \tag{3.15}$$

Obviously, the null direction is

$$\boldsymbol{u} = (1, -1, 0, \ldots, 0). \tag{3.16}$$

In this case, the two weight vectors $\boldsymbol{w}$ and $\boldsymbol{w} + c\boldsymbol{u}$ are equivalent, giving the same output because of $\boldsymbol{u} \cdot \boldsymbol{x} = 0$. This occurs in an MLP, when two neurons 1 and 2 in the same layer have the same weight vectors, $\boldsymbol{w}_1 = \boldsymbol{w}_2$. Their outputs are the same, so this causes the Fisher matrix to be singular. This is the origin of the singularity in MLP.

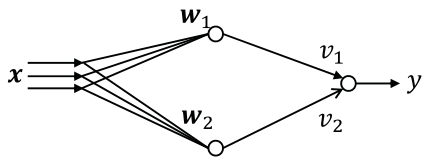


Figure 4: Elementary singular network ($n-2-1$).

Now let us focus on online learning of a neuron. Let the true parameter be $\boldsymbol{w}^*$, from which the teacher output $y^*$ is generated. When the current value of the parameter is $\boldsymbol{w}$, the error is

$$e^* = y^* - \varphi(\boldsymbol{w} \cdot \boldsymbol{x}), \tag{3.17}$$

and its square gives the instantaneous loss function

$$l(\boldsymbol{x}, y^*; \boldsymbol{w}) = \frac{1}{2}\left\{y^* - \varphi(\boldsymbol{w} \cdot \boldsymbol{x})\right\}^2. \tag{3.18}$$

The vanilla online gradient learning algorithm modifies the current $\boldsymbol{w}$ into $\boldsymbol{w} + \Delta\boldsymbol{w}$ by receiving $\boldsymbol{x}$ and $y^*$,

$$\Delta\boldsymbol{w} = -\eta \nabla_{\boldsymbol{w}} l\left(\boldsymbol{x}, y^*, \boldsymbol{w}\right), \tag{3.19}$$

and the natural gradient algorithm uses $\mathbf{G}^{-1}\nabla_{\boldsymbol{w}} l$,

$$\tilde{\Delta}\boldsymbol{w} = -\eta \tilde{\nabla}_{\boldsymbol{w}} l = -\eta \mathbf{G}^{-1} \nabla_{\boldsymbol{w}} l\left(\boldsymbol{x}, y^*, \boldsymbol{w}\right), \tag{3.20}$$

where $\eta$ is a learning constant. The natural gradient method is Fisher efficient for estimating $\boldsymbol{w}^*$ (Amari, 1998).

## 4 Elementary Singular Networks

Now let us consider a very simple three-layer $n$-2-1 MLP consisting of $n$ inputs, two hidden neurons, and one linear output, which is called an elementary singular network (see Figure 4). Its input-output behavior can be expressed as

$$y = f(\boldsymbol{x}, \boldsymbol{\theta}) + \varepsilon, \tag{4.1}$$

$$f(\boldsymbol{x}, \boldsymbol{\theta}) = v_1 \varphi\left(\boldsymbol{w}_1 \cdot \boldsymbol{x}\right) + v_2 \varphi\left(\boldsymbol{w}_2 \cdot \boldsymbol{x}\right), \tag{4.2}$$

where the parameters are

$$\boldsymbol{\theta} = \left(\boldsymbol{w}_1, v_1, \boldsymbol{w}_2, v_2\right). \tag{4.3}$$

They form a parameter manifold $M$ of elementary singular networks. We assume that input $\boldsymbol{x}$ is subject to $N(0, \mathbf{I})$. Such elementary networks are included in a general MLP as subnetworks. The manifold $M = \{\boldsymbol{\theta}\}$ includes singular regions so that the learning dynamics show strange behaviors in the neighborhood of such a region. It is useful to study the dynamical behaviors of both vanilla and natural gradient learning in detail by using such a simple model. Note that $n$-$k$-1 networks ($k \geq 2$) share the same pathological behavior. The dynamics of vanilla gradient descent learning were studied in detail by Wei et al. (2008), so we recapitulate their results briefly here.

The behavior of natural gradient learning was studied by Cousseau et al. (2008) for the case of when the true network is included in a singular region and $n = 1$. We extend their results to the case of when the true network is outside the singular regions, proving that the natural gradient method is free of pathological behaviors caused by singularities. Then we finally examine the strange topology of the singularities in the behavior space.

**4.1 Singular Region and Related Coordinate Transformation.** In an elementary MLP, a network is nonidentifiable when $\boldsymbol{w}_1 = \boldsymbol{w}_2 = \boldsymbol{w}$, because equation 4.1 reduces to

$$y = (v_1 + v_2)\,\varphi(\boldsymbol{w} \cdot \boldsymbol{x}) + \varepsilon, \tag{4.4}$$

whose behavior is the same for any $v_1$ and $v_2$ so long as $v_1 + v_2$ is the same. Thus, we cannot identify $v_1$ and $v_2$ themselves except for their sum. A similar situation holds true when $\boldsymbol{w}_1 = -\boldsymbol{w}_2$ because $\varphi$ is an odd function. Further, when $v_1 = 0$, $\boldsymbol{w}_1$ is not identifiable because $v_1\varphi\,(\boldsymbol{w}_1 \cdot \boldsymbol{x})$ vanishes. When $v_2 = 0$, the situation is similar.

We call such a set of unidentifiable parameters a singular region of $M$. For any pair $(\boldsymbol{w}, v)$ of parameters, there exists a singular region specified by them:

$$\begin{aligned} R(\boldsymbol{w}, v) = &\{\boldsymbol{w}_1 = \pm\boldsymbol{w}_2 = \boldsymbol{w}, v_1 \pm v_2 = v\}, \\ &\cup \{v_1 = 0, v_2 = v, \boldsymbol{w}_2 = \boldsymbol{w}\}, \\ &\cup \{v_2 = 0, v_1 = v, \boldsymbol{w}_1 = \boldsymbol{w}\}. \end{aligned} \tag{4.5}$$

Inside each region $R(\boldsymbol{w}, v) \subset M$, all the networks share the same input-output behavior:

$$y = v\varphi(\boldsymbol{w} \cdot \boldsymbol{x}) + \varepsilon. \tag{4.6}$$

One hidden neuron is enough to realize this behavior. Note that the singular region $R(\boldsymbol{w}, v)$ is a union of four submanifolds shown in equation 4.5. $R(\boldsymbol{w}, v)$ moves continuously as $\boldsymbol{w}$ and $v$ change, so the union of the singular regions is a continuum in the parameter space. For simplicity, we study

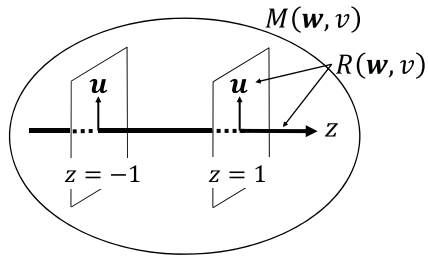


Figure 5: Singular regions $R(\boldsymbol{w}, v) \subset M(\boldsymbol{w}, v)$ in the $\boldsymbol{\xi}$-coordinates.

only the case of $\boldsymbol{w}_1 = \boldsymbol{w}_2$; the case of $\boldsymbol{w}_1 = -\boldsymbol{w}_2$ case can be studied in a very similar way.

In order to analyze the dynamics of learning near a singular region, we introduce a new (local) coordinate system, $\boldsymbol{\xi} = (\boldsymbol{w}, v, \boldsymbol{u}, z)$ (see Wei et al., 2008):

$$\boldsymbol{w} = \frac{v_1 \boldsymbol{w}_1 + v_2 \boldsymbol{w}_2}{v_1 + v_2}, \tag{4.7}$$

$$v = v_1 + v_2, \tag{4.8}$$

$$\boldsymbol{u} = \boldsymbol{w}_2 - \boldsymbol{w}_1, \tag{4.9}$$

$$z = \frac{v_2 - v_1}{v_1 + v_2}. \tag{4.10}$$

Its inverse transformation is given by

$$\boldsymbol{w}_1 = \boldsymbol{w} - \frac{1}{2}(1 + z)\boldsymbol{u}, \tag{4.11}$$

$$\boldsymbol{w}_2 = \boldsymbol{w} + \frac{1}{2}(1 - z)\boldsymbol{u}, \tag{4.12}$$

$$v_1 = \frac{1}{2}(1 - z)v, \tag{4.13}$$

$$v_2 = \frac{1}{2}(1 + z)v. \tag{4.14}$$

The singular region $R(\boldsymbol{w}, v)$ is represented in this coordinate system as

$$R(\boldsymbol{w}, v) = \left\{\boldsymbol{u} = 0 \,;\, z \text{ arbitrary}\right\} \cup \left\{z = \pm 1 \,;\, \boldsymbol{u} \text{ arbitrary}\right\}. \tag{4.15}$$

$R(\boldsymbol{w}, v)$ consists of three submanifolds: one is one-dimensional, specified by $\boldsymbol{u} = 0$, $z$ arbitrary, called a singular line; the other two are $n$-dimensional submanifolds corresponding to $z = \pm 1$, called escaping walls (see Figure 5).

Let $M(\boldsymbol{w}, v)$ be a submanifold of $M$ in which $v$ and $\boldsymbol{w}$ are fixed but $z$ and $\boldsymbol{u}$ take arbitrary values. Then $R(\boldsymbol{w}, v) \subset M(\boldsymbol{w}, v)$.

