EMPIRICAL GENERALIZATION ASSESSMENT OF NEURAL NETWORK MODELS*

Jan Larsen and Lars Kai Hansen

The Computational Neural Network Center (CONNECT)

Electronics Institute, Building 349

Technical University of Denmark

DK-2800 Lyngby, Denmark

emails: jlarsen,lkhansen@ei.dtu.dk

Abstract. This paper addresses the assessment of generalization performance of neural network models by use of empirical techniques.

We suggest to use the cross-validation scheme combined with a resampling technique to obtain an estimate of the generalization performance distribution of a specific model. This enables the formulation of a bulk of new generalization performance measures. Numerical results demonstrate the viability of the approach compared to the standard technique of using algebraic estimates like the FPE [1].

Moreover, we consider the problem of comparing the generalization performance of different competing models. Since all models are trained on the same data, a key issue is to take this dependency into account.

The optimal split of the data set of size N into a cross-validation set of size $N\gamma$ and a training set of size $N(1-\gamma)$ is discussed. Asymptotically (large data sets), $\gamma_{\mathrm{opt}} \rightarrow 1$ such that a relatively larger amount is left for validation.

INTRODUCTION

Consider a tapped-delay neural network predicting a stochastic output signal[1] $y(k)$ from observations of the L-dimensional stochastic input vector signal

*To appear in Proceedings of the 1995 IEEE Workshop on Neural Networks for Signal Processing, Boston, Aug. 31 – Sep. 2, 1995.

[1]For convenience, we focus on single output models.

$\boldsymbol{x}(k) = [x(k), x(k-1), \cdots, x(k-L+1)]^\top$. Let $f(\cdot)$ denote the mapping of the neural network, and $\boldsymbol{w}$ the vector of network weights (parameters). Then the prediction is given as: $\widehat{y}(k) = f(\boldsymbol{x}(k); \boldsymbol{w})$.

The basic object of interest in neural network modeling is the conditional input-output distribution $p(y|\boldsymbol{x})$, i.e., the probability distribution of the output conditioned on a test input vector, see e.g., [16]. Normally the network is trained to implement the conditional mean[2], $E\{y|\boldsymbol{x}\} = \int y \cdot p(y|\boldsymbol{x})\, dy$. The first source of uncertainty is the inherent prediction error $\varepsilon = y - E\{y|\boldsymbol{x}\}$ which – per definition – cannot be modeled. Another considerable source of uncertainty is the estimation of $E\{y|\boldsymbol{x}\}$ from a limited number of training data.

This paper deals with empirical assessment of model quality expressed in terms of generalization performance defined as prediction accuracy on future data. Reliable estimates of the generalization performance of a particular model is very important for practical applications. Moreover, in order to choose the best model from a pool of candidate model architectures[3], one requires a test which determines if a particular model has a significantly higher generalization performance than a competing model. The empirical framework enables both *absolute* and *comparative* generalization assessment.

The generalization performance can be decomposed into three components, see e.g., [3], [6]. The first term is due to the inherent prediction error, ε. The second term expresses the insufficiency of the neural architecture[4] to model the conditional mean, and is often referred to as the model bias. Finally, the third term reflects finite training set effects, also known as the model variance. While the first term - per definition - cannot be decreased, there will normally exist a trade off between bias and variance which is accomplished by optimizing the architecture, e.g., by using pruning techniques.

ON GENERALIZATION PERFORMANCE

Suppose the network is trained by minimizing a cost function, viz. the sum of a loss function, $S_N(\boldsymbol{w})$, and a regularization term $R(\boldsymbol{w})$, i.e.,

$$C_N(\boldsymbol{w}) = S_N(\boldsymbol{w}) + R(\boldsymbol{w}) = \frac{1}{N}\sum_{k=1}^{N} \ell\left(y(k), \widehat{y}(k); \boldsymbol{w}\right) + R(\boldsymbol{w}) \tag{1}$$

where $\ell(\cdot)$ measures the distance between the output $y(k)$ and the network prediction $\widehat{y}(k) = f(\boldsymbol{x}(k); \boldsymbol{w})$. Even though much of the material in this paper applies for general loss functions, often the mean square error loss function, $\ell = (y - \widehat{y})^2 = e^2$ is considered. N defines the number of training examples, i.e., input-output pairs of the training set: $\mathcal{D} = \{(\boldsymbol{x}(k), y(k))\}_{k=1}^{N}$.

