Maximum Entropy and Learning Theory

Griff L. Bilbro

Center for Communications and Signal Processing
Department of Electrical and Computer Engineering
North Carolina State University

TR-92/1
February 1992

Maximum Entropy and Learning Theory

Griff L. Bilbro
David E. Van den Bout
North Carolina State University, Box 7914
Raleigh, NC 27695-7914

March 22, 1991

Abstract

We derive the learning theory recently reported by Tishby, Levin, and Solla (TLS) without statistical mechanics. We show that the theory generally applies to any problem of modeling data. We analyze an elementary example for which we find the predictions consistent with intuition and conventional statistical results. We numerically examine the more realistic problem of training a competitive net to learn a one dimensional probability density from samples. We find TLS useful for predicting average training behavior.

1 Introduction

A statistical theory which describes the learning of a relation from examples was reported in (Tishby, Levin, and Solla, 1989). It built on earlier work in (Schwartz, Samalan, Solla, and Denker, 1990) and has been carefully restated in (Levin, Tishby, and Solla, 1990). In that literature, statistical mechanics was used to relate the probability of independent input-output data pairs to a layered neural network.

In this report we will show that the TLS theory can be understood without statistical mechanics; that the form of the data need not be limited to input-output pairs; and that the restriction to layered neural networks can be relaxed to include any model with adjustable parameters.

2 Maximum entropy and modeling

In this section we will show that a TLS theory can be constructed for any problem in which parameters of a model are chosen to fit data. Consider a problem in which data $\{x_i\}_{i=1}^{i=N}$ drawn from an unknown density $\overline{\rho}(x)$ are to be fitted with some model by adjusting parameters w to minimize an additive error function during a training procedure. In the case of training a layered feedforward net, the data are input-output pairs; the parameters w are the usual weights and biases; the error function and training procedure might be squared error and backpropagation. However the theory is much more general. In later sections, we will apply it to the simplest case of linear regression and to the case of learning a one dimensional probability density.

Except for the case of linear regression, there is little theoretical guidance for identifying when data is sufficient to determine the model. Often the total set of available data $\{x_i\}_{i=1}^{i=N}$ is divided into a training set $\{x_i\}_{i=1}^{i=m}$ and a remaining test set. The model is trained to a specified error ϵ_T on $\{x_i\}_{i=1}^{i=m}$ and then tested against the remaining data. In principle this procedure could be repeated for randomly selected test sets and initial values and the average results could be tabulated as functions of m and ϵ_T. Implicit in this hypothetical exercise is an average density $\langle\overline{\rho}^{(m)}(w)\rangle_x$ of nets trained to an error ϵ_T on m examples. The true $\langle\overline{\rho}^{(m)}(w)\rangle_x$ would be difficult to calculate in general, mostly because it involves the details of the training procedure. However the end result of training may depend more on the form of the model, the amount of the data, the noise in the data, and the training error ϵ_T, than on the details of the training procedure. In that case, the maximum entropy density $\langle\rho^{(m)}(w)\rangle_x$ resembles $\langle\overline{\rho}^{(m)}(w)\rangle_x$ and might be used to predict training behavior and generalization.

The *principle of maximum entropy* is a general inference tool which produces probabilities characterized by certain average values of specified functions(Jaynes, 1979). To the extent that entropy measures information, a maximum entropy estimate contains only the information implied by those average values and makes no other assumptions (*e.g.* about the training procedure). It is useful to also incorporate a prior estimate $\rho^{(0)}(w)$ of $\overline{\rho}^{(m)}(w)$ by considering the entropy of $\rho^{(m)}(w)$ relative to $\rho^{(0)}(w)$

$$R[\rho^{(m)}] = \int dw \rho^{(m)}(w) \ \ln \frac{\rho^{(m)}(w)}{\rho^{(0)}(w)}. \tag{1}$$

In our practice so little is known about $\overline{\rho}^{(m)}(w)$ that $\rho^{(0)}(w)$ is chosen merely as a restriction to reasonable portions of w space. For example, in back propagation it is unreasonable to expect the weights as large as $O(10^6)$ and perhaps $O(1)$ is a little too small. In any case the first test of this theory must be for sensitivity to $\rho^{(0)}(w)$.

