# Learning Dynamics as a Many-Body Problem

K. Y. Michael Wong

(アブストラクト訳)

ニューラルネットワークにおける計算の理論の重要な課題として、学習ダイナミクスのマクロな変数による記述の問題がある。最近よく議論されているオンライン学習の話では、学習の例題数が無制限であるとする非現実的な仮定をしていることが多い。例題数が限られている場合には、従来の理論は漸近領域や単純な学習則などに話が限定されていた。私たちの理論では、例題数が限られているバッチ学習のモデルに多体問題の理論を取り入れ、任意の学習評価関数についての取り扱いが可能になったのみならず、例題を繰り返し使うことによる時間相関の影響も完全に取り入れられるようになった。

# Learning Dynamics as a Many-Body Problem

**K. Y. Michael Wong**

Department of Physics, Hong Kong University of Science and Technology,
Clear Water Bay, Kowloon, Hong Kong.
Email: phkywong@ust.hk

**Abstract:** An important issue in neural computing concerns the description of learning dynamics with macroscopic dynamical variables. Recent progress on *on-line* learning only addresses the often unrealistic case of an infinite training set. For restricted training sets, previous studies have so far been limited to asymptotic dynamics or simple learning rules. We introduce a many-body theory to model batch learning of restricted sets of examples, widely applicable to *any* learning cost function, and fully taking into account the temporal correlations introduced by the recycling of the examples.

## 1. Introduction

The dynamics of learning in neural computing is a complex problem on both the macroscopic and microscopic levels. Recently, much progress has been made on modelling the dynamics of *on-line* learning, in which an independent example is generated for each learning step [1, 2]. Since statistical correlations among the examples can be ignored, the dynamics can be described by instantaneous dynamical variables, facilitating a simple description.

However, on-line learning represents an ideal case in which the network has access to an almost infinite training set, whereas in many applications, the collection of training examples may be costly. A restricted set of examples introduces extra temporal correlations during learning, and the dynamics is much more complicated. As a result, progress has so far been limited to Adaline learning [3, 4, 5], *linear* perceptrons learning nonlinear rules [6, 7], Hebbian learning [8] and binary weights [12].

Here we introduce a many-body theory to model batch learning of restricted sets of examples, fully taking into account the temporal correlations during learning, and exact for large networks. It is widely applicable to any learning rule which minimizes an *arbitrary* cost function by gradient descent.

## 2. Formulation

Consider the single-layer perceptron with $N \gg 1$ input nodes $\{\xi_j\}$ connecting to a single output node by the weights $\{J_j\}$. The inputs $\xi_j$ are Gaussian

variables with mean 0 and variance 1, and the output state $S$ is a function $f(x)$ of the *activation* $x$ at the output node, i.e.

$$S = f(x); \quad x = \vec{J} \cdot \vec{\xi}. \tag{1}$$

For binary outputs, $f(x) = \mathrm{sgn}x$.

The network is assigned to "learn" $p \equiv \alpha N$ examples which map inputs $\{\xi_j^\mu\}$ to the outputs $\{S_\mu\}$ $(\mu = 1, \ldots, p)$. In the case of random examples, $S_\mu$ are random binary variables, and the perceptron is used as a storage device. In the case of teacher-generated examples, $S_\mu$ are the outputs generated by a teacher perceptron $\{B_j\}$, namely

$$S_\mu = f(y_\mu); \quad y_\mu = \vec{B} \cdot \vec{\xi}^\mu. \tag{2}$$

Batch learning by gradient descent is achieved by adjusting the weights $\{J_j\}$ iteratively so that a certain cost function in terms of the activations $\{x_\mu\}$ and $\{y_\mu\}$ is minimized. Hence we consider a general cost function

$$E = -\sum_\mu g(x_\mu, y_\mu). \tag{3}$$

The precise functional form of $g(x, y)$ depends on the adopted learning algorithm. For Hebbian learning, $g(x, y) = xf(y)$, for Adaline learning, $g(x, y) = -(f(y) - x)^2/2$ and for backprop, $g(x, y) = -(f(y) - f(x))^2/2$.

