J. Phys. A: Math. Gen. 25 (1992) 1135–1147. Printed in the UK

Generalization in a linear perceptron in the presence of noise

Anders Krogh†‡ and John A Hertz§

† The Niels Bohr Institute, Blegdamsvej 17, DK-2100 Copenhagen, Denmark
§ Nordita, Blegdamsvej 17, DK-2100 Copenhagen, Denmark

Received 2 April 1991, in final form 4 November 1991

Abstract. We study the evolution of the generalization ability of a simple linear perceptron with N inputs which learns to imitate a 'teacher perceptron'. The system is trained on $p = \alpha N$ example inputs drawn from some distribution and the generalization ability is measured by the average agreement with the teacher on test examples drawn from the same distribution. The dynamics may be solved analytically and exhibits a phase transition from imperfect to perfect generalization at $\alpha = 1$, when there are no errors (static noise) in the training examples. If the examples are produced by an erroneous teacher, overfitting is observed, i.e. the generalization error starts to increase after a finite time of training. It is shown that a weight decay of the same size as the variance of the noise (errors) on the teacher improves on the generalization and suppresses the overfitting. The generalization error as a function of time is calculated numerically for various values of the parameters. Finally dynamic noise in the training is considered. White noise on the input corresponds on average to a weight decay, and can thus improve generalization, whereas white noise on the weights or the output degrades generalization. Generalization is particularly sensitive to noise on the weights (for $\alpha < 1$) where it makes the error constantly increase with time, but this effect is also shown to be damped by a weight decay. Weight noise and output noise acts similarly above the transition at $\alpha = 1$.

1. Introduction

It is very important in practical situations to know how well a neural network will generalize from the examples it is trained on to the entire set of possible inputs. This problem is the focus of much recent and current work. All this work, however, deals with the asymptotic state of the network after training. Here we study a very simple model which allows us to follow the evolution of the generalization ability in time under training. It has a single linear output unit, and the weights obey adaline learning. Despite its simplicity, it exhibits non-trivial behaviour: a dynamical phase transition at a critical number of training examples and overfitting if the training examples are corrupted by noise. Part of this work has already been reported [1].

Given some function $f(\xi)$ and a set of examples of the function, $\zeta^\mu = f(\xi^\mu)$, the ability to generalize can be measured by the average error on a random input pattern. The general concept of generalization has been studied by two very different methods. One is to use the theory of VC dimensions [2–4] to find bounds on the generalization ability, and the other is to use ideas from statistical physics to find

‡ Present address: Computer and Information Sciences, University of California, Santa Cruz, CA 95064, USA.

0305-4470/92/051135+13$04.50

the average generalization ability of a network [4–7]. Here we use a trick probably invented by Gardner and Derrida [8] but recently used by a number of authors [9–13]. The trick is to limit the set of possible functions f to those the network can actually implement, i.e. the network has to learn to imitate a teacher.

A perceptron with multiple units can always be analysed one unit at a time, so in the following we study just one linear unit or neuron with a response

$$V = N^{-1/2} \sum_i w_i \xi_i \tag{1}$$

to an input ξ_i on the ith input terminal. The unit is trained on p examples of input–output pairs (ξ_i^μ, ζ^μ) and the delta-rule learning equation [14] is then in continuous time

$$\dot{w}_i = N^{-1/2} \sum_{\mu=1}^{p} (\zeta^\mu - V^\mu)\xi_i^\mu - \lambda w_i. \tag{2}$$

This is the 'batch-learning' form of adaline learning. The last term is a simple weight decay that enables us to limit the size of the weights. This learning process has been studied in the presence of noise [14–16], and many of the results derived in this paper will draw on those results.

The teacher perceptron is characterized by a set of weights u_i, and the network is trained on p examples (ξ_i^μ, ζ^μ) with

$$\zeta^\mu = N^{-1/2} \sum_i u_i \xi_i^\mu \tag{3}$$

generated by the teacher. So we know that the perceptron can learn the task and the interesting questions concern how well it generalizes from a limited number of examples and what happens in the presence of noise.

Using these teacher-generated targets equation (2) becomes

$$\dot{w}_i = \frac{1}{N} \sum_{\mu j} (u_j - w_j)\xi_j^\mu \xi_i^\mu - \lambda w_i.$$

By introducing the difference between the teacher and the pupil, $v_i \equiv u_i - w_i$, and the training input correlation matrix

$$A_{ij} = \frac{1}{N} \sum_{\mu=1}^{p} \xi_j^\mu \xi_i^\mu \tag{4}$$

the learning equation reads

$$\dot{v}_i = \lambda u_i - \sum_j A_{ij} v_j - \lambda v_i. \tag{5}$$

Except for the first term this equation is identical to the original learning equation (2) studied in [14, 16].