[2] This is optimal when using a mean square error cost function, see e.g., [16].

[3] E.g., feed-forward neural nets with different input lag-space and number of hidden units.

[4] The architecture is presumed to be finite, i.e., the weight vector is finite dimensional.

Training on the full set of examples provides the estimated weight vector $\widehat{\boldsymbol{w}} = \arg\min_{\boldsymbol{w}} C_N(\boldsymbol{w})$. The generalization error, G, is defined as the *expected loss* of the estimated model on a test sample $(\boldsymbol{x}, y)$ independent of those in the training set,

$$G(\widehat{\boldsymbol{w}}) = E\left\{\ell(y, \widehat{y}; \widehat{\boldsymbol{w}})\right\} = \int \ell(y, \widehat{y}; \widehat{\boldsymbol{w}}) \cdot p(\boldsymbol{x}, y)\, d\boldsymbol{x}\, dy \tag{2}$$

where $E\{\cdot\}$ denotes expectation w.r.t. the unknown joint input-output probability density $p(\boldsymbol{x}, y)$. $G(\widehat{\boldsymbol{w}})$ depends on the actual training set $\mathcal{D}$ through the estimated weights $\widehat{\boldsymbol{w}}$ and has the lower bound $G_{\min} = G(\boldsymbol{w}^*)$. $\boldsymbol{w}^*$ denotes the optimal weight vector $\boldsymbol{w}^* = \arg\min_{\boldsymbol{w}} E\{C_N(\boldsymbol{w})\} = \arg\min_{\boldsymbol{w}} [G(\boldsymbol{w}) + R(\boldsymbol{w})]$ which corresponds to training on an infinite training set. Under fairly mild assumptions, it is possible to show $\lim_{N\to\infty} \widehat{\boldsymbol{w}} = \boldsymbol{w}^*$, see e.g., [10], [11], [17]. $G_{\min}$ expresses the fundamental uncertainty of y when x is known, and furthermore the potential lack of modeling capability, i.e., the network is incapable of implementing the optimal[5] function, $g(\boldsymbol{x}) = \arg\min_{\phi(\boldsymbol{x})} E\{\ell(y, \phi(\boldsymbol{x}))\}$, $\phi(\boldsymbol{x}) : \mathbb{R}^p \mapsto \mathbb{R}$. Insufficient modeling capability is due to two facts:

- In general, when using a finite architecture the model is incomplete, i.e., $f(\boldsymbol{x}; \boldsymbol{w}^\circ) \neq g(\boldsymbol{x})$ where $\boldsymbol{w}^\circ = \arg\min_{\boldsymbol{w}} G(\boldsymbol{w})$ is the weights minimizing the expected loss using the architecture embodied by $f(\cdot)$.
- Regularization implies that the optimal weight vector $\boldsymbol{w}^*$ does not equal $\boldsymbol{w}^\circ$; even when using a complete model.

Since the N samples in $\mathcal{D}$ are randomly selected from the joint density $p((\boldsymbol{x}(1), y(1)), \cdots, (\boldsymbol{x}(N), y(N)))$ the generalization error $G(\widehat{\boldsymbol{w}})$ is stochastic with a certain *generalization error probability distribution* $P(G) = \mathrm{Prob}\{G(\widehat{\boldsymbol{w}}) < G)\}$ and associated density $p(G)$.

The object of interest for model design could be either the full generalization distribution or just the generalization error $G(\widehat{\boldsymbol{w}})$ on the particular training set available. These cases are treated separately in the following. If one has a strong belief in the training set (e.g., if it is large) one might address $G(\widehat{\boldsymbol{w}})$. Otherwise, it might be better to consider the training set as a typical set drawn randomly from the joint input-output distribution in order to reveal the generic characteristics of the employed model.

