Analyzing Learning Dynamics: How to Average?

Christian Goerick

2000

Preprint:

Proceedings of the International Joint Conference on Neural Networks (IJCNN 2000),
volume II, pages 191-195, 2000

Analyzing Learning Dynamics: How to Average?

Christian Goerick *
Institut für Neuroinformatik, Lehrstuhl Theoretische Biologie
Ruhr-Universität Bochum, 44780 Bochum, Germany
goerick@neuroinformatik.ruhr-uni-bochum.de

Abstract

Pattern-based learning processes are usually analyzed by means of probability density functions of the weights or moments thereof. During the derivation of these equations, some averaging has to be performed. In this paper, we will show that the manner of averaging is crucial for the results of the analysis. We will do this by comparing two types of analysis (Langevin type and discrete-time moments) for one learning system.

Keywords: learning dynamics, pattern-based learning, Langevin equation, nonlinear analysis

1 Introduction

Learning dynamics have been proven a valuable source of information for designing artificial neural networks [1, 2, 3, 4, 5, 6, 7, 8]. There are two established ways for analyzing pattern-based learning dynamics. The first one is based on methods from statistical mechanics. It is well-suited for considerations in the thermo-dynamical limit and will not be further investigated here [1]. The second one is based on stochastic differential equations of the Master and Fokker-Planck equation type [3]. A recently suggested alternative to this approach is best characterized as "discrete-time moments", where the discrete-time evolution of weights and moments thereof are studied [9]. The latter two approaches are well-suited for the analysis of small to medium sized networks. They are based on the formulation of the learning process as a Markov process and permit a description of the evolution of the weights during the learning process either as a probability density function or as moments thereof.

During the derivation of these equations, some averaging has to be performed. The standard approach for averaging is the one corresponding to a Langevin equation [10, 11]. In this case, the averaging is performed for all relevant parameters and these averages are then plugged into the learning equations. In order to maintain the stochastic nature of the learning process, an additional noise term is added. This approach can be represented symbolically by the differential equation

$$\dot{\langle w \rangle} = H(\langle w \rangle) + \eta \quad , \tag{1}$$

where w denotes the weight, H the update function and η some noise.

In this paper we will show that this kind of averaging is not sufficient to describe the effects of nonlinear elements in the learning equations. This is a very important point if training is to be explained beyond linear analysis.

In order to make our point, we will proceed as follows. At first, we will give the stochastic formulation of the learning process and some tools for the analysis. Then an iterative system will be presented that will serve as our demonstration object. This system will be analyzed in two different ways and the results will be compared and discussed.

2 Incremental discrete-time learning systems

We will start from the generic incremental learning rule

$$\mathbf{w}^{n+1} = \tilde{\mathbf{H}}(\mathbf{w}^n, \mathbf{x}) = \mathbf{w}^n + \alpha \mathbf{H}(\mathbf{w}^n, \mathbf{x}) \tag{2}$$

* This work was sponsored by BMB+F as part of the AENEAS project, grant number 01 IN 505 C 4.
Current address: HONDA R&D Europe (Deutschland) GmbH, Carl-Legien-Str. 30, 63073 Offenbach, Germany

The change of the weights from time step n to $n+1$ is considered as a Markov process with

$$p(\mathbf{w}^{n+1}) = \int_{-\infty}^{\infty} p(\mathbf{w}^{n+1}|\mathbf{w}^n)\, p(\mathbf{w}^n)\, d\mathbf{w}^n \tag{3}$$

$$p(\mathbf{w}^{n+1}|\mathbf{w}^n) = \int_{-\infty}^{\infty} \delta^N\big(\mathbf{w}^{n+1} - \tilde{\mathbf{H}}(\mathbf{w}^n, \mathbf{x})\big)\, \rho(\mathbf{x})\, d\mathbf{x} \quad . \tag{4}$$

Here, $\mathbf{w}^n \in \mathbb{R}^N$ denotes the state of the whole network at time step n where $n \in \mathbb{N}_0$ and $\mathbf{x} \in \mathbb{R}^M$ denotes the vector of the training data with the corresponding distribution $\rho(\mathbf{x})$. Therefore, the value of the transition probability $p(\mathbf{w}^{n+1}|\mathbf{w}^n)$ at $\mathbf{w}^{n+1}$ given $\mathbf{w}^n$ is determined by integrating across all matching values of $\rho(\mathbf{x})$. For such a system, an equation for the evolution of the expected value can be derived [9], as

$$\mathrm{E}\left\{\mathbf{w}^{n+1}\right\} = \mathrm{E}\left\{\mathbf{w}^n\right\} + \alpha \mathrm{E}\left\{\mathrm{E}\left\{\mathbf{H}(\mathbf{w}^n, \mathbf{x})\right\}_{\mathbf{x}}\right\}_{\mathbf{w}^n} \quad . \tag{5}$$

This deterministic equation can by analyzed by means of nonlinear difference equations. A corresponding equation for the covariance matrix can be derived be the same means. This equation (5) and the one for the covariance form tools for analyzing the qualitative behavior of discrete time incremental learning processes. The second term on the right hand side of equation (5) is the important one. We will show how carrying out the expectations in different ways can have significant effects on the analysis.