We let the p example inputs ξ_i^μ be drawn randomly and independently according to some distribution $P(\xi)$ with mean 0 and variance 1. For a large number of examples ($p = O(N) \gg 1$), the resulting generalization ability will be independent of just which p of the possible input patterns we choose.

To measure the generalization ability, we average the squared error $(\zeta^\mu - V^\mu)^2$ over the distribution P of inputs. Thus we define the generalization error as

$$F = \frac{1}{N}\left[\int \left(\prod_i \mathrm{d}\sigma_i P(\sigma_i)\right)\left(\sum_i (u_i - w_i)\sigma_i\right)^2\right]_\xi = \frac{1}{N}\sum_i [v_i^2]_\xi \tag{6}$$

where it was used that P has variance one. $[]_\xi$ is an average over the different possible realizations of input patterns ξ_i^μ. That is, F is just proportional to the square of the difference between the teacher and pupil weight vectors. With the N^{-1} normalization factor F will then vary between 1 (*tabula rasa*) and 0 (perfect generalization) if we normalize u to length $\sqrt{N}$ and $w(t=0) = 0$. During learning, w_i and thus v_i depends on time, so F is a function of t. The complementary quantity $1 - F(t)$ could be called the generalization ability.

Although we center on generalization it will also be of interest to calculate the training error. It is defined as

$$E \equiv \left[\frac{1}{p}\sum_\mu (\zeta^\mu - V^\mu)^2\right]_\xi . \tag{7}$$

Using (3) it becomes

$$E = \left[\frac{1}{p}\sum_{ij} A_{ij} v_i v_j\right]_\xi . \tag{8}$$

2. Asymptotic solution

All the eigenvalues of **A** are non-negative (because it is a correlation matrix). Then for $\lambda > 0$ we can introduce the response function

$$\mathbf{g} = (\lambda \mathbf{I} + \mathbf{A})^{-1}. \tag{9}$$

The static solution (i.e. for $t \to \infty$) to equation (2) can then be written as

$$w_i = N^{-1/2}\sum_j g_{ij}\sum_\mu \xi_j^\mu \zeta^\mu . \tag{10}$$

For λ going to zero this solution approaches the pseudo-inverse [4, 17], while for large λ the response function (9) will be dominated by the λ-term and approach the Hebb solution [16]. The weight decay parameter is thus a nice way of interpolating between the two extremes of pseudo-inverse and Hebb synapses.

Similarly the asymptotic value of v_i is found from equation (5)

$$v_i = \sum_j g_{ij} \lambda u_j. \tag{11}$$

Most quantities of interest can be calculated from the average response function $[\mathbf{g}]_\xi$. It is diagonal in this case, and it has been calculated in [14],

$$G = [g_{ii}]_\xi = \frac{1 - \alpha - \lambda + \sqrt{(\lambda - z_+)(\lambda - z_-)}}{2\lambda}. \tag{12}$$

Here we have introduced the spectral limits

$$z_\pm = -(1 \pm \sqrt{\alpha})^2. \tag{13}$$

See [14, 16] for more details.

The subspace of the weight space spanned by the pattern vectors ξ^μ is called the pattern subspace. Any initial component of w outside this subspace is only acted upon by the λ-term in equation (2). If $\lambda = 0$ the initial component orthogonal to the pattern subspace is therefore unchanged by the learning. Thus the asymptotic solution will change: the part of $w(t = 0)$ in the orthogonal subspace should be added to the limit of (11) for $\lambda \to 0$. If $w(t = 0) = 0$ (*tabula rasa*) there is no such component and results derived from (11) holds even in the limit $\lambda \to 0$.

To find the asymptotic generalization error the size of $|v|^2$ is needed. We start directly from (11),

$$|v|^2 = \sum_{ijk} g_{ij} g_{ik} \lambda^2 u_j u_k.$$

The product **gg** was calculated in [14] (it follows easily from (9)),

$$\sum_i g_{ij} g_{ik} = -\frac{\partial g_{jk}}{\partial \lambda}. \tag{15}$$

When averaged over patterns the response function becomes diagonal, so

$$F = \frac{1}{N} [|v|^2]_\xi = \frac{\lambda^2}{N} \sum_{jk} u_j u_k \left[-\frac{\partial g_{jk}}{\partial \lambda} \right]_\xi = -\lambda^2 \frac{\partial G}{\partial \lambda}. \tag{16}$$

Here and later it is assumed that $|u| = \sqrt{N}$.