Since $p(G)$ depends on the true distribution of data, the model architecture, and the number of training data, it is impossible to fully characterize it. However, it is possible to give some general properties. Obviously, $p(G) = 0$ as $G < G_{\min}$. For finite training sets, $p(G)$ will have non-zero values for $G \geq G_{\min}$, and since $\lim_{N\to\infty} \widehat{\boldsymbol{w}} = \boldsymbol{w}^*$, $p(G)$ tends to a Dirac delta function $\delta(G - G_{\min})$ for $N \to \infty$. If the model is complete, the loss function is the mean square error and no regularization is employed, it is possible to show[6]

[5] With respect to the employed loss function.

[6] This is done by using second order expansions of $G(\widehat{\boldsymbol{w}})$ around $\boldsymbol{w}^*$, and the fact that that the fluctuations $\Delta\boldsymbol{w} = \widehat{\boldsymbol{w}} - \boldsymbol{w}^*$ are asymptotically Gaussian distributed. See e.g., [6], [7], [8] and [16].

asymptotically as $N \to \infty$, $G(\widehat{\boldsymbol{w}}) \sim \sigma_\varepsilon^2(1 + \chi^2(p)/N)$ where σ_ε^2 is the prediction error noise variance and $\chi^2(p)$ is the χ^2-distribution with $p = \dim(\boldsymbol{w})$ degrees of freedom.

The literature on generalization theory and estimation of generalization error does not in general address the problem of characterizing the full prediction risk probability density. Most work has focused on simple measures of location such as the *average generalization error*

$$avr(G) = E_{\mathcal{D}}\left\{G(\widehat{\boldsymbol{w}})\right\} = \int G \cdot p(G)\, dG. \tag{3}$$

This includes algebraic estimators like FPE [1], FPER [7], GEN [5], GPE [9] and NIC [10] which are valid asymptotically $N \to \infty$ and make several assumptions on the model and statistics of the data. However, also algebraic estimates of fractiles of $p(G)$ have been developed, see e.g., [14], [15]. Thus the $1-\alpha$ fractile $G_{1-\alpha}$ defined by $\mathrm{Prob}\{G \le G_{1-\alpha}\} = 1-\alpha$ guarantees that the probability of G exceeding $G_{1-\alpha}$ is α, which can be set to some low percentage.

EMPIRICAL GENERALIZATION ERROR ESTIMATION

If the object of interest is the generalization error $G(\widehat{\boldsymbol{w}})$ for the particular training set available, we consider the hold-out cross-validation technique [13] for estimating $G(\widehat{\boldsymbol{w}})$. Suppose that a cross-validation set $\mathcal{C}$ of $N_c = \lceil N\gamma \rceil$,[7] $0 < \gamma < 1$, samples are hold out for cross-validation and denote by $\mathcal{T}$ the remaining $N_t = N - N_c$ data for training, i.e., let $\widehat{\boldsymbol{w}} = \arg\min_{\boldsymbol{w}} C_{N_t}(\boldsymbol{w})$. The cross-validation estimate of $G(\widehat{\boldsymbol{w}})$ then reads:

$$\widehat{G}(\widehat{\boldsymbol{w}}) = \frac{1}{N_c}\sum_{k\in\mathcal{C}} \ell(y(k), \widehat{y}(k); \widehat{\boldsymbol{w}}) \tag{4}$$

Under suitable regularity conditions, $\widehat{G}(\widehat{\boldsymbol{w}}) \to G(\widehat{\boldsymbol{w}})$ as $N_c \to \infty$. However, a very large cross-validation set leaves only few data for training thus increasing $G(\widehat{\boldsymbol{w}})$. Obviously, there exists an optimal fraction γ which trades off the conflicting aims. Assume that the quality of the cross-validation estimator is measured by

$$\mathrm{MSE}(\gamma) = E_{\mathcal{D}}\left\{\left[\widehat{G}(\widehat{\boldsymbol{w}}) - G(\widehat{\boldsymbol{w}})\right]^2\right\} \tag{5}$$

where $E_{\mathcal{D}}\{\cdot\}$ is the expectation w.r.t. all training data. Further, assume that the loss is the mean square error and that the training data are independent. Since $E_{\mathcal{C}}\{\widehat{G}(\widehat{\boldsymbol{w}})\} = G(\widehat{\boldsymbol{w}})$ evaluating Eq. (5) gives

$$\mathrm{MSE}(\gamma) = E_{\mathcal{T}}\left\{\frac{1}{N_c}\left[E_{\mathcal{C}}\left\{e^4(\widehat{\boldsymbol{w}})\right\} - G^2(\widehat{\boldsymbol{w}})\right]\right\} \tag{6}$$

[7] $\lceil\cdot\rceil$ denotes rounding upwards to the nearest integer.