In the maximum entropy sense, the density $\rho^{(m)}(w)$ that contains the least information beyond $\rho^{(0)}(w)$ but is nevertheless a normalized density with integral

$$\int dw\rho^{(m)}(w) \ = 1 \tag{2}$$

and with specific average training error

$$\langle \sum_{i=1}^{i=m} \epsilon(x_i, w) \rangle_w \ \equiv \int dw\rho^{(m)}(w) \sum_{i=1}^{i=m} \epsilon(x_i, w) \ = \epsilon_T, \tag{3}$$

can be found with calculus of variations and two Lagrange multipliers α and β in the usual way. The extremum satistifies

$$\ln \frac{\rho^{(m)}(w)}{\rho^{(0)}(w)} - 1 + \alpha + \beta \sum_{i=1}^{i=m} \epsilon(x_i, w) \ = 0 \tag{4}$$

which can be solved for $\rho^{(m)}(w)$ to get

$$\rho^{(m)}(w) \ = \frac{\rho^{(0)}(w)}{Z^{(m)}} \exp\left(-\beta \sum_{i=1}^{i=m} \epsilon(x_i, w)\right) \tag{5}$$

where β has been left, but $\exp(1-\alpha)$ has been evaluated in terms of the more conventional normalization

$$Z^{(m)} \ = \int dw\rho^{(0)}(w) \ \exp\left(-\beta \sum_{i=1}^{i=m} \epsilon(x_i, w)\right) \tag{6}$$

that depends on the particular set of examples as well as their number.

Equation 5 is significant. It is an estimate of the probability density of models of the form (*i.e.* architecture for a neural net) defined by the functional relation between w and x in $\epsilon(x, w)$ after being trained from random initial values of w to an average error Equation 3 on the particular data set $\{x_i\}_{i=1}^{i=m}$. However Equation 5 is still not useful as a predictor of average training behavior because it does depend on the particular examples in the

set $\{x_i\}_{i=1}^{i=m}$. In order to remove that effect, we average Equation 5 over all possible m examples

$$\langle \rho^{(m)}(w) \rangle_x = \int dx^{(m)} \overline{p}(x_1)\overline{p}(x_2)...\overline{p}(x_m)\rho^{(m)}(w). \tag{7}$$

Equation 7 can be used to define a performance criterion, the *average prediction fraction*

$$\phi^{(m)} = \int dw \int dx_{m+1}\overline{p}(x_{m+1}) \exp(-\beta\epsilon(x_{m+1}, w))\langle \rho^{(m)}(w) \rangle_x \tag{8}$$

which is the fraction of models distributed according to $\rho^{(m)}(w)$ which have an error function ϵ within about $1/\beta$ of the next unseen example x_{m+1} on average.

Equations 7 and 8 are inconvenient to evaluate exactly because of the $Z^{(m)}$ term of Equation 6 in their denominators. TLS propose an "annealed approximation" which in the present context is equivalent to replacing $Z^{(m)}$ in Equation 7 by its average over ways of choosing $\{x_i\}_{i=1}^{i=m}$

$$\langle Z^{(m)} \rangle_x = \int dw \int dx^{(m)} \overline{p}(x_1)\overline{p}(x_2)...\overline{p}(x_m)\rho^{(0)}(w) \exp(-\beta \textstyle\sum_{i=1}^{i=m} \epsilon(x_i, w)) \tag{9}$$

which can be written

$$\langle Z^{(m)} \rangle_x = \int dw \rho^{(0)}(w) f^m(w) \tag{10}$$

where

$$f(w) \equiv \int dx \overline{p}(x) \exp(-\beta\epsilon(x, w)). \tag{11}$$

With this the average prediction fraction of Equation 8 becomes

$$\phi^{(m)} = \frac{\int dw \rho^{(0)}(w) f^{m+1}(w)}{\int dw \rho^{(0)}(w) f^m(w)}. \tag{12}$$

Equation 12 predicts generalization behavior by drawing a maximum entropy inference of the average consistency between the model represented by $\epsilon(x, w)$ and m examples drawn from $\overline{p}(x)$.

We will show that Equation 12 is is well suited for theoretical analysis and is also convenient in practical numerical calculations for small problems. This is because it is easy to produce Monte Carlo estimates for the averages over the $\{x_i\}_{i=1}^{i=m}$ by using the entire set of available data $\{x_i\}_{i=1}^{i=N}$.

