Eigenvalues of Covariance Matrices: Application to Neural-Network Learning

Yann Le Cun,[1] Ido Kanter,[2] and Sara A. Solla[1]
[1]*AT&T Bell Laboratories, Holmdel, New Jersey 07733*
[2]*Department of Physics, Bar Ilan University, Ramat Gan, 52100, Israel*
(Received 2 January 1991)

The learning time of a simple neural-network model is obtained through an analytic computation of the eigenvalue spectrum for the Hessian matrix, which describes the second-order properties of the objective function in the space of coupling coefficients. The results are generic for symmetric matrices obtained by summing outer products of random vectors. The form of the eigenvalue distribution suggests new techniques for accelerating the learning process, and provides a theoretical justification for the choice of centered versus biased state variables.

PACS numbers 87 10.+e, 02.50+s, 05.20−y

The application of statistical-physics methods to the investigation of neural-network models has provided tools to characterize a variety of aspects of the static performance of trained networks (see, for example, Ref. 1). Here we use these methods to investigate a somewhat overlooked problem of considerable conceptual and practical importance: the dynamics of learning. The goal is to characterize the time scales that control the dynamical behavior of the synaptic coefficients as they are updated during the learning process. Causes for the slowest time constants can be thus identified, and specific prescriptions to eliminate their effect result in practical methods to accelerate convergence.

The discussion focuses on layered networks with no feedback, a class of architectures remarkably successful at perceptual tasks such as speech and image recognition.[2,3] In this paper we derive rigorous results for a single linear unit, and discuss the generalization of the results to multilayer nonlinear networks, under suitable conditions. The analytic calculation provides a rigorous and general result for the distribution of eigenvalues of a symmetric matrix constructed as a sum of outer products of random vectors with independent components. This result is of interest beyond its application to the analysis of neural-network learning.

Multilayer networks are composed of model neurons interconnected through a feed-forward graph. The state x_i of the ith neuron is computed from the states $\{x_j\}$ of the set S_i of neurons that feed into it through the total input (or *induced local field*) $a_i = \sum_{j \in S_i} w_{ij} x_j$. The coefficient w_{ij} of the linear combination is the coupling from neuron j to neuron i. The local field a_i determines the state x_i through a nonlinear differentiable function f called the *activation function*: $x_i = f(a_i)$. The activation function is often chosen to be the hyperbolic tangent or a similar sigmoid function.

The connection graph of multilayer networks has no feedback loop, and the stable state is computed by propagating state information from the input units (which receive no input from other units) to the output units (which propagate no information to other units). The initialization of the state of the input units through an input vector $\mathbf{X}$ results in an output vector $\mathcal{O}$ describing the state of the output units. The network thus implements an input-output map, $\mathcal{O} = \mathcal{O}(\mathbf{X}, \mathbf{W})$, which depends on the values assigned to the vector $\mathbf{W}$ of synaptic couplings.

The learning process is formulated as a search in the space $\mathbf{W}$, so as to find an optimal configuration $\mathbf{W}^*$ which minimizes an objective function $E(\mathbf{W})$. Given a training set of p input vectors $\mathbf{X}^\mu$ and their desired outputs $\mathbf{D}^\mu$, $1 \le \mu \le p$, the cost function

$$E(\mathbf{W}) = \frac{1}{2p} \sum_{\mu=1}^{p} ||\mathbf{D}^\mu - \mathcal{O}(\mathbf{X}^\mu, \mathbf{W})||^2 \quad (1)$$

measures the discrepancy between the actual behavior of the system and the desired behavior. The minimization of E with respect to $\mathbf{W}$ is usually performed through iterative updates using some form of gradient descent:

$$\mathbf{W}(k+1) = \mathbf{W}(k) - \eta \nabla E, \quad (2)$$

where η is used to adjust the size of the updating step, and ∇E is an estimate of the gradient of E with respect to $\mathbf{W}$. The commonly used backpropagation algorithm, popularized by Ref. 4, provides an efficient way of estimating ∇E for a multilayer network.