3 The system

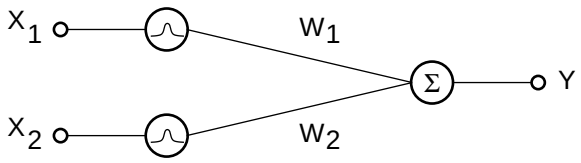

Figure 1: Schematics of the Neural Network.

We are going to study a very reduced system. It consists of two inputs x_1 and x_2 and one output y. The inputs are propagated across Gaussian type nonlinearities $\theta(z)$, weighted by w_1 and w_2 and summed to yield the output y. The functional mapping realized by this network is

$$y = \theta(x_1)w_1 + \theta(x_2)w_2 \quad . \tag{6}$$

The nonlinearity $\theta(\cdot)$ will be chosen as $\theta(z) = 1 - z^2$, which is, except for a neglected factor $1/2$ in front of z^2, a first nonlinear approximation of $e^{-\frac{z^2}{2}}$. The neglected factor does not influence the qualitative behavior of the system.

Our goal is to study the behavior of the expected value of the weights. In order to do so, we are going to follow two approaches. One kind of analysis follows the Langevin type of averaging, and the other follows the discrete-time moments approach with the full nonlinearity of the transfer functions. We will see that the results differ quite significantly.

The network is to learn the following task: The inputs x_1 and x_2 are independent Gaussian noise with the expected value $\mu = 0$ and the variance σ^2. The target output t is zero for all inputs. This task is not realistic at all but serves as an instructive example. It can be considered as a model for irrelevant inputs [9]. The error signal is defined as

$$E = \frac{1}{2}(t-y)^2 = \frac{1}{2}y^2 \quad . \tag{7}$$

A gradient based learning rule yields

$$\begin{aligned} w_j^{n+1} &= w_j^n - \alpha \frac{\partial E^n}{\partial w_j} \qquad j = 1, 2 \\ &= w_j^n - \alpha \theta(x_j^n) y^n \quad . \end{aligned} \tag{8}$$

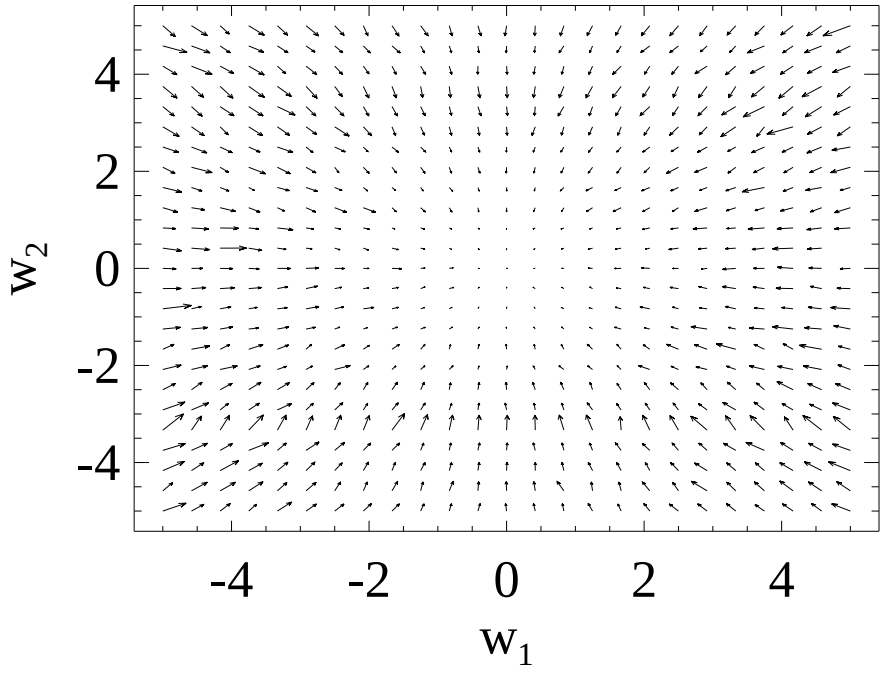

Figure 2: Difference vector field of the two-dimensional system. Averaged across 200 runs. The arrows indicate the average direction of movement for a given weight value.