2.1 Relation to TLS

In this subsection we will show that under the assumptions of Tishby, Levin, and Solla, our average predition fraction is proportional to their *Average Prediction Probability* $\langle\langle p^{(m)}\rangle\rangle$

$$\phi^{(m)} = z(\beta)\langle\langle p^{(m)}\rangle\rangle \tag{13}$$

where

$$z(\beta) = \int dx \exp(-\beta\epsilon(x,w)) \tag{14}$$

normalizes the probability density for the conditional probability

$$p(x|w) \equiv \frac{1}{z(\beta)} \exp(-\beta\epsilon(x,w)) \tag{15}$$

which TLS use to describe the behavior of a single certain net w. Equation 15 can itself be obtained from a maximum entropy argument in x space but we will not do so here.

Now z is a function of β but *is assumed in TLS to be independent of* w. TLS show this to be rigorously true for layered nets with real outputs if ϵ is the usual squared error between data and output. It is true in that case because the area under a Gaussian is independent of the mean of the Gaussian.

Our derivation here is more direct than existing derivations of TLS, who apply Bayes' rule to statistical mechanics. In their treatment, the extra factor of z appears naturally. We can demonstrate the equivalence Equation 13 as follows. We solve Equation 15 for the exponential and substitute it into Equation 11

$$f(w) = \int dx\overline{p}(x)\ z(\beta)p(x|w) = z(\beta)g(w) \tag{16}$$

where $g(w)$ was defined as "the average generalization of the network" in (Tishby, Levin, Solla, 1989) "the sample average of the likelihood" of the network (Levin, Tishby, Solla, 1990) to be

$$g(w) \equiv \int dx\overline{p}(x)p(x|w). \tag{17}$$

We now substitute this into Equation 12 to get

$$\phi^{(m)} = \frac{z^{m+1}(\beta)\ \int dw\rho^{(0)}(w)g^{m+1}(w)}{z^{m}(\beta)\ \int dw\rho^{(0)}(w)g^{m}(w)} \tag{18}$$

which yields Equation 13 since Levin, Tishby, and Solla define APP as

$$\langle\langle p^{(m)}\rangle\rangle \equiv \frac{\langle\gamma^{m+1}\rangle_{\rho^{(0)}}}{\langle\gamma^{m}\rangle_{\rho^{(0)}}} \tag{19}$$

where we have used γ to distinguish their varable g from the function $g(w)$. They define the denominator of this expression as

$$\langle\gamma^{m}\rangle_{\rho^{(0)}} = \int dw\rho^{(0)}(w) \int d\gamma\ \gamma^{m}\delta(g(w)-\gamma). \tag{20}$$

which can be evaluated with the Dirac delta function to get

$$\langle\gamma^{m}\rangle_{\rho^{(0)}} = \int dw\rho^{(0)}(w)\ g^{m}(w) \tag{21}$$

with a similar expression for the numerator, so that

$$\langle\langle p^{(m)}\rangle\rangle = \frac{\int dw\rho^{(0)}(w)g^{m+1}(w)}{\int dw\rho^{(0)}(w)g^{m}(w)}. \tag{22}$$

Equation 13 follows immediately from Equations 22 and 12. Therefore our average prediction fraction is identical (except for a scale factor $z(\beta)$) to the TLS average prediction probability under the conditions that TLS assume.

For some problems we could argue that our $\phi^{(m)}$ is a more natural performance measure than the TLS $\langle\langle p^{(m)}\rangle\rangle$. But in fact, the two are identical except for scale $z(\beta)$ and the TLS restriction on the w dependence of that scale. For consistency with the existing work of Tishby, Levin, and Solla, we will use their $\langle\langle p^{(m)}\rangle\rangle$ in the remainder of this report.

3 Analysis of an elementary example

In this section we theoretically analyze the problem of estimating a constant from noisy measurements. The utility of this elementary example is that it admits an analytic solution for the APP which can be compared with conventional analysis. All the relevant integrals can be computed with the identity

$$\int_{-\infty}^{\infty} dx\ \exp\left(-a_1(x-b_1)^2 - a_2(x-b_2)^2\right) = \sqrt{\frac{\pi}{a_1+a_2}}\ \exp\left(-\frac{a_1a_2(b_1-b_2)^2}{a_1+a_2}\right). \tag{23}$$

We take the true value of the constant to be $\overline{w}$ and assume the noise to be zero mean additive Gaussian of variance $1/2\alpha$, so that the density of a measurement with value x is

$$\overline{p}(x) = \sqrt{\frac{\alpha}{\pi}} e^{-\alpha(x-\overline{w})^2}. \tag{24}$$

We choose a simple prior density as a Gaussian with mean w_0 and variance $1/2r$

$$\rho^{(0)}(w) = \sqrt{\frac{r}{\pi}} e^{-r(w-w_0)^2}. \tag{25}$$

We choose the simplest error function

$$\epsilon(x|w) = (x-w)^2, \tag{26}$$

the squared error between a sample x and the our estimate w.