Following Wei et al. (2008), we consider the dynamics of learning near singularities. We expand $f(\boldsymbol{x}, \boldsymbol{\theta})$ in equation 4.2 in a Taylor series of $\boldsymbol{u}$ by using the coordinates $\boldsymbol{\xi}$:

$$f(\boldsymbol{x}, \boldsymbol{\xi}) = v\varphi(\boldsymbol{w} \cdot \boldsymbol{x}) + \frac{v}{8}\left(1 - z^2\right)\varphi''(\boldsymbol{w} \cdot \boldsymbol{x})(\boldsymbol{u} \cdot \boldsymbol{x})^2 - \frac{v}{24}z(1 - z^2)\varphi'''(\boldsymbol{w} \cdot \boldsymbol{x})(\boldsymbol{u} \cdot \boldsymbol{x})^3 + O\left(|\boldsymbol{u}|^4\right). \tag{4.16}$$

The error function at $\boldsymbol{\xi}$ is

$$e^*(\boldsymbol{x}, y^*, \boldsymbol{\xi}) = y^* - f(\boldsymbol{x}, \boldsymbol{\xi}) = f(\boldsymbol{x}, \boldsymbol{\xi}^*) - f(\boldsymbol{x}, \boldsymbol{\xi}) + \varepsilon, \tag{4.17}$$

where $y^*$ is the teacher signal generated from the true network specified by parameter $\boldsymbol{\xi}^*$. The instantaneous loss function is

$$l(\boldsymbol{x}, y^*, \boldsymbol{\xi}) = \frac{1}{2}\left|e^*(\boldsymbol{x}, y^*, \boldsymbol{\xi})\right|^2. \tag{4.18}$$

The averaged stochastic gradient descent vanilla online learning equation is

$$\dot{\boldsymbol{\theta}} = -\eta\langle\nabla_{\boldsymbol{\theta}} l\rangle, \tag{4.19}$$

where we use continuous time $t$, $\dot{\boldsymbol{\theta}} = d\boldsymbol{\theta}/dt$ and the average $\langle \quad \rangle$ is taken over $\boldsymbol{x}$ and $\varepsilon$. Its solution $\boldsymbol{\theta}(t)$ gives the average trajectory of learning around which the actual trajectory fluctuates. Now we change the coordinate system from $\boldsymbol{\theta}$ to $\boldsymbol{\xi}$. From

$$\dot{\boldsymbol{\xi}} = \frac{\partial \boldsymbol{\xi}}{\partial \boldsymbol{\theta}}\dot{\boldsymbol{\theta}} \tag{4.20}$$

and by using the Jacobian of the coordinate transformation,

$$\mathbf{J} = \frac{\partial \boldsymbol{\xi}}{\partial \boldsymbol{\theta}}, \quad \nabla_{\boldsymbol{\xi}} l = (\mathbf{J}^T)^{-1}\nabla_{\boldsymbol{\theta}} l, \tag{4.21}$$

we have

$$\dot{\boldsymbol{\xi}} = -\eta\mathbf{J}\mathbf{J}^T\langle\nabla_{\boldsymbol{\xi}} l\rangle. \tag{4.22}$$

Through careful calculations, the averaged learning equation can be expressed in the new coordinate system as

$$\dot{\boldsymbol{w}} = \frac{\eta}{2}\upsilon\left(1+z^2\right)\langle\varphi' e^*\boldsymbol{x}\rangle + \eta\left(\frac{1}{4}\upsilon z(1-z^2)\langle\varphi'' e^*\boldsymbol{x}(\boldsymbol{u}\cdot\boldsymbol{x})\rangle - \frac{z}{\upsilon}\boldsymbol{u}\langle\varphi e^*\rangle\right) + O(u^2), \tag{4.23}$$

$$\dot{\upsilon} = 2\eta\langle\varphi e^*\rangle - \eta z\langle\varphi' e^*(\boldsymbol{u}\cdot\boldsymbol{x})\rangle + O(u^2), \tag{4.24}$$

$$\dot{\boldsymbol{u}} = \eta\left\{\upsilon z\langle\varphi' e^*\boldsymbol{x}\rangle + \frac{\upsilon}{2}\left(1-z^2\right)\langle\varphi'' e^*\boldsymbol{x}(\boldsymbol{u}\cdot\boldsymbol{x})\rangle\right\} + O(u^2), \tag{4.25}$$

$$\dot{z} = -\eta\left\{\frac{2z}{\upsilon}\langle\varphi e^*\rangle - \frac{1+z^2}{\upsilon}\langle\varphi' e^*(\boldsymbol{u}\cdot\boldsymbol{x})\rangle + \frac{z\left(z^2+3\right)}{4\upsilon}\langle\varphi'' e^*(\boldsymbol{u}\cdot\boldsymbol{x})^2\rangle\right\} + O(u^3), \tag{4.26}$$

where $\varphi = \varphi(\boldsymbol{w}\cdot\boldsymbol{x})$, $e^* = e^*(\boldsymbol{x}, y^*, \boldsymbol{\xi})$, and $O(u)$ denotes the higher-order terms of $u$.

The dynamics (see equations 4.23 to 4.26) are split into two parts. One is the part of $(\boldsymbol{u}, z)$ in which the change is very slow near singularity $\boldsymbol{u} \approx 0$, as will be shown later. The other part is the dynamics of $(\boldsymbol{w}, \upsilon)$, which defines fast dynamics. We can solve the fast dynamics, equations 4.23 and 4.24, by putting $\boldsymbol{u} = 0$. When the dynamics converge to an equilibrium, let the stable equilibrium be $(\tilde{\boldsymbol{w}}^*, \tilde{\upsilon}^*)$. This is the best approximation of the teacher function $f(\boldsymbol{x}, \boldsymbol{\xi}^*)$ by using a single hidden unit because of $\boldsymbol{u} = 0$. The equilibrium solution satisfies

$$\langle e^*\varphi'(\tilde{\boldsymbol{w}}^*\cdot\boldsymbol{x})\boldsymbol{x}\rangle = 0, \tag{4.27}$$

$$\langle e^*\varphi(\tilde{\boldsymbol{w}}^*\cdot\boldsymbol{x})\rangle = 0. \tag{4.28}$$

By substituting them into (equations 4.25 and 4.26), the dynamics of $\boldsymbol{u}$ and $z$ are shown to be very slow when $\boldsymbol{u} \approx 0$.

Exactly speaking, the dynamical behaviors of equations 4.23 to 4.26 should be analyzed by using the center manifold theory. The center manifold $C$ is defined by

$$\boldsymbol{w} = h(\boldsymbol{u}, z), \tag{4.29}$$

$$\upsilon = k(\boldsymbol{u}, z), \tag{4.30}$$

which are solutions given by putting the right-hand sides of equations 4.23 and 4.24 equal to 0. The slow variables $(\boldsymbol{w}, \upsilon)$ are slaved by the fast

variables $(\boldsymbol{u}, z)$, and we can analyze the dynamics in the center manifold $C$. However, the exact analysis is complicated. Since our purpose is to show qualitative aspects of the dynamics near singularities, in particular to understand qualitative aspects of natural gradient dynamics, we use the submanifold $M(\tilde{\boldsymbol{w}}^*, \tilde{\upsilon}^*)$ specified by $\boldsymbol{w} = \tilde{\boldsymbol{w}}^*$ and $\upsilon = \tilde{\upsilon}^*$ instead of $C$ and analyze the dynamics projected to $M(\tilde{\boldsymbol{w}}^*, \tilde{\upsilon}^*)$. This is approximation used in Wei et al. (2008), supported by computer simulations.

When the submanifold $M(\tilde{\boldsymbol{w}}^*, \tilde{\upsilon}^*)$ is attracting, the fast dynamics cause $\boldsymbol{\xi}$ to fall quickly into $M(\tilde{\boldsymbol{w}}^*, \tilde{\upsilon}^*)$, but $\boldsymbol{u}$ and $z$ change slowly in $M(\tilde{\boldsymbol{w}}^*, \tilde{\upsilon}^*)$, which includes the critical region $R(\tilde{\boldsymbol{w}}^*, \tilde{\upsilon}^*)$. Let us analyze the dynamics of learning in a neighborhood of $R(\tilde{\boldsymbol{w}}^*, \tilde{\upsilon}^*)$ in $M(\tilde{\boldsymbol{w}}^*, \tilde{\upsilon}^*)$, assuming that the state has already fallen into $M(\tilde{\boldsymbol{w}}^*, \tilde{\upsilon}^*)$.

By putting

$$\mathbf{H} = \langle e^* \varphi'' (\tilde{\boldsymbol{w}}^* \cdot \boldsymbol{x}) \boldsymbol{x}\boldsymbol{x}^T \rangle, \tag{4.31}$$

the slow dynamics inside $M(\tilde{\boldsymbol{w}}^*, \tilde{\upsilon}^*)$ can be expressed using equations 4.25 and 4.26 as

$$\dot{\boldsymbol{u}} = \frac{\eta}{2} \tilde{\upsilon}^* \left(1 - z^2\right) \mathbf{H}\boldsymbol{u}, \tag{4.32}$$

$$\dot{z} = -\frac{\eta}{4\tilde{\upsilon}^*} z \left(z^2 + 3\right) \boldsymbol{u}^T \mathbf{H}\boldsymbol{u}, \tag{4.33}$$

because equations 4.27 and 4.28 hold. Integrating these equations explicitly gives the trajectories

$$|\boldsymbol{u}(t)|^2 = \frac{4}{3} \tilde{\upsilon}^{*2} \log \frac{\{z(t)^2 + 3\}^2}{|z(t)|} + c, \tag{4.34}$$

where $c$ represents constants corresponding to specific trajectories (see Figure 6).

We consider the following two cases separately.