Using asymptotic expansions (see e.g., [6],[7]) for the terms in Eq. (6) and considering the model to be complete, it is possible to show that the the optimal fraction is given by $\gamma_{\rm opt} = 1 - \sqrt{\beta/N}$ where $\beta = 4p\sigma_\varepsilon^2/(\xi - 1)$, ξ is the kurtosis of the inherent noise (equal to 3 for Gaussian noise), and $p = \dim(\boldsymbol{w})$. That is, $\lim_{N\to\infty} \gamma_{\rm opt} = 1$ while $N_t = O(\sqrt{N\beta})$ and $N_c = O(N - \sqrt{N\beta})$ asymptotically. It should be emphasized that the choice of γ for a finite small N still needs to be tuned by hand.

The hold-out cross-validation scheme can also be used for comparing generalization errors of different models. Consider the scenario of pruning a nested family of neural net models and suppose that two alternative models with weights $\widehat{\boldsymbol{w}}_1$, $\widehat{\boldsymbol{w}}_2$ both are estimated from $\mathcal{T}$. If we take $\widehat{\boldsymbol{w}}_2$ to be a subset of $\widehat{\boldsymbol{w}}_1$, i.e., $\dim(\boldsymbol{w}_2) < \dim(\boldsymbol{w}_1)$, the hypothesis to be tested is: $G(\widehat{\boldsymbol{w}}_2) > G(\widehat{\boldsymbol{w}}_1)$. Since the models are nested and validated on the same cross-validation set, the corresponding estimated generalization errors are highly dependent. A straight forward procedure which puts error bars on the individual generalization error estimates may fail to unveil the superiority of one model relative to another. The dependence is easily taken into account by analyzing the difference in generalization error, $\Delta\widehat{G} = \widehat{G}(\widehat{\boldsymbol{w}}_2) - \widehat{G}(\widehat{\boldsymbol{w}}_1)$. According to the central limit theorem[8] $\Delta\widehat{G}$ tends to a Gaussian distribution as $N_c \to \infty$. That is a standard t-test for the hypothesis can be used. If $\Delta\widehat{G}/std(\Delta\widehat{G}) < t_\alpha(N_c - 1)$ we reject the hypothesis on an α significance level. $t_\alpha(N_c - 1)$ is the α-fractile of the t-distribution with $N_c - 1$ degrees of freedom, and $std(\cdot)$ denotes the standard deviation. Define $\Delta e^2(k) = e^2(k, \widehat{\boldsymbol{w}}_2) - e^2(k, \widehat{\boldsymbol{w}}_1)$ then the standard deviation is estimated via $(\widehat{std}(\Delta\widehat{G}))^2 = (N_c - 1)^{-1} N_c^{-1} \sum_{k\in\mathcal{C}} (\Delta e^2(k) - \Delta\widehat{G})^2$.

EMPIRICAL GENERALIZATION ERROR DISTRIBUTIONS

We suggest to estimate the generalization error distribution by using leave-out cross-validation [12], [13] and resampling techniques. The basic algorithm is given by:

1. Specify the leave-out fraction γ and determine $N_c = \lceil N\gamma \rceil$. Further specify the number of resamplings $J \le N!/N_c!(N - N_c)!$.
2. For $j = 1, 2, \cdots, J$ split the training set randomly into a cross-validation subset, $\mathcal{C}_j$, and a training set, $\mathcal{T}_j = \mathcal{D} \setminus \mathcal{C}_j$ not used previously[9].
3. Train on $\mathcal{T}_j$ with $N_t = N - N_c$ examples to obtain the weight estimate $\widehat{\boldsymbol{w}}_j$ and calculate the empirical mean of the loss on the samples $\mathcal{C}_j$, which yields the generalization error estimate:

$$\widehat{G}_j = \widehat{G}_j(\widehat{\boldsymbol{w}}_j) = \frac{1}{N_c} \sum_{k\in\mathcal{C}_j} \ell(y(k), \widehat{y}(k); \widehat{\boldsymbol{w}}_j). \tag{7}$$

[8] This also applies when the error signal is a strongly mixing sequence (time-dependent).