According to TLS we now analyze the problem. We substitute the error function into Equation 15 to get

$$p(x|w) = \frac{1}{z(\beta)} e^{-\beta(x-w)^2} \tag{27}$$

with $z(\beta) = \sqrt{\frac{\pi}{\beta}}$ which is independent of w as assumed. We determine β by solving

$$\int dx p(x|w)\epsilon(x,w) = \epsilon_T \tag{28}$$

to get

$$\beta = \frac{1}{2\epsilon_T}. \tag{29}$$

The generalization, Equation 17, can now be evaluated with Equation 23

$$g(w) = \sqrt{\frac{\kappa}{\pi}} e^{-\kappa(w-\overline{w})^2}, \tag{30}$$

where

$$\kappa = \frac{\alpha\beta}{\alpha+\beta}, \tag{31}$$

is less than either α or β. The denominator of Equation 22 becomes

$$(\frac{\kappa}{\pi})^{m/2} \sqrt{\frac{r}{r+m\kappa}} \exp\left(-\frac{m\kappa r}{m\kappa+r}(\overline{w}-w_0)^2\right) \tag{32}$$

with a similar expression for the numerator.

3.1 Large training set size

The case of many examples or little prior knowledge is interesting. Consider Equations 22 and 32 for $m\kappa >> r$

$$\langle\langle p^{(m)}\rangle\rangle \longrightarrow \sqrt{\frac{\kappa}{\pi}}\sqrt{\frac{m}{m+1}}, \tag{33}$$

which climbs to an asymptotic value of $\sqrt{\frac{\kappa}{\pi}}$ for $m \longrightarrow \infty$. In order to compare this with intuition, consider that the sample mean of $\{x_1, x_2, ..., x_m\}$ approaches $\overline{w}$ to within a variance of $1/2m\alpha$, so that

$$\langle\rho^{(m)}(w)\rangle_x \approx \sqrt{\frac{m\alpha}{\pi}}e^{-m\alpha(x-\overline{w})^2} \tag{34}$$

which makes Equation 22 agree with Equation 33 for large enough β. In this sense, the statistical mechanical theory of learning differs from conventional Bayesian estimation only in its choice of an unconventional performance criterion APP.

3.2 Small training set size

We can demonstrate overtraining even for this elementary example when the size of the training set m is small and the the error on the training set ϵ_T is reduced too much. It will be sufficient to consider Equation 12 for $m\kappa << r$ so that

$$\langle\rho^{(m)}(w)\rangle_x \approx (\frac{\kappa}{\pi})^{m/2}e^{-m\kappa(\overline{\omega}-\omega_0)^2}, \tag{35}$$

by which Equation 22 becomes

$$\langle\langle p^{(m)}\rangle\rangle \longrightarrow \sqrt{\frac{\kappa}{\pi}}e^{-\kappa(\overline{\omega}-\omega_0)^2}. \tag{36}$$

This expression for APP depends on κ which in turn depends on β which is finally determined by the training target error ϵ_T. As a function of κ, Equation 36 exhibits a maximum at

$$\kappa_{opt} = \frac{1}{2(\omega_0 - \overline{\omega})^2}. \tag{37}$$

Using Equations 29 and 31, we find that the system will have the highest average prediction probability if training is stopped when the error on the training set is

$$\epsilon_{T,opt} = (\omega_0 - \overline{\omega})^2 - \frac{1}{2\alpha} \tag{38}$$

which depends only on the variance of the target distribution $1/2\alpha$ and the difference between the mean of the prior distribution ω_0 and the mean of the target distribution $\overline{\omega}$. Since $\langle\epsilon\rangle$ and ϵ_T cannot be negative, this optimum is physical only when it is positive. So when Equation 38 is negative, training should continue to the smallest possible error, since in that case there is no danger of overtraining. Now $\epsilon_{T,opt} > 0$ occurs only when the error of the prior estimate for the unknown constant is larger than the width of the prior distribution. Furthermore it can occur only when m is not too large, since we have assumed $m\kappa << r$. We can therefore translate Equation 38 into the following rule for this simplest of systems: overtraining occurs when the system is trained to too low an error (*i.e.* $\epsilon_T < \epsilon_{T,opt}$) on a training set that is too small (*i.e.* $m << r\kappa$) to compensate for an initial overconfidence in the prior estimate (*i.e.* $\frac{1}{2\alpha} > (\omega_0 - \overline{\omega})^2$).