*4.1.1 The True Solution $\boldsymbol{\xi}^*$ Lies in $R(\boldsymbol{w}^*, \upsilon^*)$.* The true solution is $\boldsymbol{w} = \boldsymbol{w}^*$, $\upsilon = \upsilon^*$, $\boldsymbol{u} = 0$, so the parameters are redundant. The dynamics converge to the optimal solution, which is any point in $R(\boldsymbol{w}^*, \upsilon^*)$ (see Figure 7). In a neighborhood of the true solution, $e^*$ is of order $|\boldsymbol{u}|^2$ except for the noise term $\varepsilon$, as can be seen from equations 4.16 and 4.17. Hence, $\mathbf{H} = O\left(|\boldsymbol{u}|^2\right)$ and the learning equations, 4.32 and 4.33, are of order

$$\dot{\boldsymbol{u}} = O\left(|\boldsymbol{u}|^3\right), \tag{4.35}$$

$$\dot{z} = O\left(|\boldsymbol{u}|^4\right). \tag{4.36}$$

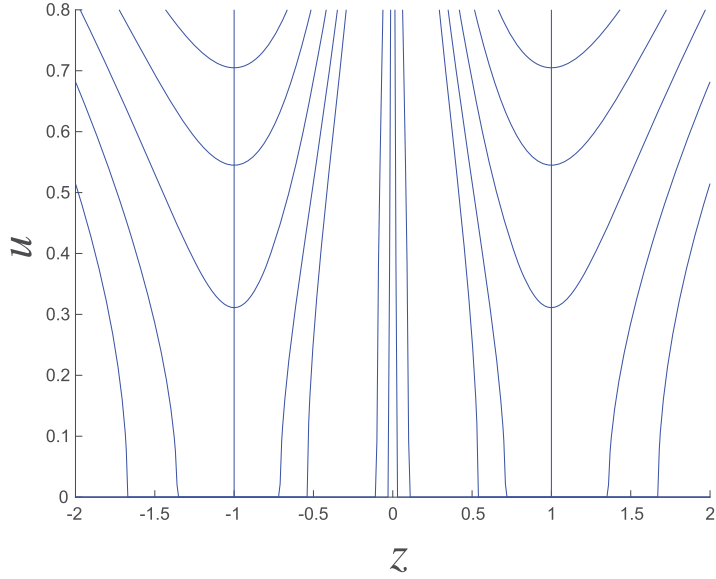


Figure 6: Trajectories of learning in $R(\boldsymbol{w}, v)$.

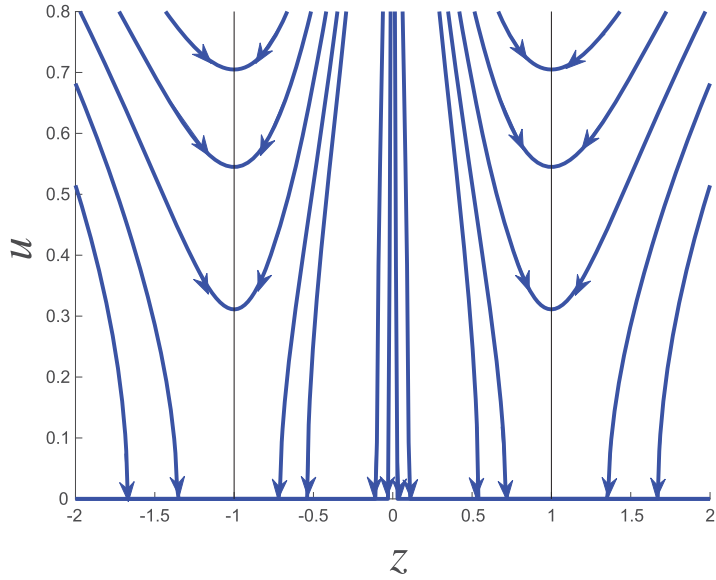


Figure 7: When $\xi^* \in R(\boldsymbol{w}, v)$: Any point in $R(\boldsymbol{w}, v)$ is optimal.

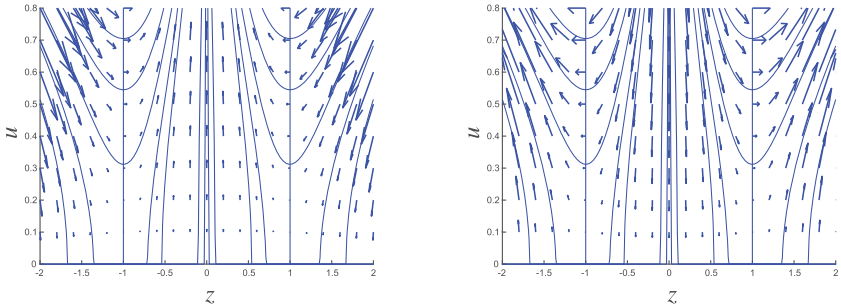


Figure 8: When $\xi^* \notin R(\boldsymbol{w}, \upsilon)$ (left) $\tilde{\upsilon}^* H > 0$ (right) $\tilde{\upsilon}^* H < 0$: Milnor attractor.

Let us define an error function by

$$E = \frac{1}{2}\mathrm{E}\left[e^{*2}\right] = \frac{1}{2}\left\langle \left| f(\boldsymbol{x}, \boldsymbol{\xi}^*) - f(\boldsymbol{x}, \boldsymbol{\xi}) \right|^2 \right\rangle. \tag{4.37}$$

Then we have

$$\dot{E} = O\left(|\boldsymbol{u}|^6\right). \tag{4.38}$$

Therefore, the convergence to $\boldsymbol{u} = 0$ is extremely slow.

*4.1.2 $R(\tilde{\boldsymbol{w}}^*, \tilde{\upsilon}^*)$ Is Not an Optimal Solution: Milnor Attractor.* A trajectory may once enter $M(\tilde{\boldsymbol{w}}^*, \tilde{\upsilon}^*)$ in this case; even $M(\tilde{\boldsymbol{w}}^*, \tilde{\upsilon})$ is not optimal. When $R(\tilde{\boldsymbol{w}}^*, \tilde{\upsilon}^*)$ has an attracting region, a trajectory starting from its basin of attraction enters it. But it is not optimal, so it eventually escapes from $R(\tilde{\boldsymbol{w}}^*, \tilde{\upsilon}^*)$ after reaching the repulsive part of $R(\tilde{\boldsymbol{w}}^*, \tilde{\upsilon}^*)$ (see Figure 8). As can be seen from equation 4.32, when $\mathbf{H}$ has both positive and negative eigenvalues, the equilibrium $\boldsymbol{u} = 0$ is a saddle and there is no attracting region in $R(\tilde{\boldsymbol{w}}^*, \tilde{\upsilon}^*)$. In this case, $\boldsymbol{\xi}$ escapes from $M(\tilde{\boldsymbol{w}}^*, \tilde{\upsilon}^*)$ without entering $R(\tilde{\boldsymbol{w}}^*, \tilde{\upsilon}^*)$.

However, $\mathbf{H}$ is positive-definite in many cases. In particular, when $n = 1$, $\mathbf{H}$ is a scalar, and no saddle part appears. We will show that the singular line definitely has an attracting region. To this end, we remark that $(\tilde{\boldsymbol{w}}^*, \tilde{\upsilon}^*)$ is the minimizer of $\langle l(\boldsymbol{x}, \boldsymbol{w}, \upsilon)\rangle$ in the singular region $\boldsymbol{u} = 0$, so the Hessian,

$$\mathbf{H}^* = \begin{bmatrix} \left\langle \dfrac{\partial^2 l}{\partial \boldsymbol{w} \partial \boldsymbol{w}} \right\rangle & \left\langle \dfrac{\partial^2 l}{\partial \boldsymbol{w} \partial \upsilon} \right\rangle \\ \left\langle \dfrac{\partial^2 l}{\partial \upsilon \partial \boldsymbol{w}} \right\rangle & \left\langle \dfrac{\partial^2 l}{\partial \upsilon \partial \upsilon} \right\rangle \end{bmatrix}, \tag{4.39}$$

is positive-definite.

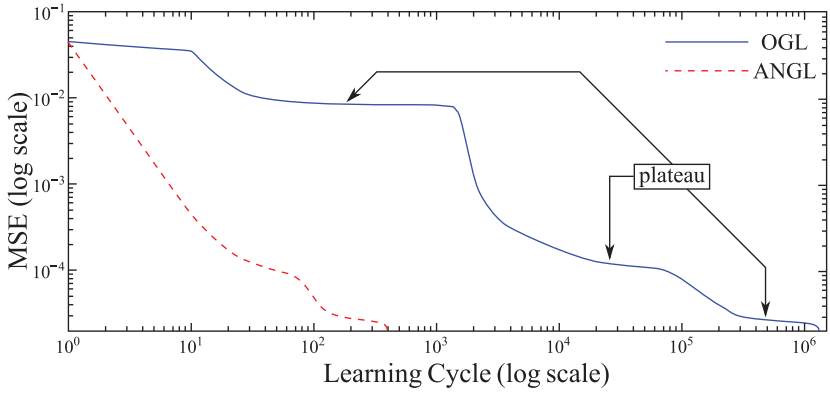


Figure 9: Plateaus in learning curve where the natural gradient has no plateaus (from Park et al., 2000). Solid line: vanilla gradient; broken line: adaptive natural gradient.

It can be rewritten as

$$\mathbf{H}^* = \begin{bmatrix} -\tilde{v}^*\mathbf{H} & 0 \\ 0 & 0 \end{bmatrix} + \left\langle \begin{bmatrix} \tilde{v}^*\varphi'\boldsymbol{x} \\ \varphi \end{bmatrix} \begin{bmatrix} \tilde{v}^*\varphi'\boldsymbol{x} & \varphi \end{bmatrix} \right\rangle. \tag{4.40}$$

Therefore, $-v^*\mathbf{H}$ is positive-definite in many cases. In such a case, the part $|z| < 1$ of the singular line is stable and attracting, as can be seen from equation 4.32, whereas the part $|z| > 1$ is unstable and repulsive. A state starting from the basin of attraction in $M(\tilde{\boldsymbol{w}}^*, \tilde{v}^*)$ once enters the stable part $|z| < 1$ of $R(\tilde{\boldsymbol{w}}^*, v^*)$. The convergence speed is slow: $\dot{E} = O(|\boldsymbol{u}|^2)$. Once the state enters the attracting region of the singular line, the average dynamics, equation 4.33, follow $\dot{z} = 0$ and $z$ does not change. However, the actual stochastic dynamics fluctuate, with $z$ changing randomly, so it reaches the unstable region $|z| > 1$ and escapes. However, the fluctuation dynamics of $z$ are very slow when $\boldsymbol{u}$ is small.