The derivative of G can be found directly from (12). For small λ it can be expanded, and in the $\lambda = 0$ limit

$$F = \begin{cases} 1 - \alpha & \text{for } \alpha \leqslant 1 \\ 0 & \text{for } \alpha > 1. \end{cases} \tag{17}$$

For $\lambda = 0$ and starting from *tabula rasa* the error falls of linearly from 1 to 0 as α goes from 0 to 1; i.e. there is a transition at $\alpha = 1$ from imperfect to perfect generalization. This behaviour can be understood in a very intuitive way. For fewer than N patterns there are not enough vectors in weight space to exactly specify the

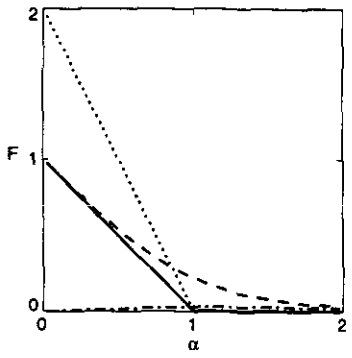

Figure 1. The asymptotic generalization error F as a function of α with no noise. Starting from *tabula rasa* with $\lambda = 0$ is shown by the full curve, starting from $|w(t = 0)| = \sqrt{N}$ by the dotted line, and learning with a weight decay by the broken curve. For $\lambda = 0.2$ the *training* error is shown by the chain curve. The asymptotic training error is zero for $\lambda = 0$.

teacher vector u. For $\alpha > 1$ the patterns span the whole space and completely specify u, and thus the generalization becomes perfect. For $\lambda > 0$ this curve is smoothed as shown in figure 1.

For $\lambda = 0$ and $\alpha \leqslant 1$ the calculation holds only if starting from *tabula rasa*; if $w(t = 0) \neq 0$ the part orthogonal to the pattern subspace will stay untouched and therefore contribute to the error. The contribution can be found by a heuristic argument. If $w(t = 0)$ is zero v is (for $t \to \infty$) equal to the part of u orthogonal to the pattern subspace, $u^{\perp}$. If $w(t = 0)$ is non-zero the part that is in the orthogonal subspace $w^{\perp}(t = 0)$ will also contribute, so $|v|^2 = (u^{\perp} - w^{\perp}(t = 0))^2$. Averaging over patterns and random initial weights will then give $|v|^2 = (1 - \alpha)(|u|^2 + |w(t = 0)|^2)$, because the patterns only 'cover' a fraction α of the weight space, so

$$F = F_{tabula\ rasa} + (1 - \alpha)|w(t = 0)|^2/N. \tag{18}$$

See [1] for a more careful treatment of this case.

Each eigenvalue A_r of $\hat{\mathbf{A}}$ (4) corresponds to an exponentially decaying mode of the system with a relaxation time of $1/(\lambda + A_r)$. Therefore the relaxation times of the learning process depend only on $\mathbf{A}$, which depends on the input patterns. Thus the relaxation times are the same as for random input/output patterns, which have been studied previously [14, 16]. For $\lambda = 0$ the relaxation time diverges when α approaches 1 from either side.

3. Learning with an unreliable teacher

Two kinds of noise will be studied in this paper: random errors in the training set which we will also call *static noise* because it is constant during learning, and time-dependent noise in the learning process, which is called dynamic noise. Here we consider the first kind.

An unreliable teacher is, in our terminology, one that supplies the pupil perceptron with erroneous targets. It is modelled by adding *static* noise to the teacher. This can be done in (at least) three different ways: directly to the targets ζ^{μ}, to the teacher weights, or to the input patterns ξ_i^{μ} before the targets are produced.

This section deals with the asymptotic generalization error; the error as a function of time will be studied later.