4 General numerical procedure

In this section we show how to numerically apply the theory. We can estimate the moments of Equation 22 by the following Monte Carlo procedure. Given a data set $\{x_i\}_{i=1}^{i=N}$ drawn from the unknown density $\overline{p}$ on domain X with finite volume V, an error function $\epsilon(x|w)$, a training error ϵ_T, and a prior density $\rho^{(0)}(w)$ of vectors such that each w specifies a candidate model,

1. Construct two sample sets: a prior set of N_P functions $\{w_p\}$ drawn from $\rho^{(0)}(w)$ and a set of N_U input vectors $\{x_u\}$ drawn uniformly from X. For each p in the prior set and every u in the uniform set, tabulate the error $\epsilon_{up} = \epsilon(x_u|w_p)$. For each i in the training set and every u in uniform set tabulate $\epsilon_{ip} = \epsilon(x_i|w_p)$.

2. Determine the sensitivity β for a specified ϵ_T by solving

$$\frac{\sum_u e^{-\beta\epsilon_{up}}\epsilon_{up}}{\sum_u e^{-\beta\epsilon_{up}}} = \epsilon_T. \tag{39}$$

Equation 39 can be shown to be monotonic in β so this involves a simple one-dimensional search.

3. Estimate the average generalization of a given w_p from Equation 17

$$g(w_p) = \frac{1}{V} \frac{1/N \sum_i e^{-\beta \epsilon_{ip}}}{1/N_U \sum_u e^{-\beta \epsilon_{up}}}. \tag{40}$$

The factor of V remains in this particular expression because each probability density is normalized to have unit integral over X so that the Monte Carlo expression for integral of their product retains one factor of V, the integral of X itself.

4. The performance after m examples is the ratio of Equation 22. By construction the w_p are drawn from $\rho^{(0)}(w)$ so that

$$\langle\langle p^{(m)} \rangle\rangle = \frac{\sum_p g^{m+1}(w_p)}{\sum_p g^m(w_p)}. \tag{41}$$

5 Learning a density

A TLS theory is completely specified by a dataset $\{x_i\}_{i=1}^{i=N}$ (or the corresponding density $\overline{p}(x)$ in pattern or data space), a prior density $\rho^{(0)}(w)$ in w space, and an error function $\epsilon(x, w)$ that relates x's to w's. In this sense, learning a density is no different from learning a map. The error function $\epsilon(x, w)$ measures some important difference between data and model and for learning a density, the negative of the log likelihood is a natural choice.

Competitive learning nets (CLNs) of the form shown schematically in Figure 1 can be trained with the algorithm of Figure 2 to "learn" the density from which a set of training examples was drawn (Rumelhart and McClelland, 1986). We consider CLNs because they are familiar and useful to us (Van den Bout and Miller, 1990), because there exist two widely known training strategies for CLNs (the neurons can learn either independently or under a global interaction called conscience (DeSieno, 1988)), and because CLNs can be applied to one-dimensional problems without being too trivial. Asymptotic behavior of these nets depends on the error function $\epsilon(x, w)$ in a known and useful way: the asymptotic density of neurons can be algebraically related to the target density used for training(Ritter, 1991).

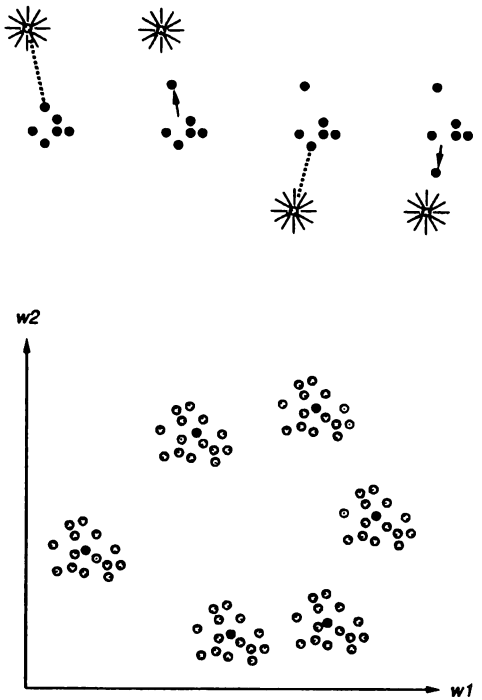

Figure 1: A 2 dimensional competitive learning net: **top** Evolution of neuron locations toward samples. **bottom** Equilibrium neuron locations in regions of high sample density.