The dynamical behavior of learning algorithms based on the minimization of $E(\mathbf{W})$ through gradient descent is controlled by the second-order properties of $E(\mathbf{W})$, as represented by its Hessian matrix $\mathbf{H}$. We now consider a simple model which can be investigated analytically. Consider the case of an N-dimensional input vector feeding onto a single output unit with a linear activation function $f(a) = a$. The output corresponding to input $\mathbf{X}^\mu$ is given by

$$\mathcal{O}^\mu = \sum_{i=1}^{N} w_i x_i^\mu = \mathbf{W}^T \mathbf{X}^\mu, \quad (3)$$

where x_i^μ is the ith component of the μth input vector, and w_i is the coupling from the ith input unit to the output. It is assumed that the input elements $\{x_i^\mu\}$ are drawn randomly and independently from a distribution with mean m and variance v.

© 1991 The American Physical Society

The dynamic rule for weight updates,

$$\mathbf{W}(k+1)=\mathbf{W}(k)-\frac{\eta}{p}\sum_{\mu=1}^{p}(\mathcal{O}^{\mu}-d^{\mu})\mathbf{X}^{\mu}, \tag{4}$$

follows from the gradient of the cost function

$$E(\mathbf{W})=\frac{1}{2p}\sum_{\mu=1}^{p}(d^{\mu}-\mathcal{O}^{\mu})^{2}=\frac{1}{2p}\sum_{\mu=1}^{p}(d^{\mu}-\mathbf{W}^{T}\mathbf{X}^{\mu})^{2}. \tag{5}$$

A complete treatment of the convergence properties of such a linear neuron is available in the literature on adaptive filters for the case of an infinite number p of training examples.[5] But in most learning problems the training data are limited; we are thus interested in the case of finite p. The size of the training set is quantified by the ratio $\alpha=p/N$ between the number of examples and the dimensionality of the input vector. Calculations are performed in the $N\to\infty$ limit while keeping α constant and finite.

Note that the cost function of Eq. (5) is quadratic in $\mathbf{W}$, and can be rewritten as

$$E(\mathbf{W})=\tfrac{1}{2}(\mathbf{W}^{T}\underline{\mathbf{R}}\mathbf{W}-2\mathbf{Q}^{T}\mathbf{W}+C), \tag{6}$$

where $\underline{\mathbf{R}}$ is the covariance matrix of the input, $R_{ij}=(1/p)\times\sum_{\mu=1}^{p}x_i^{\mu}x_j^{\mu}$, a symmetric and non-negative $N\times N$ matrix; the N-dimensional vector $\mathbf{Q}$ is $Q_i=(1/p)\times\sum_{\mu=1}^{p}d^{\mu}x_i^{\mu}$, and the constant $C=(1/p)\sum_{\mu=1}^{p}(d^{\mu})^{2}$. Note that $\underline{\mathbf{R}}=\underline{\mathbf{H}}$, the Hessian matrix of $E(\mathbf{W})$. The solution space of vectors $\mathbf{W}^{*}$ which minimize $E(\mathbf{W})$ is the subspace of solutions of the linear equation

$$\underline{\mathbf{R}}\mathbf{W}=\mathbf{Q}. \tag{7}$$

This subspace reduces to a point if $\underline{\mathbf{R}}$ is full rank. The diagonalization of $\underline{\mathbf{R}}$ provides a diagonal matrix $\underline{\Delta}$ formed by its eigenvalues and a matrix $\underline{\mathbf{U}}$ formed by its eigenvectors. The coordinate transformation $\mathbf{V}=\underline{\mathbf{U}}(\mathbf{W}-\mathbf{W}^{*})$ yields

$$E(\mathbf{V})=\tfrac{1}{2}\mathbf{V}^{T}\underline{\Delta}\mathbf{V}+E_0, \tag{8}$$

with $E_0=E(\mathbf{W}^{*})$. In the new coordinate system the rule for weight updates becomes

$$\mathbf{V}(k+1)=\mathbf{V}(k)-\eta\underline{\Delta}\mathbf{V}(k), \tag{9}$$

a set of N decoupled equations. The dynamics in this diagonalized space is fully controlled by the matrix $\underline{\Delta}$, and thus by the eigenvalues of the covariance matrix $\underline{\mathbf{R}}$.