[9] Note that this is resampling without replacement, as opposed to the Bootstrap technique.

The training in step 3 can be very time consuming and in [4] we developed an approximate technique for leave-one-out cross-validation.

Ideally, when estimating Eq. (7) we should train and test on independent sets. Moreover, the training sets should be independent. These properties only hold approximately. First, it is very important to stress the significance leaving out a *fraction* γ compared to the standard approach of leaving out a *fixed* number. In the latter case, the different training sets will be too dependent even in the limit of $N \to \infty$[10]. However, as discussed in the previous section by letting $\gamma \to 1$ and if $N_t = O(\nu \log(N))$, $N_c = O(N - \nu \log(N))$, where ν is a constant, all moments of $\widehat{G}_j$ converges[11]. The number of resamplings J should also be allowed to increase towards infinity as N grows. Secondly, for most signal processing problems, time-dependence can especially for small N cause noise in the estimates. However, asymptotically this is no problem since we expect the input signal to be a strongly mixing sequence, i.e., the time-dependence vanishes for large lags.

From the estimates $\widehat{G}_j$ in Eq. (7) it is possible to form the *empirical generalization error distribution*

$$P_{\mathrm{emp}}(G) = \frac{1}{J} \sum_{j=1}^{J} \mu(G - \widehat{G}_{(j)}) \tag{8}$$

where $\widehat{G}_{(1)} \leq \widehat{G}_{(2)} \cdots \leq \widehat{G}_{(J)}$ is the sample order statistics, and $\mu(G - G_{(j)}) = 1$ when $G \geq G_{(j)}$, and zero otherwise.

Since $p(G)$ is highly non-Gaussian and long tailed (which is demonstrated experimentally below), the mean and variance are not sufficient for characterizing the shape of $p(G)$. It may consequently be desirable to consider more robust location and dispersion measures which we are able to calculate with $P_{\mathrm{emp}}(G)$ in hand. In general the location of $p(G)$ delivers an estimate of the level of generalization error. The dispersion conveys the fluctuation in generalization error and might suggest if the current number of examples is sufficient for learning the task properly. We consider the following quantities:
Location:

- The average $avr(G) = \int G p(G)\, dG$.
- The trimmed average $tavr(G) = \int_{G_{5\%}}^{G_{95\%}} G p(G)\, dG$ which reflects the average in which the highest and lowest 5% of the data are excluded.
- The median $med(G)$ equal to the $\alpha = 50\%$ fractile $G_{50\%}$.

Dispersion:

- The standard deviation $std(G) = (\int [G - avr(G)]^2\, dG)^{1/2}$.
- The median absolute deviation $mad(G) = med(\,|G - med(G)|\,)$.

[10]This is discussed in the literature of the so-called Jackknife estimators, see e.g., [2], [11, Ch. 5.7]

[11]This is a generalization of what was stated in the previous section for convergence of the second order moment in Eq. (5).

- The interquartile range $iqr(G) = G_{75\%} - G_{25\%}$.

Due to the fact that $p(G)$ follows a χ^2 like distribution, we might consider a transformation of G in order to make it more well behaved. In the general family of Box-Cox transformations (see e.g., [11, Ch. 2.8]) we found that a suitable transformation[12] is $\widetilde{G} = \log(1+G)$.

As in the previous section it is possible to compare the generalization ability e.g, by comparing estimated average generalization errors for two models described by $\boldsymbol{w}_2$, $\boldsymbol{w}_1$. Define the associated estimates $\widehat{avr}(G)_i = J^{-1}\sum_{j=1}^{J} G_{ij}(\widehat{\boldsymbol{w}}_{ij})$, $i = 1, 2$, and the difference $\Delta\widehat{avr}(G) = \widehat{avr}(G_2) - \widehat{avr}(G_1)$. For J large $\Delta\widehat{avr}(G)$ tends to a Gaussian distribution by the central limit theorem with standard deviation given by

$$std(\Delta\widehat{avr}(G)) = \sqrt{\frac{1}{(J-1)J}\sum_{j=1}^{J}\left[G_{2j} - G_{1j} - \Delta\widehat{avr}(G)\right]^2} \tag{9}$$

Here the individual differences are assumed to be independent. A standard t-test (as described previously) can then be applied.