As shown in Figure 8, $R(\tilde{\boldsymbol{w}}^*, v^*)$ consists of four parts. The region $|z| < 1$ of the singular line $|\boldsymbol{u}| = 0$ is an attracting region having a basin of attraction of finite measure. The other part $|z| > 1$ is repulsive. The two walls $z = 1$ and $z = -1$ are escaping walls, through which a trajectory can escape from the $|z| < 1$ part to the $|z| > 1$ part.

In general, $R(\tilde{\boldsymbol{w}}^*, \tilde{v}^*)$ is not necessarily a saddle of the dynamics because the basin of attraction has a finite measure. But part of $R(\tilde{\boldsymbol{w}}^*, \tilde{v}^*)$ is repulsive, so $R(\tilde{\boldsymbol{w}}^*, \tilde{v}^*)$ has attractive and repulsive flows at the same time. Such a strange attractor is known as the Milnor attractor. Figure 9 is a typical example of plateau phenomena in learning.

**4.2 Topology of Singular Region.** The singular region $\mathcal{R}$ of MLP is the union of all $R(\boldsymbol{w}, v)$,

$$\mathcal{R} = \underset{v,\boldsymbol{w}}{\cup} R(\boldsymbol{w}, v), \tag{4.41}$$

in the $\boldsymbol{\xi}$-coordinates. All the points in an $R(\boldsymbol{w}, v)$ are equivalent, having the same input-output function (see equation 4.6). We introduce an equivalence relation $\approx$ such that $\boldsymbol{\xi} \approx \boldsymbol{\xi}'$ when $f(\boldsymbol{x}, \boldsymbol{\xi}) = f(x, \boldsymbol{\xi}')$. The quotient space

$$\mathcal{F} = M/\approx \tag{4.42}$$

consists of all different functions realized by MLP. A singular region $R(\boldsymbol{w}, v)$ reduces to a single point in $\mathcal{F}$. However, $\mathcal{F}$ is not a manifold in the strict mathematical sense, as it includes singular points. It is interesting to see the topology $\mathcal{F}$.

The singular region $R(\boldsymbol{w}, v) \subset M(\boldsymbol{w}, v)$ is a union of two $n$-dimensional submanifolds specified by $|z| = \pm 1$ and a singular line $\boldsymbol{u} = 0$ (see Figure 5). We use the polar coordinates $(u, \boldsymbol{e})$,

$$u = \sqrt{\boldsymbol{u} \cdot \boldsymbol{u}}, \quad \boldsymbol{e} = \frac{\boldsymbol{u}}{u}, \tag{4.43}$$

by separating the magnitude of $\boldsymbol{u}$ and its direction $\boldsymbol{e}$. Here, $\boldsymbol{e}$ belongs to an $(n-1)$-dimensional sphere $S_{n-1}$, satisfying

$$\boldsymbol{e} \cdot \boldsymbol{e} = 1. \tag{4.44}$$

Note that $\boldsymbol{u} = 0$ is a singular point at which $S_{n-1}$ shrinks to a single point $(0, \boldsymbol{e})$, as is the case of the polar coordinates, representing the same point $\boldsymbol{u} = 0$ for any $\boldsymbol{e}$.

We further introduce new parameters $(\tau, \sigma)$ instead of $(u, z)$:

$$\tau = \left(1 - z^2\right) u^2, \tag{4.45}$$

$$\sigma = z\left(1 - z^2\right) u^3. \tag{4.46}$$

Note that $\tau = O\left(u^2\right)$ and $\sigma = O\left(u^3\right)$. The new coordinates $\boldsymbol{\beta} = (\tau, \sigma, \boldsymbol{e})$ introduced in $M(\boldsymbol{w}, v)$ occupy a subset of $\mathbb{R} \times \mathbb{R} \times S_{n-1}$, as will be shown in the following. We study the topology of $\mathcal{F}$ by using $\boldsymbol{\beta}$. The singular region $R(\boldsymbol{v}, \boldsymbol{w})$ is mapped to $\boldsymbol{\beta} = (0, 0, \boldsymbol{e})$, which denotes a single point irrespective of $\boldsymbol{e}$. We call $\boldsymbol{\beta}$ the blow-down coordinates, since it maps a singular region $R(\boldsymbol{w}, v)$ to one point in $\mathcal{F}$, called the origin of $\boldsymbol{\beta}$.

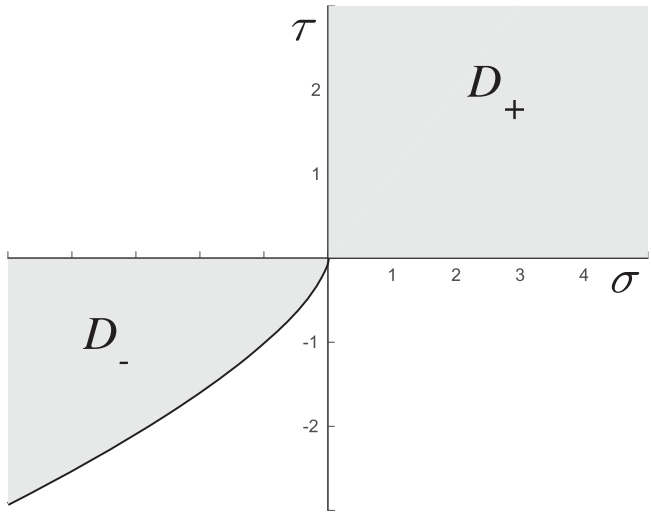


Figure 10: Image of singular region $R(\boldsymbol{w}, \upsilon)$.

The inverse transformation is given by

$$z = \frac{|\sigma|}{\sqrt{\tau^3 + \sigma^2}}, \tag{4.47}$$

$$u = \mathrm{sgn}(\sigma)\frac{\sqrt{\tau^3 + \sigma^2}}{\tau}, \tag{4.48}$$

$$\boldsymbol{u} = u\boldsymbol{e}, \tag{4.49}$$

for $\boldsymbol{\beta} \neq 0$. When $\boldsymbol{\beta} = 0$, the inverse is not unique, giving the entire $R(\boldsymbol{w}, \upsilon)$.

Now let us study the range of $\mathcal{F}$ in the new coordinates $\boldsymbol{\beta} = \{(\tau, \sigma, \boldsymbol{e})\}$. Two points $(z, \boldsymbol{u})$ and $(-z, -\boldsymbol{u})$ are equivalent, giving the same function. Hence, we may consider only the region, $z \geq 0$, which implies $\tau\sigma \geq 0$. Since $(\tau, \sigma)$ satisfies

$$\tau^3 + \sigma^2 = \left(1 - z^2\right)^2 u^6 \geq 0, \tag{4.50}$$

$\mathcal{F}$ covers the outside of the cusp line

$$\tau^3 + \sigma^2 = 0 \tag{4.51}$$

(see Figure 10). When $\tau = 0$, the $\sigma$-axis shrinks to a point $\sigma = 0$.

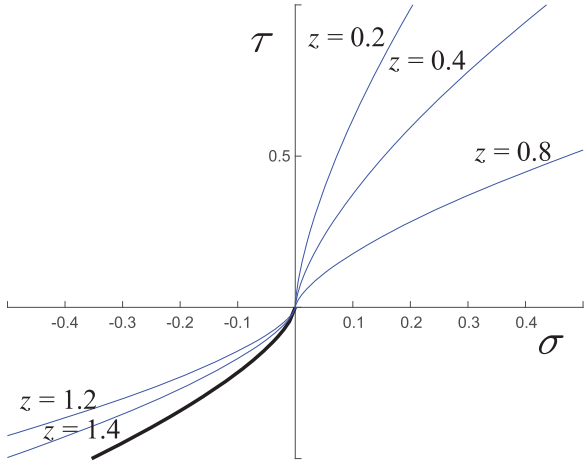


Figure 11: Images of $z = \text{const.}$

The region we consider is composed of two leaves,

$$D_+ = \left\{(\tau, \sigma) | \tau > 0\right\}, \tag{4.52}$$

$$D_- = \left\{(\tau, \sigma) | \tau < 0\right\}, \tag{4.53}$$

which are connected at $\tau = \sigma = 0$ (see Figure 10). The range of $\mathcal{F}$ is $\left\{D_+, D_-, S_{n-1}(0,0)\right\}$. We may draw curves $z = \text{const.}$ on the leaf $D_+$. It is a cusp line,

$$\tau^3 = \frac{1 - z^2}{z^2}\sigma^2, \tag{4.54}$$

when $z = 0$, $\sigma = 0$, and the curve coincides with the $\tau$-axis. As $z$ increases from 0 to 1, the curve changes (see Figure 11) in $D_+$ and when $z = 1$, it shrinks to the origin $\tau = \sigma = 0$. As $z$ increases from 1, the curve moves to $D_-$.

We draw the curves corresponding to $u = \text{const}$, which is

$$\sigma^2 = -\tau^3 + u^2\tau^2, \tag{4.55}$$

in Figure 12.

The $\boldsymbol{\beta}$ coordinates consist of the direct product of the $(\tau, \sigma)$-plane and $S_{n-1} = \{\boldsymbol{e}\}$. However, at the $\tau$-axis on which $z = 0$, $\boldsymbol{e}$ and $-\boldsymbol{e}$ are equivalent. Therefore, $S_{n-1}$ reduces to a projective manifold $P_{n-1}$ on the $\tau$-axis.