It is precisely the spectrum $\rho(\lambda)$ of eigenvalues of the covariance matrix $\underline{\mathbf{R}}$ that has been computed here. Since $\underline{\mathbf{R}}$ is a non-negative matrix, all eigenvalues satisfy $\lambda\geq 0$. If, as stated above, the input elements $\{x_i^{\mu}\}$ are independently drawn from a distribution with mean m and variance v, the spectrum exhibits three dominant features:[6] (a) A singular contribution at $\lambda=0$ with weight $1-a$ for $a<1$. (b) A continuous part of the spectrum within a bounded interval $\lambda_{-}<\lambda<\lambda_{+}$. The bounds are well defined and of order 1. For $a\neq 1$, $\lambda_{-}>0$, and there is a gap at the lower end of the spectrum. (c) One eigenvalue of order N, λ_N, present in the case of biased inputs ($m\neq 0$).

The continuous part (b) of the spectrum collapses onto a δ function at $\lambda=v$ as $p\to\infty$. True correlations between pairs of input components (x_i,x_j) might lead to a quite different spectrum from the one described above.

The dynamic equation (9) indicates that $\mathbf{V}$ will converge to zero (and thus $\mathbf{W}$ to the solution $\mathbf{W}^{*}$) provided that $0<\eta<2/\lambda_{\max}$. The slowest time constant in the system is given by $(\eta\lambda_{\min})^{-1}$, where $\lambda_{\min}$ is the smallest nonzero eigenvalue. The optimal choice $\eta=1/\lambda_{\max}$ leads to a slowest time constant given by the ratio $\lambda_{\max}/\lambda_{\min}$, and a *learning time* $\tau=a\lambda_{\max}/\lambda_{\min}$ proportional to the number of examples in the training set.

The spectrum $\rho(\lambda)$ also yields the average learning time $\langle 1/\lambda\rangle$, as well as information about the final entropy of the trained network. Further discussion of the full impact and implications of these results is briefly postponed to provide an outline of the calculation that produced them. Full details of the calculation will be given elsewhere.[7]

Results (a) and (b) for the spectral density follow from using a standard Fresnel representation for the determinant of a symmetric matrix $\underline{\mathbf{R}}$.[8] In combination with the identity

$$\rho(\lambda)=-\frac{2}{N\pi}\mathrm{Im}\frac{\partial}{\partial\lambda}\lim_{n\to 0}\frac{1}{n}\{[\det^{-1/2}(\underline{1}\lambda-\underline{\mathbf{R}})]^{n}-1\}, \tag{10}$$

the Fresnel representation yields

$$\rho(\lambda)=-\frac{2}{N\pi}\mathrm{Im}\frac{\partial}{\partial\lambda}\lim_{n\to 0}\frac{1}{n}\left\{\left(\frac{e^{i\pi/4}}{\sqrt{\pi}}\right)^{Nn}\int_{-\infty}^{\infty}\prod_{k}^{N}dy_k\exp\left[-i\sum_{k,l}^{N}\sum_{\gamma}^{n}y_k^{\gamma}(\lambda\delta_{kl}-R_{kl})y_l^{\gamma}\right]-1\right\}. \tag{11}$$

The expression in curly brackets in Eq. (11) can be written as

$$\{\cdots\}=\int\prod_{\gamma\beta}^{n}\frac{dq_{\gamma\beta}d\varphi_{\gamma\beta}}{2\pi}\int\prod_{\gamma}^{n}\frac{dM_{\gamma}}{2\pi}\exp\left[N\left\{i\sum_{\gamma\beta}\varphi_{\gamma\beta}q_{\gamma\beta}\ln\left[\int\prod_{\gamma}^{n}dy_{\gamma}\exp\left(-i\lambda\sum_{\gamma}y_{\gamma}^{2}-i\sum_{\gamma\beta}\varphi_{\gamma\beta}y_{\gamma}y_{\beta}\right)\right]\right.\right.$$
$$\left.\left.+a\ln\left[\int\prod_{\gamma}^{n}\frac{dt_{\gamma}}{2\pi}\exp\left(-\tfrac{1}{2}\sum_{\gamma}t_{\gamma}^{2}+\sqrt{2i}m\sum_{\gamma}t_{\gamma}M_{\gamma}+iv\sum_{\gamma\beta}t_{\gamma}t_{\beta}q_{\gamma\beta}\right)\right]\right\}\right], \tag{12}$$

and can be evaluated in the $N\to\infty$ thermodynamic limit using a saddle-point method.