What is the expected behavior of the weights w_1 and w_2 during learning? The desired mapping of noisy inputs to a constant value of zero can only be realized by setting both weights w_1 and w_2 to zero. Other solutions are not possible (cf. figure 2, where 200 runs of equation (8) are averaged). It can clearly be seen that the expected direction of movement of the weights is towards the origin.

4 Langevin type analysis: results

For the Langevin type approach, equation (5) looks as follows:

$$\mathrm{E}\left\{\mathbf{w}^{n+1}\right\} = \mathrm{E}\left\{\mathbf{w}^{n}\right\} + \alpha\mathbf{H}(\mathrm{E}\left\{\mathbf{w}^{n}\right\}, \mathrm{E}\left\{\mathbf{x}^{n}\right\}) + \mathrm{E}\left\{\eta\right\} \tag{9}$$

i.e., the expectation is draw into the function H. The noise η is here without loss of generality chosen as expectation free, $\mathrm{E}\left\{\eta\right\} = 0$. As a direct result we have the equations for the expected value $\mathrm{E}\left\{w^{n}\right\} = \mu^{n}$

$$\mu_j^{n+1} = \mu_j^n - \alpha(\mu_1^n + \mu_2^n) \qquad j = 1, 2 \tag{10}$$

One can show that both equations have stable fixed points at

$$\mu_1^n = -\mu_2^n \tag{11}$$

and that the covariance at these points converges towards zero.

What does this mean? The results shown here are in direct contradiction to the expected results from the previous section. Figure 3 shows the difference vector field for the expected values of the weights for the Langevin type of analysis. It has not much in common with figure 2, where the averaged difference vector field of the original system is shown.

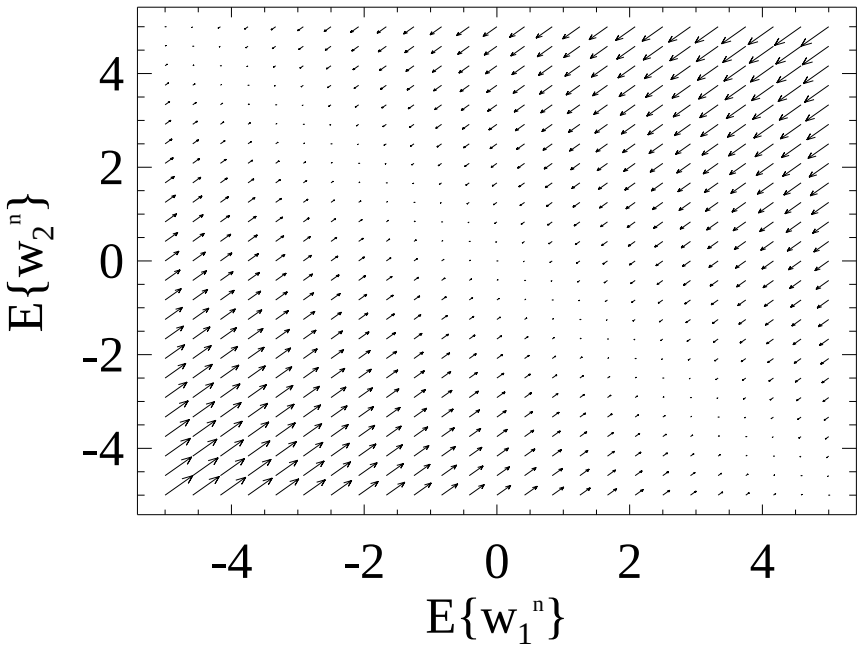

Figure 3: Difference vector field of the two-dimensional system. Langevin type analysis.

5 Full nonlinearity: results

If the expectations in equation (5) are carried out as accurate as possible, the following equations for the evolution of the expected values are obtained:

$$\mu_1^{n+1} = \mu_1^n - \alpha\left(\mu_1^n(1-2\sigma^2+3\sigma^4) + \mu_2^n(1-\sigma^2)^2\right) \quad (12)$$

$$\mu_2^{n+1} = \mu_2^n - \alpha\left(\mu_1^n(1-\sigma^2)^2 + \mu_2^n(1-2\sigma^2+3\sigma^4)\right) \quad (13)$$

One can show that, except for isolated values of σ^2, this system of equations does not have any fixed point except the origin. I.e., $\mu_1^n = \mu_2^n = 0$ is the only fixed point. The variance σ^2 of the input variable x_1 and x_2 is the crucial contribution. This variance influences the expected values of the weights due to the nonlinearity of the transfer functions. Figure 4 show the difference vector field for the equations (12) and (13). It coincides with figure 2 quite nicely.