NUMERICAL EXAMPLES

Consider the following data generating system: $y(k) = \boldsymbol{x}^\top(k)\boldsymbol{w}^\circ + \varepsilon(k)$. $\boldsymbol{x}(k)$ follows a $L = 10$ variate Gaussian distribution $\mathcal{N}(\boldsymbol{0}, \boldsymbol{H})$ with $\boldsymbol{H}$ chosen as a random positive definite symmetric matrix. $\boldsymbol{x}(k)$ is time-dependent: each component is a first order AR-process with coefficient 0.6518 scaled to give unit variance; thus implementing a low-pass filter with memory length approx. equal to 7. The noise $\varepsilon(k) \sim \mathcal{N}(0, \sigma_\varepsilon^2)$ is i.i.d. and independent of $\boldsymbol{x}(k)$. The weights, $\boldsymbol{w}^\circ$, were chosen independently from a $\mathcal{N}(0,1)$ distribution.

We generated $Q = 30000$ independent training sets of size $N = 20$ and trained with a $p = 10$ dimensional linear model using the mean square error cost (without regularization) to obtain the estimates $\widehat{\boldsymbol{w}}^{(i)}$, $i \in [1; Q]$. This enables a highly accurate estimate of the considered generalization performance measures. As an example, the "true" average generalization error is calculated by $avr(G) = Q^{-1}\sum_{i=1}^{Q} G(\widehat{\boldsymbol{w}}^{(i)})$ where $G(\widehat{\boldsymbol{w}}^{(i)}) = \sigma_\varepsilon^2 + (\widehat{\boldsymbol{w}}^{(i)} - \boldsymbol{w}^\circ)^\top \boldsymbol{H}(\widehat{\boldsymbol{w}}^{(i)} - \boldsymbol{w}^\circ)$. For $q = 500$ of the $Q = 30000$ training sets we applied the leave-out procedure with $\gamma = 0.25$, $J = 500$ to obtain the weight estimates $\widehat{\boldsymbol{w}}_j^{(i)}$, and corresponding generalization error estimates $\widehat{G}_j$, $j \in [1; J]$. For comparison we calculated the FPE [1] estimate of $avr(G)$ by $\mathrm{FPE}^{(i)} = S_N(\widehat{\boldsymbol{w}}^{(i)}) \cdot (N+p)/(N-p)$.

Fig. 1 shows the obtained generalization error probability distributions. Table 1 shows a comparison of the suggested measures of location and dispersion. We consider the transformed variables $\widetilde{G}$ which experimentally showed to improve the performance significantly over G. The table indicates that the

[12] This implies: $\widetilde{G} = 0$ for $G = 0$.