A replica-symmetric solution,[9] with $q_{\gamma\beta}=q_0$ and $\varphi_{\gamma\beta}=\varphi_0$ for $\gamma=\beta$, $q_{\gamma\beta}=q_1$ and $\varphi_{\gamma\beta}=\varphi_1$ for $\gamma\neq\beta$, and $M_\gamma=M$, yields the main result

$$\rho(\lambda)=\begin{cases}(1-\alpha)\Theta(1-\alpha)\delta(\lambda), & \lambda=0,\\ \{4\alpha v^2-[\lambda\alpha-v(1+\alpha)]^2\}^{1/2}/2\pi\lambda v, & \lambda_-\le\lambda\le\lambda_+,\end{cases} \tag{13}$$

with $\lambda_-=[(1-\sqrt{\alpha})^2/\alpha]v$ and $\lambda_+=[(1+\sqrt{\alpha})^2/\alpha]v$. Averages $\langle\cdots\rangle\equiv\int d\lambda\rho(\lambda)\cdots$ over the eigenvalue distribution (13) can be easily computed to obtain $\langle\lambda\rangle=v$, $\lambda(\rho_{\max})=v(1-\alpha)^2/(1+\alpha)$, and $\langle 1/\lambda\rangle=(1+\alpha)^2/4v(\alpha-1)$ for $\alpha>1$. Note that the results are heavily controlled by the variance v of the distribution from which the inputs $\{x_i^\mu\}$ are drawn. Also, $\langle\lambda\rangle^{-1}\neq\langle 1/\lambda\rangle$ and $\langle\lambda\rangle\neq\lambda(\rho_{\max})$.

The stability of the replica-symmetric solution of Eq. (13) is difficult to establish in general. In order to assess such stability as well as the magnitude of finite-size effects, we have numerically investigated systems with $N\le 200$ for various values of α. Results of the simulations for $N=200$, shown in Fig. 1, indicate that corrections due to replica-symmetry breaking[10] are negligible, and that finite-size effects are unimportant: The distribution $\rho(\lambda)$ exhibits no long tails, and is well bounded within the predicted values of λ_- and λ_+, even for such small systems.

For $m=0$, $\lambda_{\max}=\lambda_+$ and $\lambda_{\min}=\lambda_-$. The learning time $\tau=\alpha\lambda_{\max}/\lambda_{\min}=\alpha\lambda_+/\lambda_-$ can be easily computed using Eq. (13): $\tau=\alpha(1+\sqrt{\alpha})^2/(1-\sqrt{\alpha})^2$. As a function of α, τ diverges at $\alpha=1$, and, surprisingly, goes through a minimum at $\alpha=(1+\sqrt{2})^2=5.83$ before diverging linearly for $\alpha\to\infty$. Numerical simulations were performed to estimate τ by counting the number T of presentations of training examples needed to reach an allowed error level $\tilde{E}$ through gradient descent. If the prescribed error $\tilde{E}$ is sufficiently close to the minimum error E_0, T is controlled by the slowest mode, and it provides a good estimate for τ. Numerical results for T as a function of α, shown in Fig. 2, were obtained by training a single linear neuron on randomly generated vectors. As predicted, the curve exhibits a clear maximum at $\alpha=1$, as well as a minimum between $\alpha=4$ and 5.

For $m\neq 0$, $\lambda_{\max}=\lambda_N$, an eigenvalue proportional to N. Then $\tau=\alpha\lambda_N/\lambda_-$, much larger than in the $m=0$ case. The eigenvalue λ_N arises because in the thermodynamic limit the off-diagonal elements of $\mathbf{R}$ are equal to m^2 and the diagonal elements are equal to $v+m^2$. The eigenvector $\mathbf{u}_N=(1,\ldots,1)$ thus corresponds to the eigenvalue $\lambda_N=Nm^2+v$. Since $\mathrm{Tr}\mathbf{R}=N(m^2+v)$ and $\langle\lambda\rangle=v$, λ_N is the only eigenvalue larger than λ_+. The large part of λ_N is eliminated for centered distributions with $m=0$, such as $x_i^\mu=\pm 1$ with probability $\frac{1}{2}$, or $x_i^\mu=3,-1,-2$ with probability $\frac{1}{3}$. Note that although m plays a crucial role in controlling the existence of an isolated eigenvalue of order N; it plays no role in the spectral density of Eq. (13).