When $n = 1$, $S_0 = \{1, -1\}$, consisting of two points. Hence, $\mathcal{F}$ consists of two planes corresponding to $(\tau, \sigma, \boldsymbol{e} = 1)$ and $(\tau, \sigma, \boldsymbol{e} = -1)$, but they are

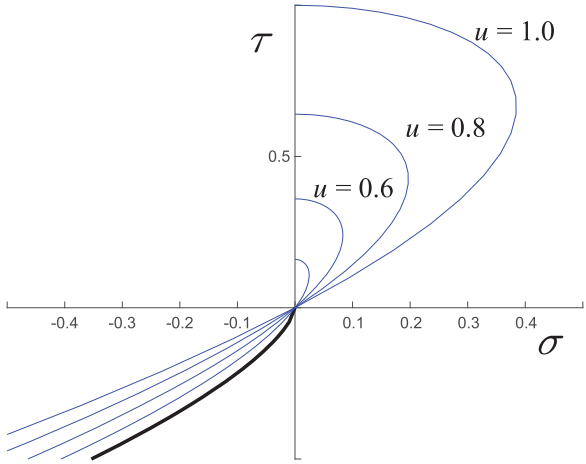


Figure 12: Images of $u = \text{const.}$

connected at $\sigma = 0$. When $n = 2$, $S_1$ is a circle, and it deforms to $P_1$ as $\sigma$ approaches 0. The circle deforms in a double circle, and the corresponding points are pasted to form $P_1$ at $\sigma = 0$.

**4.3 Natural Gradient Learning.** Now let us analyze the dynamics of learning in the blow-down coordinates. The input-output function, equation 4.16, is written as

$$f(\boldsymbol{x}, \boldsymbol{\beta}) = v\varphi(\boldsymbol{w} \cdot \boldsymbol{x}) + \frac{v}{8}\tau(\boldsymbol{e} \cdot \boldsymbol{x})^2\varphi'' - \frac{v}{24}\sigma(\boldsymbol{e} \cdot \boldsymbol{x})^3\varphi''' \tag{4.56}$$

by neglecting higher-order terms. We put

$$a(\boldsymbol{x}, \boldsymbol{e}) = \frac{v}{8}(\boldsymbol{e} \cdot \boldsymbol{x})^2\varphi'', \tag{4.57}$$

$$b(\boldsymbol{x}, \boldsymbol{e}) = -\frac{v}{24}(\boldsymbol{e} \cdot \boldsymbol{x})^3\varphi'''. \tag{4.58}$$

Then we have

$$f(\boldsymbol{x}, \boldsymbol{\beta}) = v\varphi(\boldsymbol{w} \cdot \boldsymbol{x}) + \tau a + \sigma b, \tag{4.59}$$

neglecting higher-order terms. The Fisher information $\mathbf{G}(\boldsymbol{\beta})$ is calculated from

$$\nabla_{\boldsymbol{\beta}} f = \left[a(\boldsymbol{x}), b(\boldsymbol{x}), \nabla_e f\right] \tag{4.60}$$

and

$$\mathbf{G} = \langle (\nabla_\beta f)^T \nabla_\beta f \rangle = \langle [a, b, \nabla_e f]^T [a, b, \nabla_e f] \rangle, \tag{4.61}$$

which is regular except for one point $\boldsymbol{\beta} = 0$. Note that $a$ and $b$ are of order 1, but $\nabla_e f$ is of order $u^2$.

The loss function is

$$l(\boldsymbol{x}, y^*, \boldsymbol{\beta}) = \frac{1}{2} \left\{ f(\boldsymbol{x}, \boldsymbol{\beta}^*) - f(\boldsymbol{x}, \boldsymbol{\beta}) + \varepsilon \right\}^2, \tag{4.62}$$

and hence its gradient is

$$\nabla_\beta l = -e^* \nabla_\beta f = -\left\{ \bar{f}^*(\boldsymbol{x}) - (\tau a + \sigma b) + \varepsilon \right\} \nabla_\beta f, \tag{4.63}$$

where

$$\bar{f}^*(\boldsymbol{x}) = f(\boldsymbol{x}, \boldsymbol{\beta}^*) - v\varphi(\boldsymbol{w} \cdot \boldsymbol{x}). \tag{4.64}$$

In order to invert $\mathbf{G}$, we use the relation

$$\mathbf{G} \begin{bmatrix} \tau \\ \sigma \\ 0 \end{bmatrix} = \left\langle (\tau a + \sigma b)(\nabla_\beta f)^T \right\rangle. \tag{4.65}$$

Then,

$$\mathbf{G}^{-1} \left\langle (\tau a + \sigma b)(\nabla_\beta f)^T \right\rangle = [\tau, \sigma, 0]^T. \tag{4.66}$$

Using this, we have the natural gradient,

$$-\mathbf{G}^{-1} \langle \nabla_{\boldsymbol{\beta}} l \rangle^T = \mathbf{G}^{-1} \left\langle \bar{f}^*(\boldsymbol{x}) \nabla_\beta f \right\rangle^T - [\tau, \sigma, 0]^T. \tag{4.67}$$

When the true model is included in $R(\boldsymbol{w}^*, v^*)$, $\bar{f}^*(\boldsymbol{x}) = 0$. Therefore, the dynamics of learning are

$$\dot{\boldsymbol{\beta}} = -\eta \mathbf{G}^{-1} \langle \nabla_{\boldsymbol{\beta}} l \rangle^T = -\eta [\tau, \sigma, 0]^T + \text{higher-order terms}. \tag{4.68}$$

Disregarding $\boldsymbol{e}$-components, we have the following simple dynamics concerning $\tau$ and $\sigma$,

$$\dot{\tau} = -\eta\tau, \tag{4.69}$$

$$\dot{\sigma} = -\eta\sigma, \tag{4.70}$$

except for higher-order terms,

$$\tau = c_1 \exp\{-\eta t\}, \tag{4.71}$$

$$\sigma = c_2 \exp\{-\eta t\}, \tag{4.72}$$

without any slowdown. Cousseau et al. (2008) obtained this result for $n = 1$. When using the blow-down coordinates, the result holds for any $n$.

It is more interesting to study the learning behavior when the singular region $\tau = \sigma = 0$ is not the optimal solution. Natural gradient dynamics eliminate the Milnor attractor, from which vanilla gradient dynamics suffer seriously. Next, we show that the trajectory passes through the singular point $\tau = \sigma = 0$ with a constant speed as if there were no singularity. In the case of natural gradient learning, we have

$$e^*(\boldsymbol{x}) = \bar{f}^*(\boldsymbol{x}) - (\tau a + \sigma b) + \varepsilon. \tag{4.73}$$

Hence, the main part of the averaged gradient is

$$\left\langle \nabla_{\boldsymbol{\beta}} l \right\rangle = -\left\langle \bar{f}^*(\boldsymbol{x}) \nabla_\beta f \right\rangle. \tag{4.74}$$

We need to evaluate the $(\tau, \sigma)$-part of $\mathbf{G}^{-1}\nabla_\beta f$. Since we have

$$\nabla_e f = \frac{v\tau}{4}(\boldsymbol{e}\cdot\boldsymbol{x})\boldsymbol{x}\varphi'' + O(u^3), \tag{4.75}$$

we rewrite $\mathbf{G}$ as

$$\mathbf{G} = \begin{bmatrix} 1 & & \\ & 1 & \\ & & \tau \end{bmatrix} \mathbf{K} \begin{bmatrix} 1 & & \\ & 1 & \\ & & \tau \end{bmatrix}, \tag{4.76}$$

where $\mathbf{K}$ is

$$\mathbf{K} = \left\langle \left[a, b, \frac{v}{4}(\boldsymbol{e}\cdot\boldsymbol{x})\boldsymbol{x}\varphi''\right]^T \left[a, b, \frac{v}{4}(\boldsymbol{e}\cdot\boldsymbol{x})\boldsymbol{x}\varphi''\right] \right\rangle. \tag{4.77}$$

We assume that $\mathbf{K}$ is nonsingular. Then we have

$$\mathbf{G}^{-1}\left(\nabla_\beta f\right)^T = \begin{bmatrix} O(1) \\ O\left(u^{-2}\right) \end{bmatrix}. \tag{4.78}$$

This proves that the velocities $\dot{\tau}$ and $\dot{\sigma}$ are of order 1 in the natural gradient dynamics. Therefore, the origin is no longer a critical point and the trajectory passes through it with a finite speed.

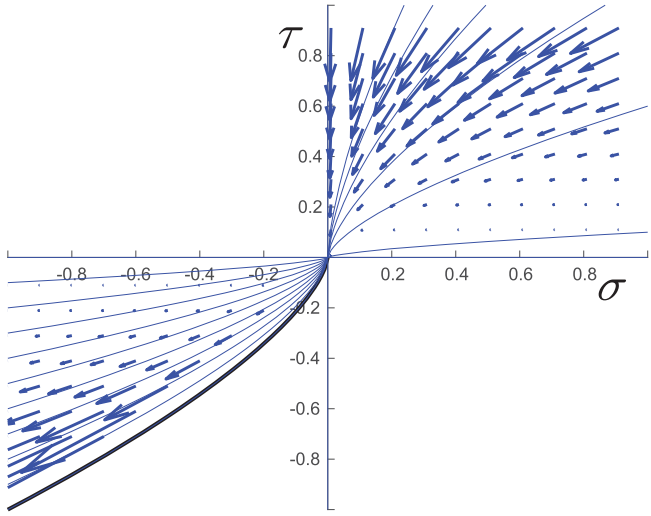


Figure 13: Milnor attractor ($v^*H < 0$).

Now let us examine how the error decreases near the origin. When $\langle e^* \rangle = 0$, we have

$$\dot{E}(t) = c \exp\{-\eta t\}, \tag{4.79}$$

showing exponential convergence to the origin. When $\langle e^* \rangle \neq 0$, $\dot{E}$ is of order 1, so the trajectory passes through the origin with a constant speed (no slowdown).

In the case of the vanilla gradient, the origin $\tau = \sigma = 0$ is a Milnor attractor having a finite measure of the basin of attraction in many cases. The state stays at the origin for long time, as is seen in Figure 13.