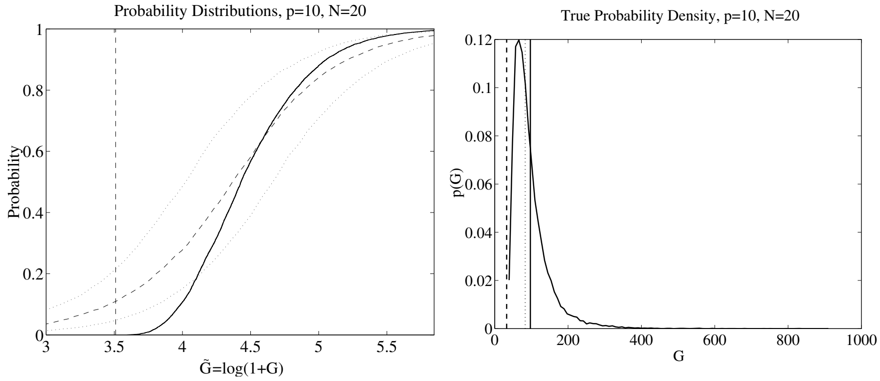

Figure 1: *Left panel*: True (solid) and empirical generalization error distributions as a function of $\widetilde{G} = \log(1+G)$. The dashed line indicates the median distribution of $q = 500$ obtained by leave-out cross-validation while the dotted lines denote the 25% and 75% fractiles. The vertical dashed line is the lower bound $\log(1+\sigma_\varepsilon^2)$. *Right panel*: True generalization error density $p(G)$ estimated from $Q = 30000$ replications. The vertical dashed line is the lower bound $\sigma_\varepsilon^2 = 32.45$. The vertical solid line denotes the average, and the vertical dotted the median. Note that $p(G)$ is highly non-Gaussian and long-tailed (ranges to $G = 1000$ approx.). This implies that the classical measure of location, viz. the average overestimates the typical (the mode) generalization error.

proposed leave-out technique is fairly accurate for estimating the location and dispersion measures even though the number of training data is only twice as large as the number of weights. Definitely, the leave-out method outperforms the classical FPE estimate at the expense of increased computational complexity. However, the framework offers the possibility of estimating other quantities which are not possible in the asymptotic framework on which FPE relies.

We considered furthermore the comparison of two competing linear models: $\boldsymbol{w}_1$ with dimension $p_1 = 10$, and $\boldsymbol{w}_2$ with dimension $p_2 = 9$ which consequently is an incomplete model of the true data generating system. The true difference in average generalization ability $\Delta avr(G)$ is positive thus indicating that one should prefer model 1 over model 2. Using the same simulation setup as described above the t-test on a specified $\alpha = 5\%$ significance level resulted in that the hypothesis fails to be accepted in approx. 30% of the cases. More over we considered estimating $\mathrm{Prob}(G_2 > G_1)$ from the empirical distributions. It turned out that the estimate tend to under estimate the probability by 20%. Further, it is somewhat more robust than the estimates of the location measures of the generalization error difference.

[14]When considering $\widetilde{G}$ the FPE estimate becomes: $\log(1+\sigma_\varepsilon^2) + \sigma_\varepsilon^2 p/(1+\sigma_\varepsilon^2)N$ with $\sigma_\varepsilon^2 = S_N(\widehat{\boldsymbol{w}})/(N-p)$.

Measure	Min.	25% fract.	Median	75% fract.	Max.
FPE	−62.0	−19.3	−12.3	−5.83	19.7
$\widehat{avr}(\widetilde{G})$	−24.0	−15.5	−7.00	2.27	6.72
$\widehat{tavr}(\widetilde{G})$	−23.5	−15.2	−6.47	2.38	7.05
$\widehat{med}(\widetilde{G})$	−23.2	−15.3	−6.47	3.36	9.31
$\widehat{std}(\widetilde{G})$	44.6	49.9	53.8	62.9	89.4
$\widehat{mad}(\widetilde{G})$	21.6	29.9	40.7	51.4	83.6
$\widehat{iqr}(\widetilde{G})$	20.1	28.6	40.3	53.2	89.4

Table 1: The values are deviations from the true measures in percent when considering the transformed value $\widetilde{G} = \log(1+G)$, e.g., $100\% \cdot (\widehat{avr}(\widetilde{G}) - avr(\widetilde{G}))/avr(\widetilde{G})$. The columns indicate the fluctuation in the deviations w.r.t. the $q = 500$ times the leave-out cross-validation procedure is replicated. As regards FPE[14], the fluctuations are based on $Q = 30000$ replications. In median the location measures $avr(\widetilde{G})$, $tavr(\widetilde{G})$, and $med(\widetilde{G})$ seem to underestimate but are still fairly close to zero, and closer than the estimate of $avr(\widetilde{G})$ obtained by FPE. Moreover, the fluctuations are much smaller when considering the fractiles. As regards the dispersion measures $std(\widetilde{G})$, $mad(\widetilde{G})$, and $iqr(\widetilde{G})$ we note that they overestimates by roughly 50%; however, with fairly small amount of fluctuation. The small fluctuation relates strongly to the fact that the transformed variable is used.