Competitive learning nets with conscience qualitatively change their behavior when they are trained on finite sample sets containing fewer examples than neurons; except for that regime we found the theory satisfactory.

A CLN with k neurons is specified by a vector w of k neuron locations. We restricted ourselves to a one-dimensional problem on the unit interval. All experiments we will present in this section were conducted upon data drawn from the following one-dimensional training density

$$\overline{p}(x) = \begin{cases} \frac{1}{2\sqrt{x}} & 0 \leq x \leq 1, \\ 0 & \text{otherwise.} \end{cases}$$

The simplest choice for the prior density $\rho^{(0)}(w)$ of CLNs is a uniform density in a k dimensional space. This was our initial choice, however we eventually spent a large part of our time testing the sensitivity of predictions to variety

```
/* initialize neurons with random weights */
for( i ⇐ 1; i ≤ N; i ⇐ i + 1 )
        for( j ⇐ 1; j ≤ W; j ⇐ j + 1 )
                w_ij ⇐ random()

/* apply training vectors */
for( v∈ {training set} )

        /* compute distance of neurons from vector */
        for( i ⇐ 1; i ≤ N; i ⇐ i + 1 )
                d_i ⇐ 0
                for( j ⇐ 1; j ≤ W; j ⇐ j + 1 )
                        d_i ⇐ d_i + |v_j − w_ij|

        /* find the closest neuron to the vector */
        k ⇐ 1
        for( i ⇐ 1; i ≤ N; i ⇐ i + 1 )
                if( d_i < d_k )
                        k ⇐ i

        /* adjust weights of winning neuron */
        for( j ⇐ 1; j ≤ W; j ⇐ j + 1 )
                w_kj ⇐ w_kj + ε · (v_j − w_kj)

        /* reduce the learning rate */
        ε ⇐ α_ε · ε
```

Figure 2: Algorithm for training a CLN.

of prior densities. We chose the error function

$$\epsilon(x, w) = \min_{i=1}^{i=k} |x - w_i| \tag{42}$$

which is the distance between the sample x and location of the nearest neuron w_i. The error of several samples is the sum of the separate errors. If the samples are assumed to be independently drawn so that the joint probability of a set of samples is the product of the component probabilities, it can be shown that ϵ is proportional to the log of the conditional probability of observing x given the CLN w. Our chosen form for ϵ therefore corresponds to an exponential density around each neuron

$$p(x|w_i) \propto \exp(-|x - w_i|) \tag{43}$$

for x close enough to the i^{th} neuron. The neurons of a trained CLN tend to cluster in regions where the data density is highest. For $m \gg k \gg 1$, the distribution of neurons can be represented by a density which approaches a power the density of the data (Ritter, 1991).

In Figure 3 is the Average Prediction Probability (APP) for $k = 20$ versus m, for several values of target error ϵ_T and for two prior densitsities; first consider predictions from the uniform prior. For $\epsilon_T = 0.01$, APP practically attains its asymptote of 1.5 by $m = 40$ examples. Assuming the APP to be dominated in the limit by the largest g, we expect a CLN trained to an error of 0.01 on a set of 40 examples to perform 1.5 times better than an untrained net on unseen samples drawn from the same probability density. We can therefore define a predicted probable error of order

$$\epsilon_{prob} = \frac{1}{2\ k\ p^{(m)}}. \tag{44}$$

For $k = 20$, $\epsilon_{prob} = 0.017$ for $\epsilon_T = .01$ and $\epsilon_{prob} = 0.021$ for $\epsilon_T = 0.02$.