The existence of only *one* eigenvector of order N, while the remaining $N-1$ eigenvalues are of order 1, can be also established for the case in which the mean of

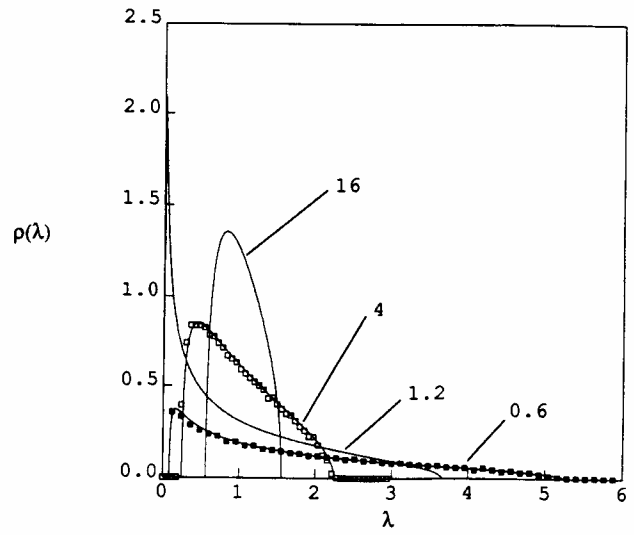

FIG. 1. Spectral density $\rho(\lambda)$ predicted by Eq. (13) for $m=0$, $v=1$, and $\alpha=0.6$, 1.2, 4, and 16. Experimental histograms for $\alpha=0.6$ (solid squares) and $\alpha=4$ (open squares) are averages over 100 trials with $N=200$ and $x_i^\mu=\pm 1$ with probability $\frac{1}{2}$ each.

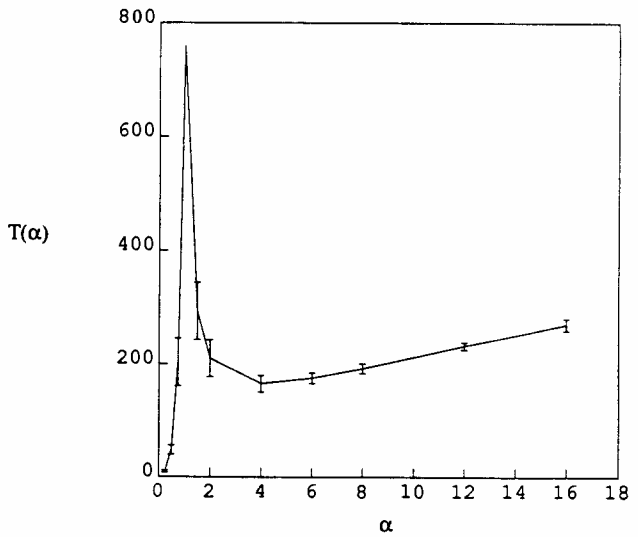

FIG. 2. Number of iterations T (averaged over 20 trials) needed to train a linear neuron with $N=100$ inputs. The x_i^μ are uniformly distributed between -1 and $+1$. Initial and target couplings $\mathbf{W}$ are chosen randomly from a uniform distribution within the $[-1,+1]^N$ hypercube Gradient descent, with $\eta=1/2\lambda_{\max}$, is considered complete when the error reaches the prescribed value $\tilde{E}=0.001$ above the $E_0=0$ minimum value.

the distribution is not uniform but position dependent: m is replaced by $\{m_i\}$ $(i=1,\ldots,N)$. For such nonuniformly biased inputs with $m_i \neq 0$ there is only one eigenvalue λ_N of order N, and the components of the associated eigenvector $\mathbf{u}_N$ are given by $u_{iN} = m_i$.

Such information can be exploited to achieve considerable reduction in learning times. The goal is to minimize the ratio $\lambda_{max}/\lambda_{min}$. A simple approach is to center each input variable x_i by subtracting its mean m_i, thereby suppressing λ_N. Nonuniform variances which also cause a spread of the spectrum can be treated by a trivial rescaling.