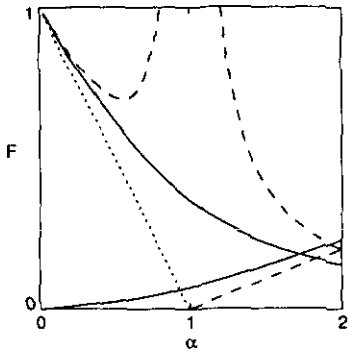

Figure 2. Generalization and training error with static noise of size $\sigma^2 = 0.2$ on the input or the output of the teacher. Curves starting in the upper left corner represent the generalization error F while the lower ones represent the training error. The full curve is for $\lambda = \sigma^2$, and the broken curve $\lambda = 0$. The dotted line is the generalization error with no noise and $\lambda = 0$ for reference.

3.1. Noise on input or output

If noise f^μ with mean zero and variance σ^2_{out} is added to the outputs of the teacher, the learning equation (5) changes to

$$\dot{v}_i = \left(\lambda u_i - N^{-1/2}\sum_\mu \xi_i^\mu f^\mu\right) - \sum_j A_{ij} v_j - \lambda v_i \tag{19}$$

and the asymptotic solution is then by analogy with (11)

$$v_i = \sum_j g_{ij}(\lambda u_j - N^{-1/2}\sum_\mu \xi_j^\mu f^\mu). \tag{20}$$

Squaring and averaging over the static noise and using (15) gives

$$|v|^2 = \sum_{ijk} g_{ij} g_{ik}[\lambda^2 u_j u_k + \overline{(f^\mu)^2} A_{jk}] \tag{21}$$

where the bar means averaging over the static noise. Using (15) and averaging over the patterns gives

$$F = \sigma^2_{\text{out}} \frac{\partial}{\partial\lambda}(\lambda G) - \lambda^2 \frac{\partial G}{\partial\lambda}. \tag{22}$$

We observe that the last term is identical to the error in the noise-free case. The noise simply adds a (positive) term to the noise-free error. For $\lambda \to 0$ the derivative of λG goes to $(1 + \alpha - |1 - \alpha|)/2|1 - \alpha|$, so in this limit the generalization error is

$$F = \begin{cases} 1 - \alpha + \dfrac{\alpha\sigma^2_{\text{out}}}{1-\alpha} & \text{for } \alpha \leqslant 1 \\ \dfrac{\sigma^2_{\text{out}}}{\alpha - 1} & \text{for } \alpha > 1. \end{cases} \tag{23}$$

Differentiating F with respect to λ we find that F has a minimum for $\lambda = \sigma^2_{\text{out}}$. Thus, it pays to learn with a weight decay. At this minimum point the error is

$$F = \sigma^2_{\text{out}}\, G|_{\lambda=\sigma^2_{\text{out}}} \tag{24}$$

which is compared in figure 2 to the error without the weight decay.

Calculating the training error yields

$$\alpha E = (\alpha - 1)\sigma_{\text{out}}^2 + (\lambda + \sigma_{\text{out}}^2)\lambda G - \lambda F.$$

This is plotted in figure 2. For $\lambda \to 0$ it becomes

$$E = \begin{cases} 0 & \text{for } \alpha \leqslant 1 \\ \sigma_{\text{out}}^2(\alpha - 1) & \text{for } \alpha > 1 \end{cases}. \tag{26}$$

For $\alpha \leqslant 1$ the learning is perfect, whereas the error grows linearly with α above 1. This behaviour was also found for random targets, see [16]. Interestingly the generalization error decreases with α while the learning error increases in this limit. (To compare the learning and generalization errors the first should really be divided by α to give the training error per pattern. But it is still true that the training error increases, and for large α the generalization error becomes smaller than the training error).

Noise on the inputs can be treated in a similar manner, and it turns out that the result is basically the same.

3.2. Noise on the weights

Noise on the teachers weights is in a sense qualitatively different from noise added to the input or the output as studied earlier. This is because the training set resulting from the noisy weights can still be learned *perfectly* by the pupil—w just has to converge to the noisy weights. For input or output noise the pupil cannot learn the patterns perfectly for $\alpha > 1$. If noise η_i with mean zero and a variance σ_w^2 is added to the teacher weights the targets for training are then corrupted according to

$$\zeta^\mu = N^{-1/2}\sum_i (u_i + \eta_i)\xi_i^\mu = N^{-1/2}\sum_i u_i\xi_i^\mu + N^{-1/2}\sum_i \eta_i\xi_i^\mu. \tag{27}$$

As usual this leads to a solution at infinite time,

$$v_i = \sum_j g_{ij}\left(\lambda u_j - \frac{1}{N}\sum_{\mu,k}\xi_j^\mu\xi_k^\mu\eta_k\right) = \sum_j g_{ij}(\lambda u_j - \sum_k A_{jk}\eta_k). \tag{28}$$