We performed 5,000 training trials of a 20-neuron CLN on randomly selected sets of 40 samples from the training density. Each network was trained to a target error in the range $[0.005, 0.03]$ on its 40 samples, and the average error on the total density was then calculated for the trained network. Figure 4 is a plot of 500 of these trials along with the predicted errors for various target errors. The probable error is qualitatively correct and the scatter of actual experiments increases in width by about the ratio

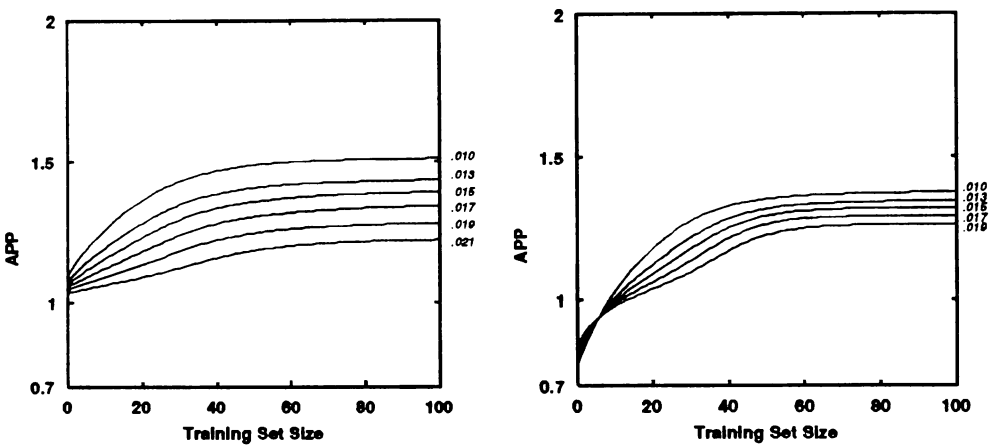

Figure 3: Predicted APP versus number of training samples for a 20-neuron competitive network trained to various target errors where the neuron weights were initialized from (a) a uniform density, (b) an antisymmetrically skewed density.

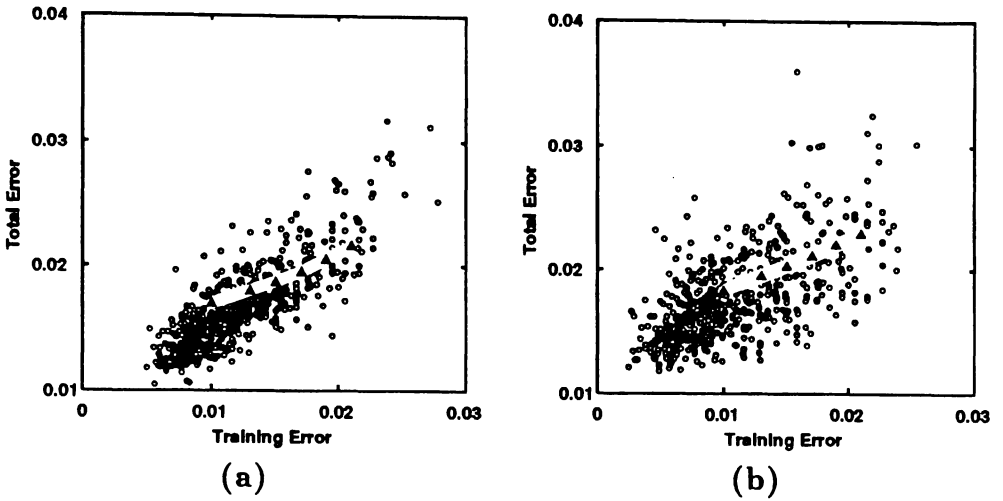

Figure 4: Experimentally determined and predicted values of total error across the training density after competitive learning was performed using a 20-neuron network trained to various target errors (**a**) with 40 samples, (**b**) with 20 samples.

of APPs for $m = 20$ and $m = 40$. For the case of $m = 20$ examples, the same net can only be expected to exhibit probable errors of .019 and .023 for corresponding training target errors, which is compared graphically in Figure 4 with the experimentally determined errors for $m = 20$.

The APP curves saturate at a value of m that is insensitive to the prior density from which the nets are drawn. The vertical scale does depend somewhat on the prior however. Consider Figure 3, which also shows the APP curves for the same $k = 20$ net with the prior density antisymmetrically skewed *away* from the true density by the following function:

$$\rho^{(0)}(\omega) = \begin{cases} \frac{1}{2\sqrt{1-\omega}} & 0 \leq \omega \leq 1, \\ 0 & \text{otherwise.} \end{cases}$$

For $m > 20$ the *shapes* of the curves are almost unchanged, even though the vertical scale is different: saturation occurs at about the same value of m. Even when the prior greatly overrepresents poor nets, their effect on the prediction rapidly diminishes with training set size. This is important

because in actual training, the effect of the initial configuration is also quickly lost. For $m < 20$ the predictions are not valid in any case, since our simple error function does not reflect the actual probability even approximately for $m < k$ in these nets. It is for $m < 20$ where the only significant differences between the two families of curves occur. We have also been able to draw the same conclusions from less structured prior densities generated by assigning positive normalized random numbers to intervals of the domain. In fact, we were not able to produce curves worse than those of Figure 3.