To ensure that the perceptron is regularized after learning, it is customary to introduce a weight decay term. Furthermore, to avoid the system from getting trapped in local minima, it is customary to add noise in the dynamics. Hence the gradient descent dynamics for batch learning is given by

$$\frac{dJ_j(t)}{dt} = \frac{1}{N}\sum_\mu g'(x_\mu(t), y_\mu)\xi_j^\mu - \lambda J_j(t) + \eta_j(t), \tag{4}$$

where $g'(x, y)$ represents the partial derivative of $g(x, y)$ with respect to $x$, $\lambda$ is the weight decay strength, and $\eta_j(t)$ is noise at temperature $T$ with

$$\langle \eta_j(t) \rangle = 0 \quad \text{and} \quad \langle \eta_j(t)\eta_k(s) \rangle = \frac{2T}{N}\delta_{jk}\delta(t - s). \tag{5}$$

# 3. The Cavity Method and Many-Body Theory

The cavity method is the starting point of our work [9]. It has been used in studying the physical properties of magnetic and disordered systems. For neural networks, it has been used to study the *steady-state* properties of learning [10, 11]. The *dynamics* of learning has been studied for perceptrons with *discrete* weights, using a generating function approach, which is mathematically equivalent to the cavity method [12]. However, the cavity method has not been applied to the *dynamics* of learning with *analog* weights and general learning rules, especially in the transient regime.

The method uses a self-consistency argument to consider what happens when a new example is added to a training set. The central quantity in this method is the *cavity activation*, which is the activation of the new example on a node for a perceptron trained without that example. Since the original network has no information about the new example, the cavity activation is a Gaussian variable. Specifically, denoting the new example by the label 0, its cavity activation at time $t$ is

$$h_0(t) = \vec{J}(t) \cdot \vec{\xi}^0. \tag{6}$$

For large $N$, $h_0(t)$ is a Gaussian variable. For random examples, its covariance is given by the correlation function $C(t,s)$ of the weights at times $t$ and $s$,

$$\langle h_0(t) h_0(s) \rangle = \vec{J}(t) \cdot \vec{J}(s) \equiv C(t,s), \tag{7}$$

where we have made use of the independence of the random variables $\xi_j^0$ and $\xi_k^0$ for $j \neq k$. For teacher-generated examples, the distribution is further specified by the teacher-student correlation $R(t)$, given by

$$\langle h_0(t) y_0 \rangle = \vec{J}(t) \cdot \vec{B} \equiv R(t). \tag{8}$$

Now suppose the perceptron incorporates the new example at the batch-mode learning step at time $s$. Then the activation of this new example at a subsequent time $t > s$ will no longer be a random variable. Furthermore, the activations of the original $p$ examples at time $t$ will also be adjusted from $\{x_\mu(t)\}$ to $\{x_\mu^0(t)\}$ because of the newcomer, which will in turn affect the evolution of the activation of example 0, giving rise to the so-called Onsager reaction effects. This makes the dynamics complex, but fortunately for large $p \sim N$, we can assume that the adjustment from $x_\mu(t)$ to $x_\mu^0(t)$ is small, and linear response theory can be applied.

In the linear response theory for many-body systems, one is interested in how the effects of a delta-function disturbance at time $s$ propagates to a later time $t$. This is called the Green's function $G(t,s)$. In the simulational experiment in Fig. 1(a), we compare the evolution of two perceptrons $\{J_j(t)\}$ and $\{J_j^0(t)\}$ in Adaline learning. At the initial state $J_j^0(0) - J_j(0) = 1/N$ for all $j$, but otherwise their subsequent learning dynamics are exactly identical. Hence the total sum $\sum_j (J_j^0(t) - J_j(t))$ provides an estimate for the averaged Green's function $G(t,0)$, which gives an excellent agreement with the Green's function obtained from many-body theory.

Superposing the effects of the gradient term $g_0'(s)$ of example 0 due to its presence in the learning history, we have

$$x_0(t) - h_0(t) = \int ds G(t,s) g_0'(s). \tag{9}$$

Statistically, this equation enables us to express the activation distribution in terms of the cavity activation distribution, and we can study the macroscopic dynamics.

In another simulational experiment in Fig. 1(b), we first measure the Green's function as in Fig. 1(a), and monitor the evolution of the activations $x_\mu(t)$ of the examples. The corresponding cavity activations are then computed from Eq. (9) and presented in the histogram in Fig. 1(b). It resembles a Gaussian distribution, whose computed mean and variance agree with those obtained from the macroscopic dynamics of the order parameters.