Squaring and averaging over the static weight noise gives

$$|v|^2 = \sum_{jj'} g_{ij}g_{ij'}[\lambda^2 u_j u_{j'} + \sum_{kk'} A_{jk}A_{j'k'}\overline{\eta_k\eta_{k'}}] \tag{29}$$

which is a little different from the previous case. The last term can be reduced using $\mathbf{AAg} = \mathbf{A} - \lambda\mathbf{Ag} = \mathbf{A} - \lambda\mathbf{I} + \lambda^2\mathbf{g}$, leading to

$$F = \sigma_w^2\left(1 - \frac{\partial\lambda^2 G}{\partial\lambda}\right) - \lambda^2\frac{\partial G}{\partial\lambda}. \tag{30}$$

This also has a minimum for a finite λ, but less pronounced, and the optimal value of λ does not have such a nice solution as for the output noise. In the $\lambda \to 0$ limit,

$$F = \begin{cases} 1 - \alpha + \sigma_w^2\alpha & \text{for } \alpha \leqslant 1 \\ \sigma_w^2 & \text{for } \alpha > 1. \end{cases} \tag{31}$$

The training error will in this case go to zero for $\lambda = 0$ as mentioned before. It will actually behave exactly as in the noise-free case except that σ_w^2 has to be added to $|u|^2$ whenever it occurs, because from the pupil's point of view the training set is just produced by a teacher with weights $u_i + \eta_i$.

4. Generalization error as a function of time

Until now we have only considered the asymptotic behaviour of the network. In this section we calculate the generalization error as a function of time.

For the two kinds of *static* noise the learning equation can be written as $\dot{v}_i = B_i - \sum_j A_{ij} v_j - \lambda v_i$. The particular form of B_i can be found in (19) and (28). In the basis where **A** is diagonal the general solution to this equation is

$$v_r(t) = (1 - e^{-(\lambda + A_r)t}) \frac{B_r}{\lambda + A_r} + v_r(0) e^{-(\lambda + A_r)t} \tag{32}$$

where the subscript r indicates transformation to this basis, and A_r are the eigenvalues of **A**. If the learning starts at *tabula rasa* ($v_r(0) = u_r$) the square of v_r is

$$v_r^2 = \frac{(1 - e^{-(\lambda + A_r)t})^2 B_r^2}{(\lambda + A_r)^2} + u_r^2 e^{-2(\lambda + A_r)t} + e^{-(\lambda + A_r)t}(1 - e^{-(\lambda + A_r)t}) \frac{2u_r B_r}{\lambda + A_r}. \tag{33}$$

Now we will average over the noise included in B_r. First of all the average of the B_r in the last term is just λu_r for both kinds of static noise. The term B_r^2 contributes a $\lambda^2 u_r^2$ plus something that depends on exactly which kind of noise we are dealing with. For static input noise we find

$$\overline{B_r^2} = \overline{\left(\lambda u_r - \sum_i S_{ri} N^{-1/2} \sum_\mu \xi_i^\mu f^\mu \right)^2} = \lambda^2 u_r^2 + \sigma_{\text{out}}^2 A_r \tag{34}$$

where the orthogonal transformation matrix S_{ri} that diagonalizes **A** was introduced.

The average error as a function of time is

$$F(t) = \frac{1}{N} \sum_r \int \mathrm{d}A_r \rho(A_r) v_r^2(t) \tag{35}$$

where $\rho(x)$ is the density of eigenvalues of **A** for large N. It can be found from the response function as shown in [16]:

$$\rho(x) = (1 - \alpha)\theta(1 - \alpha)\delta(x) + \frac{\sqrt{-(x + z_+)(x + z_-)}}{2\pi x} \tag{36}$$

where $\theta(\,)$ is the unit step function. The second term contributes only when it is real, i.e. when x lies between the roots $-z_+$ and $-z_-$.

It is obvious that we will have an exponential behaviour in the long-time limit if $\lambda > 0$. If $\lambda = 0$ the smallest non-zero eigenvalue will determine the long-time behaviour, but at $\alpha = 1$ the eigenvalue density will extend all the way down to zero, which will lead to a non-exponential behaviour. We first take a look at this situation. At $\alpha = 1$ and $\lambda = 0$ the eigenvalue density is $\sqrt{(4 - x)/x}$. In that limit and no noise ($\sigma^2 = 0$) the generalization error then becomes

$$F(t) = \frac{1}{2\pi} \int_0^4 \mathrm{d}x \sqrt{\frac{4 - x}{x}} e^{-2xt} = \frac{1}{2\pi t} \int_0^{4t} \mathrm{d}y e^{-2y} \sqrt{\frac{4t}{y} - 1} \underset{t \to \infty}{\longrightarrow} \frac{1}{\sqrt{2\pi t}}. \tag{37}$$

It approaches 0 as $t^{-1/2}$, i.e. a lot slower than exponential.