## 5 Connecting Elementary Singular Networks: Eliminating Attracting Singular Regions

We have studied the dynamical behavior of an isolated elementary singular network. In a general MLP, elementary networks are connected hierarchically. However, an ($n$-$m$-1) network shows the same pathological behavior as an ($n$-2-1) network (Wei et al., 2008). To our surprise, we were able to prove that although singular regions exist, pathological singular behaviors of MLP are suppressed by connecting output neurons. Let us consider an ($n$-2-$m$) network having $m$ outputs $y_1, \ldots, y_m$,

$$y_i = v_{i1}\varphi\left(\boldsymbol{w}_1 \cdot \boldsymbol{x}\right) + v_{i2}\varphi\left(\boldsymbol{w}_2 \cdot \boldsymbol{x}\right) + \varepsilon, \quad i = 1, \ldots, m \tag{5.1}$$

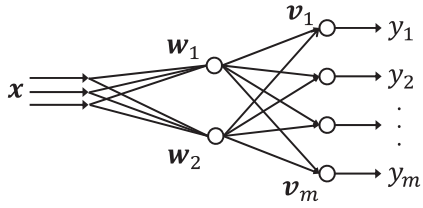


Figure 14: Connecting elementary networks.

(see Figure 14). Here, $\boldsymbol{v}_i = (v_{i1}, v_{i2})$ are the weights of the output neuron $i$. When $\boldsymbol{w}_1 = \boldsymbol{w}_2$, the network is singular and $v_{i2} - v_{i1}$ is not identifiable.

We introduce the $\boldsymbol{\xi}$-coordinate system,

$$\boldsymbol{w} = \alpha \boldsymbol{w}_1 + \bar{\alpha} \boldsymbol{w}_2, \quad (\alpha + \bar{\alpha} = 1), \tag{5.2}$$

$$\boldsymbol{u} = \boldsymbol{w}_2 - \boldsymbol{w}_1, \tag{5.3}$$

$$v_i = v_{i1} + v_{i2}, \tag{5.4}$$

$$z_i = \frac{v_{i2} - v_{i1}}{v_i}, \tag{5.5}$$

where the coefficients $\alpha$ and $\bar{\alpha}$ are functions of $v_i$ and $z_i$ to be determined later.

The inverse transformation is

$$\boldsymbol{w}_1 = \boldsymbol{w} - \bar{\alpha} \boldsymbol{u}, \tag{5.6}$$

$$\boldsymbol{w}_2 = \boldsymbol{w} + \alpha \boldsymbol{u}, \tag{5.7}$$

$$v_{i1} = \frac{1}{2} v_i (1 - z_i), \tag{5.8}$$

$$v_{i2} = \frac{1}{2} v_i (1 + z_i). \tag{5.9}$$

We expand the output function of the $i$th output in terms of $\boldsymbol{\xi}$-coordinates,

$$\begin{aligned} f_i(\boldsymbol{x}, \boldsymbol{\xi}) &= v_{i1} \varphi \left\{ (\boldsymbol{w} - \bar{\alpha} \boldsymbol{u}) \cdot \boldsymbol{x} \right\} + v_{i2} \varphi \left\{ (\boldsymbol{w} + \alpha \boldsymbol{u}) \cdot \boldsymbol{x} \right\} \\ &= v_i \varphi (\boldsymbol{w} \cdot \boldsymbol{x}) + A_i \varphi' (\boldsymbol{w} \cdot \boldsymbol{x}) \boldsymbol{u} \cdot \boldsymbol{x} \\ &\quad + \frac{1}{2} B_i \varphi'' (\boldsymbol{w} \cdot \boldsymbol{x}) (\boldsymbol{u} \cdot \boldsymbol{x})^2, \end{aligned} \tag{5.10}$$

where $\boldsymbol{A} = (A_i)$ and $\boldsymbol{B} = (B_i)$ are

$$A_i = \alpha v_{i2} - \bar{\alpha} v_{i1}, \tag{5.11}$$

$$B_i = \bar{\alpha}^2 v_{i1} + \alpha^2 v_{i2}. \tag{5.12}$$

Let $(\boldsymbol{w}^*, \boldsymbol{v}^*)$ be the best approximation of the vector output $\boldsymbol{f}(\boldsymbol{x}, \boldsymbol{\xi})$ by using a single hidden unit, corresponding to the case of $\boldsymbol{w}^* = \boldsymbol{w}_1 = \boldsymbol{w}_2$. In the case of a single hidden unit, the output function is

$$\boldsymbol{f} = \boldsymbol{v}\varphi(\boldsymbol{w} \cdot \boldsymbol{x}), \tag{5.13}$$

so the best approximation $(\boldsymbol{w}^*, \boldsymbol{v}^*)$ is obtained by setting the derivatives of the average of $l(\boldsymbol{x}, \boldsymbol{y}^*, \boldsymbol{\xi})$ equal to 0,

$$\boldsymbol{v}^* \cdot \langle \boldsymbol{e}^* \varphi'(\boldsymbol{w}^* \cdot \boldsymbol{x})\boldsymbol{x} \rangle = 0, \tag{5.14}$$

$$\langle \boldsymbol{e}^* \varphi(\boldsymbol{w}^* \cdot \boldsymbol{x}) \rangle = 0, \tag{5.15}$$

where $\boldsymbol{e}^*$ is the error vector corresponding to $m$ outputs (see equation 4.17).

The dynamics of $\boldsymbol{w}$ and $\boldsymbol{v}$ are not necessarily faster than those of $\boldsymbol{u}$, but we will assume that they have converged to $(\boldsymbol{w}^*, \boldsymbol{v}^*)$ and analyze the dynamics taking place in the submanifold $M(\boldsymbol{w}^*, \boldsymbol{v}^*)$.

The dynamics in $M(\boldsymbol{w}^*, \boldsymbol{v}^*)$ can be written by using

$$\langle \nabla_u l \rangle = -\left\{ \boldsymbol{A} \cdot \langle \boldsymbol{e}^* \varphi' \boldsymbol{x} \rangle + \boldsymbol{B} \cdot \langle \boldsymbol{e}^* (\boldsymbol{u} \cdot \boldsymbol{x}) \varphi'' \boldsymbol{x} \rangle \right\}, \tag{5.16}$$

$$\langle \nabla_z l \rangle = -\left\{ \nabla_z \boldsymbol{A} \cdot \langle \boldsymbol{e}^* (\boldsymbol{u} \cdot \boldsymbol{x}) \varphi' \rangle \right\}. \tag{5.17}$$

Comparing them with equations 4.32 and 4.33 in the case of $m = 1$, we see that $\boldsymbol{u} = 0$ is a critical point of the dynamics when $\boldsymbol{A} \cdot \langle \boldsymbol{e}^* \varphi' \boldsymbol{x} \rangle$ vanishes. When $\boldsymbol{u} = 0$ is not critical, even though $\boldsymbol{z}$ changes very slowly in a neighborhood of $R(\boldsymbol{w}^*, \boldsymbol{v}^*)$, a trajectory passes through $R(\boldsymbol{w}^*, \boldsymbol{v}^*)$, and does not stay in it. The Milnor-type pathological behavior occurs when $\boldsymbol{u} = 0$ is a critical point and $R(\boldsymbol{w}^*, v^*)$ has an attracting region. Below, we search for a condition for $\boldsymbol{u} = 0$ to be critical.

When $m = 1$, by choosing $\alpha$ and $\bar{\alpha}$ as is shown in equation 4.7, they satisfy

$$\alpha v_{12} = \bar{\alpha} v_{11}, \tag{5.18}$$

and $A = 0$ is derived. In the case of $m \geq 2$, when $\alpha$ and $\bar{\alpha}$ satisfy

$$\alpha v_{i2} - \bar{\alpha} v_{i1} = k v_i \tag{5.19}$$

for constant $k$,

$$\boldsymbol{A} \propto \boldsymbol{v}^* \tag{5.20}$$

holds. Then the first term of equation 5.16 vanishes because of equation 5.14. However, equation 5.19 is not solvable when $m \geq 2$, because there are $m$

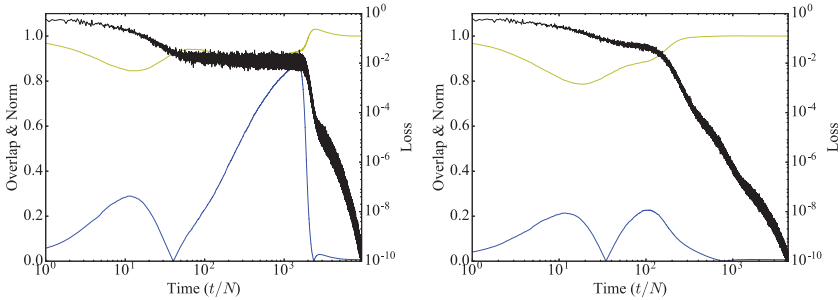


Figure 15: Plateaus still exist. Typical time course of loss (black line) during learning, with overlap of connections of first layer $|\boldsymbol{w}_1 \cdot \boldsymbol{w}_2|/\|\boldsymbol{w}_1\| \|\boldsymbol{w}_2\|$ (blue line), and minimum norm of connections of second layer $\min\{\sqrt{\sum_{i=1}^m v_{i1}^2}, \sqrt{\sum_{i=1}^m v_{i2}^2}\}$ (yellow line). The $(n-2-m)$ network ($n = 1000$, $m = 1$ (left) and $m = 2$ (right)) is trained to predict output of another $(n-2-m)$ network, with fixed weights. The first layers of two networks are initialized so that each element of their weight matrix follows $N(0, 1/\sqrt{n})$, and the weights of second layers of two networks are initialized as (left) $(v_{ij}) = [[1, 1]]$ and (right) $(v_{ij}) = [[1, 1/\sqrt{2}], [0, 1/\sqrt{2}]]$. In the left panel, the overlap takes high value during first $\sim 2000$ steps, which means the approach to the singular region $\boldsymbol{u} = 0$, accompanied by the emergence of plateau. In the right panel, the plateau is alleviated a little, and the increase of the overlap (i.e., the approach to the singular region) is not observed. In both cases, the minimum norm of the second layer's connections does not shrink to 0, and therefore the network does not get close to the other singular region $\{z_i\} = \pm 1$.

equations for one variable $\alpha$, since $\bar{\alpha} = 1 - \alpha$. Therefore, no attracting region appears in $R(\boldsymbol{w}^*, \boldsymbol{v}^*)$, and the trajectory passes through it when $m \geq 2$.