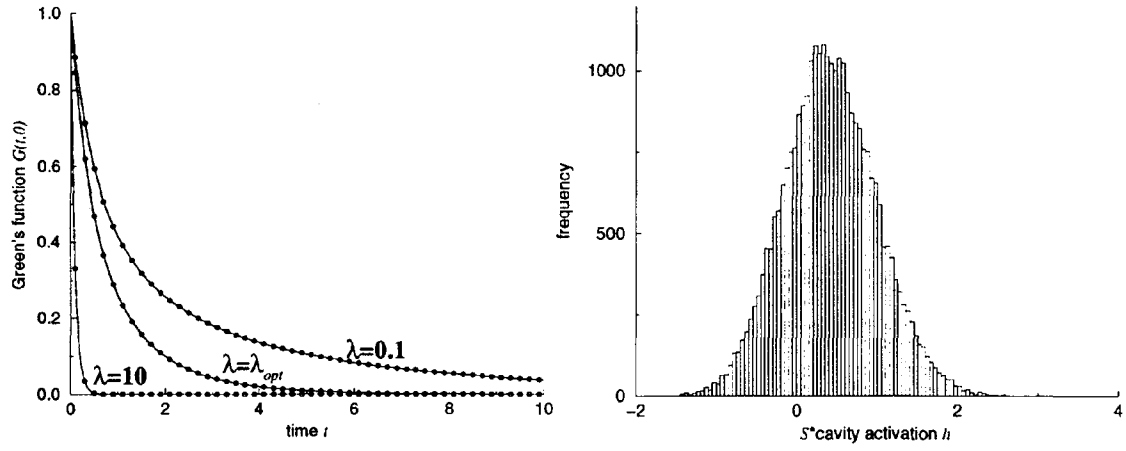


Figure 1: (a) The Green's function $G(t,0)$ for Adaline learning at $\alpha = 1.2$ and $T = 0$ for different weight decay strengths $\lambda$, $\lambda_{opt}$ being given by Eq. (11). Theory: solid line, simulation: symbols. (b) Histogram of the cavity activations as computed from Eq. (9) for $\alpha = 1.2$, $\lambda = 0.1$, $N = 500$, 50 samples at $t = 2$. The mean and width of the histogram are 0.413 and 0.574 respectively, compared with the values of 0.415 and 0.573 computed from the dynamics of the order parameters.

# 4. A Solvable Case

Here for illustration, we present the results for the Adaline rule. This is a common learning rule and bears resemblance with the more common back-propagation rule. Theoretically, its dynamics is particularly convenient for analysis since $g''(x) = -1$, rendering the weight Green's function time translation invariant, i.e. $G(t,s) = G(t-s)$. In this case, the dynamics can be solved by Laplace transform.

To monitor the progress of learning, we are interested in three performance measures: (a) *Training error* $\epsilon_t$, which is the probability of error for the training examples. It is given by $\epsilon_t = \langle \Theta(-x\mathrm{sgn}y) \rangle_{xy}$, where the average is taken over the joint distribution $p(x,y)$. (b) *Test error* $\epsilon_{test}$, which is the probability of error when the inputs $\xi_j^\mu$ of the training examples are corrupted by an additive Gaussian noise of variance $\Delta^2$. This is a relevant performance measure when the perceptron is applied to process data which are the corrupted versions of the training data. It is given by $\epsilon_{test} = \langle H(x\mathrm{sgn}y/\Delta\sqrt{C(t,t)}) \rangle_{xy}$, where $H(x)$ is the probability that a Gaussian variable, with mean 0 and variance 1, is larger than $x$. When $\Delta^2 = 0$, the test error reduces to the training

error. (c) *Generalization error* $\epsilon_g$, which is the probability of error for an arbitrary input $\xi_j$ when the teacher and student outputs are compared. It is given by $\epsilon_g = \arccos[R(t)/\sqrt{C(t,t)}]/\pi$.

To verify the theoretical predictions, simulations were done with $N = 500$ and using 50 samples for averaging. As shown in Figs. 2-3, the agreement is excellent. For illustration, we discuss the following aspects of the results.

*1) Convergence time:* Figure 2(a) shows the evolution of the average activation. Figure 2(b) shows the behaviour of the *convergence time*, which is defined as the time for the average activation to reach half its asymptotic value. In the limit of few and numerous examples, the convergence times $\tau$ are respectively given by

$$\lim_{\alpha\to 0} \tau = \frac{\ln 2}{1+\lambda}, \quad \text{and} \quad \lim_{\alpha\to\infty} \tau = \frac{\ln 2}{\alpha}. \tag{10}$$

Thus for few examples, the convergence rate is determined by the weight decay strength, whereas for numerous examples, the convergence rate is determined by the size of the training set.