Now, what happens if the noise is *not* zero? The generalization error (35) has a minimum at finite t for some values of λ. This can easily be seen from differentiation of F with respect to t. Differentiation of the integrand of (35) (call it $F(x,t)$) gives

$$\frac{\partial}{\partial t}F(x,t) = \frac{2x\mathrm{e}^{-(\lambda+x)t}}{\lambda+x}[(\sigma_{\mathrm{out}}^2-\lambda)-(x+\sigma_{\mathrm{out}}^2)\mathrm{e}^{-(\lambda+x)t}]. \tag{38}$$

For $\lambda \geqslant \sigma_{\mathrm{out}}^2$ this is always negative, so F is decreasing with time, whereas it is positive for sufficiently large t if $\lambda < \sigma_{\mathrm{out}}^2$, meaning that F has a minimum for finite t. Note that the critical λ where this crossover appears is the one giving the smallest asymptotic error as calculated earlier.

Formally putting this derivative equal to zero leads to an x-dependent time,

$$\mathrm{e}^{-(\lambda+x)t_x} = (\sigma_{\mathrm{out}}^2-\lambda)/(\sigma_{\mathrm{out}}^2+x). \tag{39}$$

So t_x is the point where $F(t,x)$ has minimum. Therefore the minimal value of the error F_{min} must obey an inequality

$$F_{\mathrm{min}} \geqslant \int \mathrm{d}x\,\rho(x)F(x,t_x) = \int \mathrm{d}x\rho(x)\frac{\sigma_{\mathrm{out}}^2}{\sigma_{\mathrm{out}}^2+x}. \tag{40}$$

(This expression requires a few lines of algebra.) The value of F at infinite time for $\lambda = \sigma_{\mathrm{out}}^2$ is obtained directly from (35), and it turns out to be identical to the last line above. That is the lowest possible value of F is obtained at infinite time with $\lambda = \sigma_{\mathrm{out}}^2$:

$$F_\lambda(t) \geqslant F_{\lambda=\sigma_{\mathrm{out}}^2}(t\to\infty) \tag{41}$$

for any value of λ, including 0.

In figure 3 generalization curves are shown for $\alpha = 0.8$ without noise, with noise of size $\sigma_{\mathrm{out}}^2 = 0.2$, and with noise and the optimal weight decay.

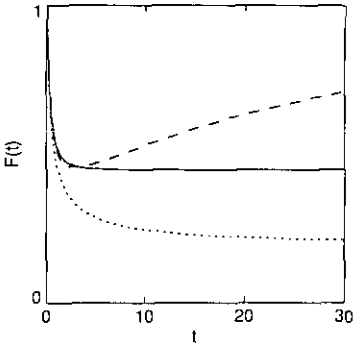

Figure 3. Generalization error as a function of time for $\alpha = 0.8$. The error when noise of size $\sigma_{\mathrm{out}}^2 = 0.2$ on the input or the output of the teacher is present is shown by the full curve ($\lambda = \sigma_{\mathrm{out}}^2$) and the dashed curve ($\lambda = 0$). Without weight decay the error starts to increase after a finite time of learning; that is called overfitting. The dotted curve is the generalization error with no noise and $\lambda = 0$.

It is important to note that the learning always decreases the *learning* error, because it is a gradient descent procedure on a cost function. It is a smooth monotonically decreasing curve with the asymptotic value already calculated in the previous section. Here we see that while the learning error keeps decreasing the generalization error starts to increase after some time of training. This behaviour is usually referred to as 'overfitting': the network starts to learn the irrelevant details of the noise, and that spoils the generalization.

The same calculation of the generalization error can be done for noise on the teacher's weights. The only change is in the B_r^2 term, which now becomes

$$\overline{B_r^2} = \lambda^2 u_r^2 + \sigma_w^2 A_r^2 \tag{42}$$

instead of (34). This change means that $\sigma_{\text{out}}^2 x$ in equation (35) is changed to $\sigma_w^2 x^2$. In this case there is no sign of overfitting.