Here, we assumed that $\langle \boldsymbol{e}^* \varphi' \boldsymbol{x} \rangle$ is of full rank in general and its kernel is one-dimensional at the equilibrium $(\boldsymbol{w}^*, \boldsymbol{v}^*)$. Our discussions do not hold when $m$ vectors $\boldsymbol{w}_1, \ldots, \boldsymbol{w}_m$ have the same direction. However, $m$ output neurons are essentially identical in this case, meaning that there is only one output neuron.

We have shown that the stable region disappears when $m \geq 2$ and the trajectory of $\boldsymbol{u}$ passes through it with a finite velocity. In the case that the true solution lies in $R(\boldsymbol{w}^*, v^*)$, the error $\langle \boldsymbol{e}^* \rangle$ is a linear order of $|\boldsymbol{u}|$. Therefore, the trajectory converges to $\boldsymbol{u} = 0$ in the linear order. Hence, any retardation of learning due to a singularity disappears for $m \geq 2$. However, as is shown in Figure 15, plateaus nevertheless exist. They are saddle points including very small absolute eigenvalues, which are ubiquitous in a high-dimensional case (Dauphin et al., 2014). The natural gradient method eliminates the plateaus.

## 6 Natural Gradient Learning of MLP

Here, we briefly recapitulate various invariant Riemannian metrics defined by error backpropagation as described by Ollivier (2015a). Let $\boldsymbol{y}_r$ be the output vector of the $r$th layer, where $\boldsymbol{y}_0 = \boldsymbol{x}$ is the input vector and $\boldsymbol{y} = \boldsymbol{y}_L$ is the final output. Let $\mathbf{W}_r$ be the connection matrix from the $(r-1)$st-layer to the $r$th layer. Then we have a recursive expression of the feedforward behavior,

$$\boldsymbol{y}_r = \varphi\left(\mathbf{W}_r \boldsymbol{y}_{r-1}\right), \quad r = 1, \ldots, L-1, \tag{6.1}$$

where $\varphi$ operates componentwise; that is, $\varphi(\boldsymbol{u})$ is a vector whose components are $\varphi(u_i)$, and the last layer $L$ is linear,

$$\boldsymbol{y}_L = \mathbf{W}_L \boldsymbol{y}_{L-1} + \boldsymbol{\varepsilon}, \tag{6.2}$$

with gaussian noise $\boldsymbol{\varepsilon}$. We can summarize the input-output behavior as

$$\boldsymbol{y} = \boldsymbol{f}(\boldsymbol{x}, \mathbf{W}) + \boldsymbol{\varepsilon}, \tag{6.3}$$

where the parameters are

$$\mathbf{W} = (\mathbf{W}_1, \ldots, \mathbf{W}_L). \tag{6.4}$$

Assume that the true network has parameters

$$\mathbf{W}^* = (\mathbf{W}_1^*, \ldots, \mathbf{W}_L^*). \tag{6.5}$$

Given an input $\boldsymbol{x}$, the accompanying teacher signal $\boldsymbol{y}^*$ is given by

$$\boldsymbol{y}^* = \boldsymbol{f}(\boldsymbol{x}, \mathbf{W}^*) + \boldsymbol{\varepsilon}. \tag{6.6}$$

For the current network with parameters $\mathbf{W}$, the error vector is

$$\boldsymbol{e}^* = \boldsymbol{y}^* - \boldsymbol{f}(\boldsymbol{x}, \mathbf{W}). \tag{6.7}$$

Define the instantaneous loss function by

$$l(\boldsymbol{x}, \boldsymbol{y}^*, \mathbf{W}) = \frac{1}{2}|\boldsymbol{e}^*|^2 = \frac{1}{2}\left|\boldsymbol{f}(\boldsymbol{x}, \mathbf{W}) - \boldsymbol{f}(\boldsymbol{x}, \mathbf{W}^*) - \boldsymbol{\varepsilon}\right|^2. \tag{6.8}$$

The stochastic gradient learning method uses the gradient of $l$ with respect to each $\mathbf{W}_r$. It is recursively calculated as

$$\frac{\partial l}{\partial \mathbf{W}_r} = \boldsymbol{e}_r^* \boldsymbol{y}_{r-1}, \tag{6.9}$$

where $\partial l/\partial \mathbf{W}_r$ is the gradient of $l$ with respect to $\mathbf{W}_r$, with the backpropagated error

$$\boldsymbol{e}_r^* = \boldsymbol{e}_{r+1}^* \mathbf{W}_{r+1} \varphi' \left( \mathbf{W}_r \boldsymbol{y}_{r-1} \right), \quad r = 1, \ldots, L-1, \tag{6.10}$$

$$\boldsymbol{e}_L = -\boldsymbol{e}^* \tag{6.11}$$

Here, $\boldsymbol{ey}$ and $\boldsymbol{e}\varphi'(\mathbf{W}\boldsymbol{y})$ are each a tensor product whose component expressions are, for example, $e_i y_j$ and $e_i \varphi' \left( \sum_k W_{jk} y_k \right)$. The vanilla stochastic descent method updates the current $\mathbf{W}$ by

$$\Delta \mathbf{W}_r = -\eta \frac{\partial l}{\partial \mathbf{W}_r} = -\eta \boldsymbol{e}_r^* \boldsymbol{y}_{r-1}, \quad r = 1, \ldots, L \tag{6.12}$$

whenever the input $\boldsymbol{x}$ and teacher signal $\boldsymbol{y}^*$ are given. Equation 6.12 is a stochastic difference equation since $\boldsymbol{x}$ is a random vector. In order to analyze its behavior, we use the continuous time $t$ and average with respect to $(\boldsymbol{y}^*, \boldsymbol{x})$. Accordingly, we get an averaged differential equation:

$$\dot{\mathbf{W}}_r = -\eta \langle \boldsymbol{e}_r^* \boldsymbol{y}_{r-1} \rangle. \tag{6.13}$$

Equation 6.3 defines the joint probability of $(\boldsymbol{x}, \boldsymbol{y})$ parameterized by $\mathbf{W}$,

$$p(\boldsymbol{x}, \boldsymbol{y}; \mathbf{W}) = \frac{q(\boldsymbol{x})}{(\sqrt{2\pi})^m} \exp \left\{ -\frac{1}{2} |\boldsymbol{y} - \boldsymbol{f}(\boldsymbol{x}, \mathbf{W})|^2 \right\}, \tag{6.14}$$

where $q(\boldsymbol{x})$ is the probability density function of $\boldsymbol{x}$. The probability structure defines a Riemannian metric $\mathbf{G}$ in the parameter space $M$ of $\mathbf{W}$, given by the Fisher information,

$$\mathbf{G} = \mathrm{E} \left[ \frac{\partial l}{\partial \mathbf{W}} \frac{\partial l}{\partial \mathbf{W}} \right] = E \left\{ \boldsymbol{eeyy} \right\}, \tag{6.15}$$

where we use tensor notation neglecting suffixes $r, r-1$ denoting the layers in equation 6.9 and $\boldsymbol{e}$ denotes the backpropagated error when the true $\mathbf{W}^*$ is assumed to be equal to the current $\mathbf{W}$. Here, the error vector $\boldsymbol{e}$ is calculated from $\boldsymbol{e}_L^* = -\boldsymbol{e}^*$ backward, by the same way shown in equation 6.10. The Fisher metric is represented in component form as

$$G_{ik;\,jl}^{r,r'} = \mathrm{E} \left[ e_i^r e_j^{r'} y_k^{r-1} y_l^{r'-1} \right], \tag{6.16}$$

which is the element of $\mathbf{G}$ corresponding $(i, k)$ element of the connection matrix of the $r$th layer and $(j, l)$ element of the connection matrix of the $r'$th layer.

By using the Riemannian metric $\mathbf{G}$, the natural gradient learning method can use the following update rule,

$$\Delta \mathbf{W} = -\eta \tilde{\nabla}_{\mathbf{W}} l, \tag{6.17}$$

$$\tilde{\nabla}_{\mathbf{W}} l = \mathbf{G}^{-1} \nabla_{\mathbf{W}} l, \tag{6.18}$$

taking the Riemannian structure into account.

There exist various invariant Riemannian metrics other than the Fisher information. Ollivier (2015a) studied various of them in detail. If we use $\boldsymbol{e}^*$ instead of $\boldsymbol{e}$, we have $\mathbf{G}^* = E\left[\boldsymbol{e}^* \boldsymbol{e}^* \boldsymbol{y}\boldsymbol{y}\right]$, called the outer product metric (Ollivier, 2015b). It is different from the Fisher metric $\mathbf{G}$. It is suggestive to see the relations among these metrics. We have

$$\mathbf{G}^* = \left\langle \left| f(\boldsymbol{x}, \boldsymbol{\xi}^*) - f(\boldsymbol{x}, \boldsymbol{\xi}) \right|^2 \nabla_{\boldsymbol{\xi}} f \cdot \nabla_{\boldsymbol{\xi}} f \right\rangle + \mathbf{G}. \tag{6.19}$$

Hence, $\mathbf{G}^*$ is equal to $\mathbf{G}$ at $\boldsymbol{\xi} = \boldsymbol{\xi}^*$. Obviously $\mathbf{G}^*$ is positive-definite except in singular regions. Dauphin et al. (2014) used the absolute Hessian metric $|\mathbf{H}|$. Let us diagonalize $\mathbf{H}$ as

$$\mathbf{H} = \mathbf{O}^T \boldsymbol{\Lambda} \mathbf{O}, \tag{6.20}$$

where $\boldsymbol{\Lambda}$ is a diagonal matrix and $\mathbf{O}$ is an orthogonal matrix. Then we have

$$|\mathbf{H}| = \mathbf{O}^T |\boldsymbol{\Lambda}| \mathbf{O}. \tag{6.21}$$

The Hessian of $l(\boldsymbol{x}, \boldsymbol{\xi})$ is

$$\mathbf{H} = \left\langle \nabla_{\boldsymbol{\xi}} \nabla_{\boldsymbol{\xi}} l(\boldsymbol{x}, \boldsymbol{\xi}) \right\rangle = \left\langle \left\{ f(\boldsymbol{x}, \boldsymbol{\xi}^*) - f(\boldsymbol{x}, \boldsymbol{\xi}) \right\} \cdot \nabla_{\boldsymbol{\xi}} \nabla_{\boldsymbol{\xi}} f \right\rangle + \mathbf{G} \tag{6.22}$$

and is not always positive-definite.