The convergence time is different from the *relaxation time* in early studies, which is more appropriate for asymptotic dynamics rather than the transient behaviour [3]. For example, in the limit of few examples and weak weight decay, the convergence time approaches a constant, whereas the relaxation time diverges as $\lambda^{-1}$. This is because transient learning is dominated by a significant growth of the projection onto the highly degenerate space of zero training error, whereas steady-state learning merely involves relaxation in this space by weight decay.

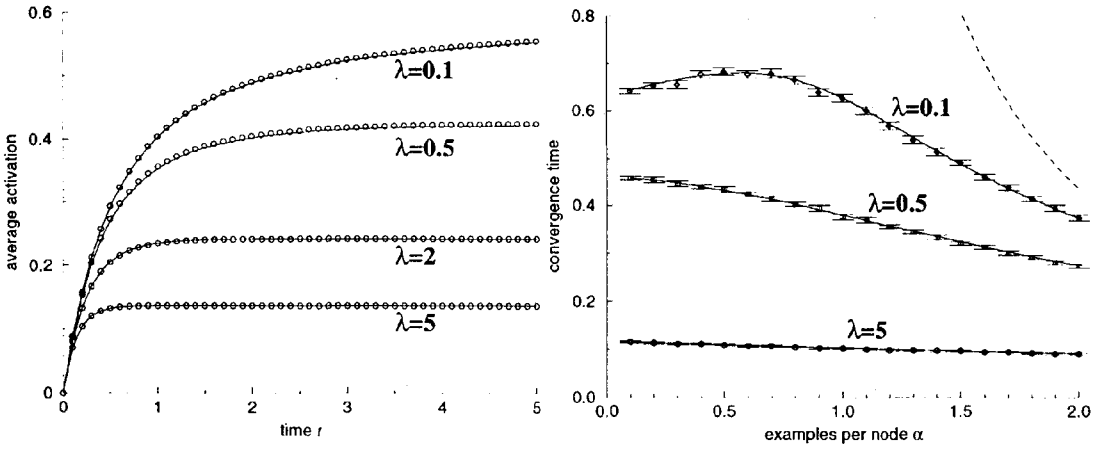


Figure 2: (a) The evolution of the average activation at $\alpha = 1.5$ for different weight decay strengths $\lambda$. (b) Dependence of the convergence time on the examples per node $\alpha$ for different weight decay strengths $\lambda$. The dashed line is 0.3 times the relaxation time in [3] for $\lambda = 0.1$.

*2) Behaviour of* $\epsilon_g$*:* Figure 3(a) shows the evolution of the generalization error at $T = 0$. When the weight decay strength varies, the steady-state

generalization error is minimized at the optimum

$$\lambda_{opt} = \frac{\pi}{2} - 1, \tag{11}$$

which is independent of $\alpha$. For $\lambda < \lambda_{opt}$, the generalization error is a non-monotonic function in learning time. Hence the dynamics is plagued by *over-training*, and it is desirable to introduce *early stopping* to improve the perceptron performance. Similar behaviour is observed in linear perceptrons [5, 6, 7].

Figure 3(b) compares the generalization errors at the steady-state and the early stopping point. It shows that early stopping improves the performance for $\lambda < \lambda_{opt}$, which becomes near-optimal when compared with the best result at $\lambda = \lambda_{opt}$. Hence early stopping can speed up the learning process without significant sacrifice in the generalization ability. However, it cannot outperform the optimal result at steady-state. This agrees with a recent empirical observation that a careful control of the weight decay may be better than early stopping in optimizing generalization [13].

*3) Behaviour of $\epsilon_{test}$:* Again, there is an optimal weight decay parameter $\lambda_{opt}$ which minimizes the test error. Furthermore, when the weight decay is weak, early stopping is desirable. The optimal weight decay $\lambda_{opt}$ for the test error depends on input noise. For random examples, $\lambda_{opt} = \alpha\Delta^2$. Hence when the perceptron is applied to process increasingly noisy data, weight decay becomes more and more important in performance enhancement. For teacher-generated examples, $\lambda_{opt} \propto \Delta^2$ approximately.

It is interesting to consider the weight decay $\lambda_{ot}$ below which overtraining occurs for the test error. For random examples, $\lambda_{ot}$ coincides with $\lambda_{opt}$. For teacher-generated examples, $\lambda_{ot} \approx \lambda_{opt}$ to the lowest order approximation.