5. Dynamic noise in the learning process

Another type of noise is dynamic noise in the learning process. This can, analogously to the static noise in the patterns, come in three different ways, namely noise on the input, weights and output. The effect of this kind of noise on the learning was studied in detail in [14, 16], and we will here see how it affects generalization.

5.1. Input noise

Consider noise $\epsilon_i^\mu(t)$ added to the input patterns. It is assumed that the variance is

$$\langle \epsilon_i^\mu(t)\epsilon_j^\nu(t')\rangle_\gamma = \gamma\delta_{ij}\delta_{\mu\nu}\exp(-\Delta|t-t'|) \tag{43}$$

where γ is the strength of the noise. In [16] it was shown that if Δ is very large this noise acts on average as a weight decay of size $\lambda = \gamma\alpha$. This kind of noise can then have a beneficial effect on generalization. This has been found in nonlinear networks as well [10, 18, 19].

5.2. Weight noise

Consider white noise $\eta_i(t)$ of variance

$$\langle \eta_i(t)\eta_j(t')\rangle_T = 2T\delta_{ij}\delta(t-t') \tag{44}$$

added to the learning equation (5).

The autocorrelation function is defined as

$$C = \frac{1}{N}\sum_i \left[\langle v_i^2\rangle_T - \langle v_i\rangle_T^2\right]_\xi . \tag{45}$$

Then the average generalization error can be written as

$$F_T = \frac{1}{N}\left[\langle |\boldsymbol{v}|^2\rangle_T\right]_\xi = C + \frac{1}{N}\sum_i \left[\langle v_i\rangle_T^2\right]_\xi . \tag{46}$$

The last part is just the same as before, because if the learning equation is averaged the noise disappears. Then

$$F_T = F_{T=0} + C. \tag{47}$$

For this kind of weight noise we have previously proven [16] the fluctuation-response relation $C = TG$, so

$$F_T = F_{T=0} + TG. \tag{48}$$

This equation shows that the asymptotic generalization error increases linearly with the noise level for $\lambda > 0$. For $\alpha \leqslant 1$ G (see (12)) and thus F diverges as λ goes to 0. This simply implies that the noise makes the weight vector perform a random walk in the subspace orthogonal to the pattern subspace because for $\lambda = 0$ there will be no damping forces. Therefore $|v|^2$ and thus F will keep increasing with time.

Above $\alpha = 1$ the response function is $G = (\alpha - 1)^{-1}$ in the limit of $\lambda = 0$, which can be found by expanding (12) for small λ. The T-dependent term in F, called δF, is then

$$\delta F = TG = \frac{T}{\alpha - 1}. \tag{49}$$

This diverges as $\alpha \to 1^+$. In the other extreme, $\alpha \to \infty$, it follows T/α, as has also been found for nonlinear networks [5–7, 9–12, 20, 22].

5.3. Output noise

If instead noise $f^\mu(t)$ is added to the output or targets, the result is an effective noise $\hat{\eta}_i(t)$ of variance

$$\langle \hat{\eta}_i(t)\hat{\eta}_j(t')\rangle_T = A_{ij}\langle f^\mu(t) f^\mu(t')\rangle_T = 2TA_{ij}\delta(t - t'). \tag{50}$$

The same analysis can now be carried through with the only change that the fluctuation-response relation $C = T(1 - \lambda G)$ (see [15, 16]).

This leads to the generalization error

$$F_T = F_{T=0} + T(1 - \lambda G). \tag{51}$$

The term λG does not diverge for $\lambda \to 0$, as seen directly from (12). This noise only acts in the pattern subspace, and this is the reason there is no divergence for $\lambda = 0$, contrary to ordinary weight noise which acts in the whole weight space. For any parameters we see a linear increase in the error with T.

6. Conclusion

In this simple network we have found many interesting properties that can also be found in nonlinear networks. Without noise the pupil perceptron can learn to imitate the teacher within an arbitrarily small margin for $\alpha > 1$, but close to $\alpha = 1$ the error decreases very slowly. For $\alpha < 1$ the asymptotic error decreases linearly as $\alpha = 1$ is approached. If there is noise in the training set the phenomenon of overfitting turns up; after a certain training time the generalization error of the pupil starts to increase while the training error still decreases.

It was shown that weight decay is beneficial to generalization in several cases:

(i) If there is (static) noise in the training set a weight decay leads to better generalization.