An alternative approach is to use the available eigenvector to subtract or reduce the component of the gradient along the fast direction. The modified update rule

$$\mathbf{W}(k+1) = \mathbf{W}(k) - \eta \left[\nabla E + (c-1) \frac{\mathbf{u}_N \cdot \nabla E}{||\mathbf{u}_N||^2} \mathbf{u}_N \right] \quad (14)$$

interpolates between standard steepest descent for $c=1$ [for which Eq. (2) is recovered] and complete elimination of updates in the direction of $\mathbf{u}_N$ for $c=0$. The constant c controls the time constant in the direction of $\mathbf{u}_N$; its optimal value is that which makes relaxation along that direction comparable to that along the other directions, characterized by eigenvalues of order 1. The reasonable choice $c = \langle\lambda\rangle/\lambda_N$ provides an approximation to the Newton-Raphson algorithm, which cannot be applied in its exact form to multilayer networks due to lack of definite positivity in the Hessian matrix[11] as well as excessive storage and computation time requirements.

Several eigenvalues of order N may appear in the case of true correlations between pairs of input components:

$$\overline{x_i x_j} \neq \overline{x_i}\,\overline{x_j} \text{ for } i \neq j .$$

The approach of Eq. (14) can be easily generalized to deal with this case; it requires knowledge of the corresponding eigenvectors.

The extension of these results to multilayer networks rests on the observation that each neuron i receives state information $\{x_j\}$ from the $j \in S_i$ neurons that feed into it, and can be viewed as minimizing a local objective function E_i whose Hessian matrix involves the covariance matrix of such inputs. If all input variables are uncorrelated and have zero mean, no large eigenvalues will appear. But states with $\overline{x_j} = m \neq 0$ produce eigenvalues proportional to the number of input neurons N_i in the set S_i, resulting in slow convergence if the connectivity is large.

An obvious source of systematic bias m is the use of activation functions which restrict the state variables to the interval [0,1]. Symmetric activation functions such as the hyperbolic tangent are empirically known to yield faster convergence than their nonsymmetric counterparts such as the logistic function. Our results provide an explanation to this observation, and justify the empirical rule of choosing individual learning rates η_i inversely proportional to the number of inputs N_i to the ith neuron.

Our results are based on a rigorous calculation of the eigenvalue spectrum for a symmetric matrix constructed from the outer product of random vectors. Such spectral density provides a full description of the relaxation of a single adaptive linear unit, and illuminates various aspects of the dynamics of learning in multilayer networks composed of nonlinear units.

One of us (I.K.) thanks AT&T Bell Laboratories for their hospitality while this work was carried out. The authors thank John Denker, Larry Jackel, and Rich Howard for encouragement and helpful discussions.

[1] D. J. Amit, *Modelling Brain Function* (Cambridge Univ. Press, New York, 1989).

[2] A. Waibel, Neural Comput. **1**, 39 (1989).

[3] Y. Le Cun, B. Boser, J. S. Denker, D. Henderson, R. E. Howard, W. Hubbard, and L. D. Jackel, Neural Comput. **1**, 541 (1989).

[4] D. E. Rumelhart, G. E. Hinton, and R. J. Williams, Nature (London) **323**, 533 (1986).

[5] B. Widrow and S. D. Stearns, *Adaptive Signal Processing* (Prentice Hall, Englewood Cliffs, NJ, 1985).

[6] J. A. Hertz, G. I. Thorbergsson, and A. Krogh, J. Phys. A **22**, 2133 (1989), have obtained a Green's function from which results (a) and (b) can be extracted for the $x_i^\mu = \pm 1$ case. Also, F. Vallet, J.-G. Cailton, and Ph. Refregier, Europhys. Lett. **9**, 315 (1989), discuss, for the $x_i^\mu = \pm 1$ case, the α dependence of λ_- and of the standard deviation of the eigenvalue distribution.

[7] I. Kanter, Y. Le Cun, and S. A. Solla (to be published)

[8] S. F. Edwards and R. C. Jones, J. Phys. A **9**, 1595 (1976).

[9] D. Sherrington and S. Kirkpatrick, Phys. Rev. Lett. **35**, 1792 (1975).

[10] G. Parisi, Phys. Rev. Lett. **43**, 1754 (1979).

[11] S. Becker and Y. Le Cun, in *Proceedings of the 1988 Connectionist Models Summer School*, edited by D. Touretzky, G. Hinton, and T. Sejnowski (Morgan Kaufmann, San Mateo, CA, 1989), p. 29.