Moreover, we generally find that TLS predicts that about twice as many samples as neurons are needed to train competitive nets of other sizes.

The previous curves were produced with large total sample sets $N = 1000$. We subsequently reran the experiments with $N = 100$ with essentially identical results (we do not plot them because the two are indistinguishable). It is reassuring that the predictions of the theory are reliable even for relatively small sample sets.

6 Conclusion

We have derived the TLS theory of learning using the principle of maximum entropy instead of a combination of statistical mechanics and Bayes' rule. We show that TLS can be applied generally to learning and modeling. We apply it to learning a constant and to learning a one dimensional density in the annealed approximation of TLS. We considered the effects of varying the number of examples m, the target training error ϵ_T (or equivalently β), and the choice of prior density $\rho^{(0)}(w)$. These experiments on learning a density are consistent with learning a binary output (Bilbro and Snyder, 1990), a ternary output (Chow, Bilbro, and Yee, 1990), and a continuous output (Bilbro and Klenin, 1990). We find if saturation occurs for m substantially less than the total number of available samples, say $m < T/2$, that m is a good predictor of sufficient training set size. Moreover there is evidence from a reformulation of the learning theory based on the grand canonical ensemble that also supports this statistical approach (Klenin,1990). For small problems the theory is very easy to use. We have no experience yet in applying the theory to large problems with say more than 50 unknown weights or parameters.

The TLS theory appears promising as a predictor of sufficient training

set size, however another question remains. It is difficult to obtain large overtraining effects from this theory even though some traces of overtraining remain: The curves of Figure 3 can be made to cross for small m and poor prior. In the Equation 38 of the elementary example, it is possible to obtain a positive $\epsilon_{T,opt}$. However it appears that some aspects of overtraining do not survive the annealed approximation. This is consistent with the experience of other workers(Solla, 1990).

References

G. L. Bilbro and M. Klenin. (1990) Thermodynamic Models of Learning: Applications. Unpublished.

G. L. Bilbro and W. E. Snyder. (1990) Learning theory, linear separability, and noisy data. CCSP-TR-90/7, Center for Communications and Signal Processing, Box 7914, Raleigh, NC 27695-7914.

M. Y. Chow, G. L. Bilbro and S. O. Yee. (1990) Application of Learning Theory to Single-Phase Induction Motor Incipient Fault Detection Artificial Neural Networks. Submitted to *International Journal of Neural Systems.*

D. DeSieno. (1988) Adding a conscience to competitive learning. In *IEEE International Conference on Neural Networks*, pages I:117–I:124.

E. T. Jaynes. (1979) Where Do We Stand on Maximum Entropy?. In R. D. Leven and M. Tribus (Eds.), *Maximum Entropy Formalism*, M. I. T. Press, Cambridge, pages 17-118.

M. Klenin. (1990) Learning Models and Thermostatistics: A Description of Overtraining and Generalization Capacities. NETR-90/3, Center for Communications and Signal Processing, Neural Engineering Group, Box 7914, Raleigh, NC 27695-7914.

E. Levin, N. Tishby, and S. A. Solla. (1990) A Statistical approach to learning and generalization in layered neural networks. *Proceedings of the IEEE* , Vol. 78, No. 10, pages 1568-1574.

D. B. Schwartz, V. K. Samalan, S. A. Solla & J. S. Denker. (1990) Exhaustive Learning. *Neural Computation.*

H. Ritter. (1991) Asymptotic Level Density for a Class of Vector Quantization Processes. IEEE Trans. NN.

D. Rumelhart and J. McClelland. (1986) *Parallel Distributed Processing: Explorations in the Microstructure of Cognition*, Chapter 5, MIT Press.

S. A. Solla. (1990) Private communication.

N. Tishby, E. Levin, and S. A. Solla. (1989) Consistent inference of probabilities in layered networks: Predictions and generalization. *IJCNN*, IEEE, New York, pages II:403-410.

D. E. Van den Bout and T. K. Miller III. (1990) TInMANN: The integer Markovian artificial neural network. Accepted for publication in the *Journal of Parallel and Distributed Computing.*