(ii) If there is noise on the weights and $\alpha < 1$ the weight vector performs a random walk in the subspace orthogonal to the pattern subspace which increases the generalization error with time, but this is suppressed by a weight decay.

(iii) If the network starts from a random initial weight vector different from zero the part of it in the orthogonal subspace will be left alone (for $\alpha < 1$), degrading the generalization unless a weight decay makes it decay.

White noise on the input has to a first approximation the same effect as a weight decay, and can thus improve generalization. This improvement of the generalization ability if a weight decay is used has been observed in layered nonlinear networks as well, see for instance [21].

Varying the weight decay parameter λ can be viewed as an interpolation between the pseudo-inverse solution ($\lambda = 0$) and the Hebbian solution (large λ) to the learning problem. For threshold perceptrons it has previously been shown that the Hebbian solution sometimes leads to better generalization than the pseudo-inverse [11, 22], and in [23] another way of interpolating between the two was proposed.

Many of these effects will to some extend carry over to nonlinear networks. Any network with differentiable activation functions behaves like a linear network close to the asymptotic state. More specifically for small differences between the network and its teacher the learning equation can be written like

$$\dot{v} = c - \mathcal{A}v \tag{52}$$

where c is a constant. If W is the total number of weights (or adjustable parameters) in the network $\mathcal{A}$ is a $W \times W$ matrix depending *only* on the input patterns. This implies that if the rank of $\mathcal{A}$ is less than W there is a part of weight space not 'covered' by the patterns, and e.g. a weight decay would improve the generalization. We speculate that at least for $p < W$ this would always happen.

References

[1] Krogh A and Hertz J A 1991 *Neural Information Processing Systems 3* ed Lippmann *et al* (San Mateo: Morgan Kaufmann) pp 897–903

[2] Abu-Mostafa Y S 1989 *Neural Comput.* **1** 312–7

[3] Baum E B and Haussler D 1989 *Neural Comput.* **1** 151–60

[4] Hertz J A, Krogh A and Palmer R G 1991 *Introduction to the Theory of Neural Computation* (Redwood City: Addison-Wesley)

[5] Schwartz D B, Samalam V K, Solla S A and Denker J S 1990 *Neural Comput.* **2** 371–82

[6] Tishby N, Levin E and Solla S A 1989 *Proc. IJCNN Washington 1989* vol 2 (Hillsdale, NJ: Erlbaum)

[7] Levin E, Tishby N and Solla S A 1990 *Preprint* AT&T Bell Labs

[8] Gardner E and Derrida B 1989 *J. Phys. A: Math. Gen.* **22** 1983–94

[9] Györgyi G 1990 *Phys. Rev. Lett.* **64** 2957–60

[10] Györgyi G and Tishby N 1990 *Neural Networks and Spin Glasses* ed W K Theumann and R Koeberle (Singapore: World Scientific)

[11] Opper M, Kinzel W, Kleinz J and Nehl R 1990 *J. Phys. A: Math. Gen.* **23** L581–6

[12] Sompolinsky H, Tishby N and Seung H S 1990 *Phys. Rev. Lett.* **65** 1683–6

[13] Opper M and Haussler D 1991 *Phys. Rev. Lett.* **66** 2677–80

[14] Hertz J A, Krogh A and Thorbergsson G I 1989 *J. Phys. A: Math. Gen.* **22** 2133–50

[15] Der R 1990 *J. Phys. A: Math. Gen.* **23** L763–6

[16] Krogh A 1992 *J. Phys. A: Math. Gen.* **25** 1119–33

[17] Kohonen T 1989 *Self-Organization and Associative Memory* (Berlin: Springer)
[18] Gardner E J, Stround N and Wallace D J 1989 *J. Phys. A: Math. Gen.* **22** 2019–30
[19] Wong K Y M and Sherrington D 1990 *J. Phys. A: Math. Gen.* **23** L175; 1990 *Statistical Mechanics of Neural Neworks* ed H Araki *et al* (Berlin: Springer)
[20] Hansel D and Sompolinsky H 1990 *Europhys. Lett.* **11** 687–92
[21] Hinton G E 1987 *PARLE: Parallel Architectures and Languages Europe (Lecture Notes in Computer Science)* ed G Goos and J Hartinanis (Berlin: Springer)
[22] Vallet F, Cailton J and Refregier P 1989 *Europhys. Lett.* **9** 315–20
[23] Refregier Ph and Vignolie J M 1989 *Europhys. Lett.* **10** 387–92