CONCLUSION

This paper reports on generalization performance measures which can be attained empirically by using the cross-validation technique in combination with resampling. The major advantage is that the framework provides insight into the shape of the generalization error probability distribution by considering different location and dispersion measures. Traditionally, only the average generalization error has been investigated; however a simple simulation study shows that this measure overestimates the typical generalization performance of a model estimated from a randomly selected set of N examples. Moreover, the assessment of dispersion measures allows for testing the hypothesis whether a model generalizes significantly better than a competitor.

ACKNOWLEDGMENTS

This research was supported by the Danish Natural Science and Technical Research Councils through the Computational Neural Network Center (CONNECT). JL furthermore acknowledge the Radio Parts Foundation for financial support.

REFERENCES

[1] H. Akaike, "Fitting Autoregressive Models for Prediction," Annals of the Institute of Statistical Mathematics, vol. 21, pp. 243–247, 1969.

[2] T. Fox, D. Hinkley & K. Larntz, "Jackknifing in Nonlinear Regression," Technometrics, vol. 22, pp. 29–33, 1980.

[3] S. Geman, E. Bienenstock & R. Doursat, "Neural Networks and the Bias/Variance Dilemma," Neural Computation, vol. 4, pp. 1–58, 1992.

[4] L.K. Hansen & J. Larsen, "Linear Unlearning for Cross-Validation," submitted for publication, 1995. harlar.luloo.ps.Z is retrieved by anonymous ftp at ei.dtu.dk.

[5] J. Larsen, "A Generalization Error Estimate for Nonlinear Systems," in S.Y. Kung, F. Fallside, J. Aa. Sørensen & C.A. Kamm (eds.) Neural Networks for Signal Processing 2: Proceedings of the 1992 IEEE-SP Workshop, Piscataway, New Jersey: IEEE, 1992, pp. 29–38.

[6] J. Larsen, Design of Neural Network Filters, Ph.D. Thesis, Electronics Institute, The Technical University of Denmark, March 1993.

[7] J. Larsen & L.K. Hansen, "Generalization Performance of Regularized Neural Network Models," in J. Vlontzos, J.-N. Hwang & E. Wilson (eds.), Proceedings of the IEEE Workshop on Neural Networks for Signal Processing IV, Piscataway, New Jersey: IEEE, pp. 42–51, 1994.

[8] L. Ljung, System Identification: Theory for the User, Englewood Cliffs, New Jersey: Prentice-Hall, 1987.

[9] J. Moody, "Note on Generalization, Regularization, and Architecture Selection in Nonlinear Learning Systems," in B.H. Juang, S.Y. Kung & C.A. Kamm (eds.) Proceedings of the first IEEE Workshop on Neural Networks for Signal Processing, Piscataway, New Jersey: IEEE, pp. 1–10, 1991.

[10] N. Murata, S. Yoshizawaand & S. Amari, "Network Information Criterion — Determining the Number of Hidden Units for an Artificial Neural Network Model," IEEE Transactions on Neural Networks, vol. 5, no. 6, pp. 865–872, Nov. 1994.

[11] G.A.F. Seber & C.J. Wild, Nonlinear Regression, New York, New York: John Wiley & Sons, 1989.

[12] M. Stone, "Cross-validatory Choice and Assessment of Statistical Predictors," Journal of the Royal Statistical Society B, vol. 36, no. 2, pp. 111–147, 1974.

[13] G.T. Toussaint, "Bibliography on Estimation of Misclassification," IEEE Transactions on Information Theory, vol. 20, no. 4, pp. 472–479, July 1974.

[14] V.N. Vapnik, Estimation of Dependences Based on Empirical Data, New York, New York: Springer-Verlag, 1982.

[15] V.N. Vapnik, "Principles of Risk Minimization for Learning Theory," in J.E. Moody, S.J. Hanson, R.P. Lippmann (eds.) Advances in Neural Information Processing Systems 4, Proceedings of the 1991 Conference, San Mateo, California: Morgan Kaufmann Publishers, 1992, pp. 831–838.

[16] H. White, "Learning in Artificial Neural Networks: A Statistical Perspective," Neural Computation, vol. 1, pp. 425–464, 1989.

[17] H. White, "Consequences and Detection of Misspecified Nonlinear Regression Models," Journal of the American Statistical Association, vol. 76, no. 374, pp. 419–433, June 1981.