There have been proposed many types of approximations of the true $\mathbf{G}$, since inverting it is costly. The simplest one is to use only the diagonal elements, setting all the off-diagonal elements equal to 0; however, it does not work well. An interesting idea is the unit-wise diagonalization proposed by Kurita (1994) and further studied by Ollivier (2015a, 2015b) in detail, in which $\mathbf{G}$ is block-diagonalized such that off-diagonal blocks are put equal to 0 and diagonal blocks consist of the weight vectors of each neuron. Let us take the $i$th neuron of layer $r$; the corresponding diagonal block is

$$\mathbf{G}_i^r = E\left[ \left(e_i^r\right)^2 \boldsymbol{y}^{r-1} \boldsymbol{y}^{r-1} \right], \tag{6.23}$$

$$G_{jk}^{r,i} = E\left[\left(e_i^r\right)^2 y_j^{r-1} y_k^{r-1}\right]. \tag{6.24}$$

Since the inverse of the block-diagonalized Fisher information matrix $\mathbf{G}$ consists of the inverses of the blocks, we consider one diagonal block corresponding to one neuron. The singular region of this neuron appears when two neurons in the previous layer become identical: $\boldsymbol{w}_1 = \boldsymbol{w}_2 = \tilde{\boldsymbol{w}}^*$. Let the weights $\boldsymbol{v} = (v_1, v_2)$ on these two neurons be transformed to $(v, z)$ as before. The latter $z = z_2 - z_1$ is not identifiable and $\partial_z l = 0$. Hence, the $z$-direction does not change. In a neighborhood of $R(\boldsymbol{w}^*, \boldsymbol{v}^*)$, the gradient is of order $u = |\boldsymbol{w}_2 - \boldsymbol{w}_1|$. However, the $(z, z)$ element of the Fisher information in this block is of order $u^2$. Hence, the change $\delta l$ due to the natural gradient is of order 1 even when $\boldsymbol{u} \approx 0$. This implies that the change in the $z$-direction is accelerated by the natural gradient and the unit-wise natural gradient method in a similar way. This is true for other unit-wise natural gradient methods, and they have the same effect of avoiding singular regions. Ollivier (2015b) shows that the unit-wise natural gradient method is much more effective than the LSTM method for recurrent networks.

## 7 Conclusion

We reviewed the long history of the dynamics of supervised learning in MLP and gave new results. An MLP is a peculiar model that includes singular regions in the parameter space. We analyzed the dynamics of learning in the neighborhood of a singular region by using an elementary MLP model. We found that such a model has an attracting singular region, a Milnor attractor, which causes plateau phenomena in learning. We showed that the natural gradient can overcome this retardation of learning.

It is surprising that large-scale networks are free of this problem. We proved that backpropagated signals from multiple output nodes prevent the existence of attracting singular regions, so no Milnor-type singular regions appear. This explains one of the merits of large-scale networks.

Studies of dynamics also have a long history; a trade-off between the speed of learning and accuracy of convergence was studied (Amari, 1967), only to be rediscovered by Heskes and Kappen (1991). The problem of singularities arose much later, and the natural gradient method was proposed to overcome it. The natural gradient is still effective, although we have proved that the Milnor-type attractor is not serious in larger-scale networks. We conclude the article with a remark that many theoretical problems aspects of deep neural networks remain to be elucidated.

## References

Amari, S. (1967). Theory of adaptive pattern classifiers. *IEEE Trans.*, *EC-16*(3), 299–307.

Amari, S. (1968). *Information theory II: Geometrical theory of information.* Tokyo: Kyouritsu Shuppan. (in Japanese)

Amari, S. (1998). Natural gradient works efficiently in learning. *Neural Computation*, *10*, 251–276.

Amari, S. (2016). *Information geometry and its applications*. New York: Springer.

Amari, S., Park, H., & Fukumizu, K. (2000). Adaptive method of realizing natural gradient learning for multilayer perceptrons. *Neural Computation*, *12*, 1399–1409.

Amari, S., Park, H., & Ozeki, T. (2006). Singularities affect dynamics of learning in neuromanifolds. *Neural Computation*, *18*, 1007–1065.

Bengio, Y. (2009). Learning deep architecture. *Foundation and Trends in Machine Learning*, *2*, 1–127.

Bryson, A. E. (1962). A steepest-ascent method for solving optimum programming problems. *Journal of Applied Mechanics*, *29*, 247–258.

Choromanska, A., Henaff, M., Mathieu, M., Aous, G. B., & LeCun, Y. (2015). The loss surfaces of multilayer networks. In *Proceedings of the 18th International Conference on Artificial Intelligence and Statistics*. Cambridge, MA: MIT Press.

Cousseau, F., Ozeki, T., & Amari, S. (2008). Dynamics of learning in multilayer perceptrons near singularities. *IEEE Transactions on Neural Networks*, *19*(8), 1313–1328.

Dauphin, Y., Pascanu, R., Gulcehre, C., Cho, K., Ganguli, S., & Bengio, Y. (2014). Identifying and attacking the saddle point problem in high-dimensional non-convex optimization. In Z. Ghahramani, M. Welling, C. Cortes, N. D. Lawrence, & N. D. Weinberger (Eds.), *Advances in neural information processing systems, 27* (pp. 2933–2941). Red Hook, NY: Curran.

Fukumizu, K., & Amari, S. (2000). Local minima and plateaus in hierarchical structures of multilayer perceptrons. *Neural Networks*, *13*, 317–327.

Fukushima, K. (1980). Neocognitron: A self-organizing neural network model for a mechanism of pattern recognition unaffected by shift in position. *Biological Cybernetics*, *36*, 93–102.

Heskes, T. M., & Kappen, B. (1991). Learning processes in artificial neural networks. *Physical Review A*, *44*, 2718–2726.

Hinton, G. E., Osindero, S., & Teh, Y. (2006). A fast learning algorithm for deep belief nets. *Neural Computation*, *18*, 1527–1554.

Hinton, G. E., & Salakhutdinov, R. (2006). Reducing the dimensionality of data with neural networks. *Science*, *313*, 504–507.

Hochreiter, S., & Schumdhuber, J. (1997). Long short-term memory. *Neural Computation*, *9*, 1735–1780.

Kurita, T. (1994). Iterative weighted least squares algorithms for neural networks classifiers. *New Generation Computing*, *12*, 375–394.

LeCun, Y., Bottou, L., Bengio, Y., & Haffner, P. (1998). Gradient-based learning applied to document recognition. *Proceedings of IEEE*, *86*, 2278–2324.

Oizumi, M., Tsuchiya, N., & Amari, S. (2016). Unified framework for information integration based on information geometry. *Proceedings of the National Academy of Sciences*, *113*(51), 14817–14822.

Ollivier, Y. (2015a). Riemannian metrics for neural networks I: Feedforward networks. *Information and Inference*, *4*(2), 108–153.

Ollivier, Y. (2015b). Riemannian metrics for neural networks II: Recurrent networks and learning symbolic data sequences. *Information and Inference*, *4*(2), 154–193.

Park, H., Amari, S., & Fukumizu, K. (2000). Adaptive natural gradient learning algorithms for various stochastic models. *Neural Networks*, *13*, 755–764.

Pascanu, R., & Bengio, Y. (2013). *Revisiting natural gradient for deep networks*. arXiv:1301.3584v7.

Poole, B., Lahiri, S., Raghu, M., Sohl-Dickstein, J., & Ganguli, S. (2016). Exponential expressivity in deep neural networks through transient chaos. In D. D. Lee, M. Sugiyama, U. V. Luxburg, I. Guyon, & R. Garnett (Eds.), *Advances in neural information processing systems*, *29* (pp. 3360–3368). Red Hook, NY: Curran.

Robbins, H., & Monro, S. (1951). A stochastic approximation method. *Annals of Mathematical Statistics*, *22*(3), 400. doi:10.1214/aoms/1177729586

Rosenblatt, F. (1962). *Principles of neurodynamics.* Washington, DC: Spartan Books.

Rumelhart, D. E., Hinton, G. E., & Williams, R. J. (1986). Learning representations by back-propagating errors. *Nature*, *323*, 533–536.

Saxe, A. M., McClelland, J. L., & Ganguli, S. (2013). *Exact solutions to the nonlinear dynamics of learning in deep linear neural networks*. arXiv:1312.6120.

Schmidhuber, J. (2015). Deep learning in neural networks: An overview. *Neural Networks*, *61*, 85–117.

Tsypkin, Y. Z. (1966). Adaptation, training and self-organization in automatic control systems. *Automation and Remote Control*, *27*, 23–61. (in Russian)

Watanabe, S. (2009). *Algebraic geometry and statistical learning theory.* Cambridge: Cambridge University Press.

Wei, H., Zhang, J., Cousseau, F., Ozeki, T., & Amari, S. (2008). Dynamics of learning near singularities in layered networks. *Neural Computation*, *20*, 813–843.

Werbos, W. J. (1974). *Beyond regression: New tools for prediction and analysis in the behavioral sciences.* PhD diss., Harvard University.

---

Received March 8, 2017; accepted July 31, 2017.