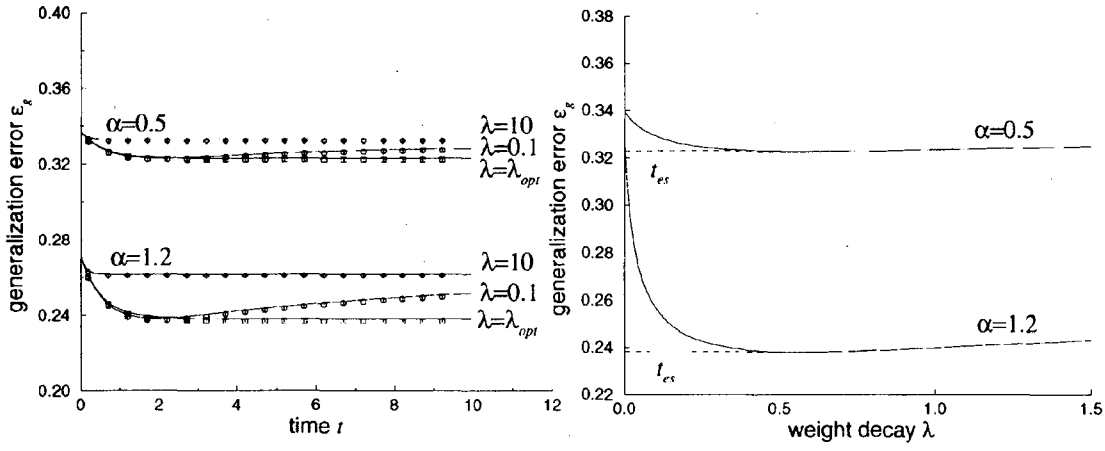


Figure 3: (a) The evolution of the generalization error at $T = 0$ for $\alpha = 0.5, 1.2$ and different weight decay strengths $\lambda$. (b) Comparing the generalization error at the steady state ($\infty$) and at the early stopping point ($t_{es}$) for $\alpha = 0.5, 1.2$ and $T = 0$.

# 5. Conclusion

Based on the cavity method, we have introduced a many-body theory for modelling the dynamics of learning, which is much more versatile than existing theories. It is more realistic in many situations than theories of on-line learning. Compared with early work on Adaline learning [3, 4], which focuses more on the asymptotic dynamics, we have a better understanding on the transient behaviour and the convergence time. Compared with recent work on Hebbian learning [8], which is based on certain self-averaging assumptions, our theory develops naturally from the stochastic nature of the cavity activations. Hence our theory has the best potential to extend to more sophisticated multilayer networks of practical importance.

We consider the present work as only the beginning of a new area of study. Many interesting and challenging issues remain to be explored. For example, while the dynamics in the present work corresponds to the limit of very low learning rate, it is interesting to generalize the method to dynamics with discrete learning steps of finite learning rates.

**Acknowledgments**

The author wishes to thank Song Li and Yiu Wai Tong for collaborations, and the Yukawa Institute for hospitality and invitation to deliver the presentation. He is indebted to Prof. Satoshi Takada, who provided guidance on many-body theory many years ago. This work was supported by the Research Grant Council of Hong Kong (HKUST6130/97P).

# References

[1] D. Saad and S. Solla, *Phys. Rev. Lett.* **74**, 4337 (1995).
[2] D. Saad and M. Rattray, *Phys. Rev. Lett.* **79**, 2578 (1997).
[3] J. Hertz, A. Krogh and G. I. Thorbergssen, *J. Phys. A* **22**, 2133 (1989).
[4] M. Opper, *Europhys. Lett.* **8**, 389 (1989).
[5] A. Krogh and J. A. Hertz, *J. Phys. A* **25**, 1135 (1992).
[6] S. Bös and M. Opper, *J. Phys. A* **31**, 4835 (1998).
[7] S. Bös, *Phys. Rev. E* **58**, 833 (1998).
[8] A. C. C. Coolen and D. Saad, Preprint KCL-MTH-98-08 (1998).
[9] M. Mézard, G. Parisi and M. Virasoro, *Spin Glass Theory and Beyond* (World Scientific, Singapore) (1987).
[10] K. Y. M. Wong, *Europhys. Lett.* **30**, 245 (1995).
[11] K. Y. M. Wong, *Advances in Neural Information Processing Systems* **9**, 302 (1997).
[12] H. Horner, *Z. Phys. B* **86**, 291 (1992); *Z. Phys. B* **87**, 371 (1992).
[13] L. K. Hansen, J. Larsen and T. Fog, *IEEE Int. Conf. on Acoustics, Speech, and Signal Processing* **4**, 3205 (1997).