6 Summary

In this paper, we have shown that the decision how to average when analyzing the learning dynamics is crucial for obtaining convergence to the proper solution. For this, the basically linear Langevin type approach was compared to a fully nonlinear analysis based on discrete-time moments of the weights. The expected value of the weights is influenced by the variance of the input variables due to the nonlinearity of the transfer functions. A linear analysis is not sufficient to explain this kind of behavior.

References

[1] David Saad and Sara A. Solla. On-Line learning in soft committee machines. *Physical Review E*, Vol. 52(No. 4):pp.4225–4243, 1995.

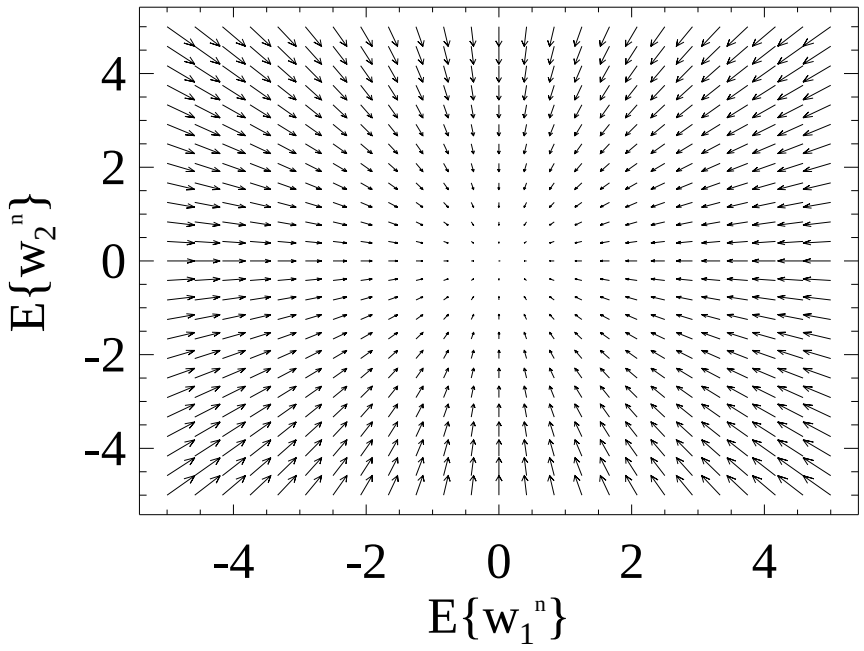

Figure 4: Difference vector field of the expected value of the two-dimensional system. Full input nonlinearity.

[2] Tom M. Heskes and Bert Kappen. Learning processes in neural networks. *Physical Review A*, Vol. 44:pp. 2718–2726, 1991.

[3] Tom M. Heskes. On Fokker-Planck approximations of on-line learning processes. *Journal of Physics, Part A: Mathematical and General Physics*, Vol. 27:pp.5145–5160, 1994.

[4] Wim Wiegerinck and Tom Heskes. How dependencies between successive examples affect on-line learning. *Neural Computation*, Vol. 8(No. 8):pp.1743–1765, 1996.

[5] David Saad and Sara A. Solla. Learning with noise and regularizers in multilayer neural networks. In *Advances in Neural Information Processing NIPS 9*, 1996.

[6] Peter Sollich and David Barber. Online learning from finite training sets: An analytical case study. In *Advances in Neural Information Processing NIPS 9*, 1996.

[7] K.-R. Müller, M. Finke, N. Murata, K. Schulten, and S. Amari. A numerical study on learning curves in stochastic multilayer feedforward networks. *Neural Computation*, Vol. 8(No. 5):pp.1085–1106, 1996.

[8] Nikolaos Ampazis, S.J. Perantonis, and J.G. Taylor. Dynamics of multilayer networks in the vicinity of temporary minima. *Neural Networks*, Vol. 12:pp.43–58, 1999.

[9] Christian Goerick. How irrelevant inputs affect MLP pattern based learning. In *ICANN99, Proceedings of the International Conference on Artificial Neural Networks*, 1999.

[10] N.G. van Kampen. *Stochastic processes in physics and chemistry*. North-Holland, revised and enlarged edition, 1992.

[11] C.W. Gardiner. *Handbook of Stochastic Methods for Physics, Chemistry and Natural Sciences*. Springer Verlag, 1990.