Presenting and Analyzing the Results of AI Experiments: Data Averaging and Data Snooping

C. Lee Giles and Steve Lawrence

NEC Research Institute, 4 Independence Way, Princeton NJ 08540
{giles,lawrence}@research.nj.nec.com

Abstract

Experimental results reported in the machine learning AI literature can be misleading. This paper investigates the common processes of data averaging (reporting results in terms of the mean and standard deviation of the results from multiple trials) and data snooping in the context of neural networks, one of the most popular AI machine learning models. Both of these processes can result in misleading results and inaccurate conclusions. We demonstrate how easily this can happen and propose techniques for avoiding these very important problems. For data averaging, common presentation assumes that the distribution of individual results is Gaussian. However, we investigate the distribution for common problems and find that it often does not approximate the Gaussian distribution, may not be symmetric, and may be multimodal. We show that assuming Gaussian distributions can significantly affect the interpretation of results, especially those of comparison studies. For a controlled task, we find that the distribution of performance is skewed towards better performance for smoother target functions and skewed towards worse performance for more complex target functions. We propose new guidelines for reporting performance which provide more information about the actual distribution (e.g. box-whiskers plots). For data snooping, we demonstrate that optimization of performance via experimentation with multiple parameters can lead to significance being assigned to results which are due to chance. We suggest that precise descriptions of experimental techniques can be very important to the evaluation of results, and that we need to be aware of potential data snooping biases when formulating these experimental techniques (e.g. selecting the test procedure). Additionally, it is important to only rely on appropriate statistical tests and to ensure that any assumptions made in the tests are valid (e.g. normality of the distribution).

Introduction

It is known that the analysis and presentation of AI machine learning simulation results needs to be done carefully. For the specific case of neural networks, it has been recognized that experimental evaluation needs improvement (Prechelt 1996; Flexer 1995). Current recommendations include the reporting of the mean and standard deviation of results from a number of trials (herein called "data averaging"), and the computation of statistical tests such as the t-test for performance comparisons (Flexer 1995). However, these recommendations assume that the distribution of results from multiple trials is Gaussian, and that potential biases such as "data snooping" are taken into account. The first part of this paper shows that the distribution of results may differ significantly from a Gaussian distribution, and that the common procedure of reporting the mean and standard deviation of a number of trials can lead to misleading or incorrect conclusions. The second part of this paper investigates data snooping, and shows that this can also lead to incorrect conclusions.

While the experiments contained in this paper are done with neural networks, the issues raised and the guidelines presented are much broader and relevant to other AI machine learning paradigms.

Data Averaging – Presenting the Results of Multiple Trials

In this part of the paper we investigate the common practice of reporting neural network simulation results in terms of the mean and standard deviation of multiple trials. We first provide some background on the neural network training problem and descriptive statistics.

Complexity of Neural Network Training

The performance of a neural network simulation is the result of a training process and it is therefore of interest to consider the properties of the training problem. In general, the training problem of a multi-layer perceptron (MLP) neural network is NP-complete (Faragó & Lugosi 1993; Blum & Rivest 1992), i.e. in general, there is no algorithm capable of finding the optimal set of parameters which has computation time that is bounded by a polynomial in d, the input dimension. A typical compromise is to use an iterative optimization technique such as backpropagation (BP). In most cases, such techniques are only guaranteed to find a local minimum of the cost function. When the problem and the training algorithm make it hard to find a globally

optimal solution, it may be difficult to predict the expected quality or the distribution of the solutions found. In such cases, there is typically no reason to expect that the distribution of results will always be Gaussian, and therefore the actual distribution is of interest.

Performance Measures

The typical method of assessing performance (by running multiple simulations, each beginning from a different starting point in weight space, and reporting the mean and standard deviation of the results (Flexer 1995)) is most suitable when the distribution of the results is Gaussian. For example, if a particular network and training algorithm has a distribution of results which is skewed or multimodal, this will not be observed using the mean and standard deviation. In this case, an alternative method of describing the results can provide a more accurate understanding of the true nature of the performance of the network and the algorithm.

Descriptive Statistics

Median and Interquartile Range We will use the median and the interquartile range (IQR) in the following sections. The median and the interquartile range (IQR) are simple statistics which are not as sensitive to outliers as the commonly used mean and standard deviation (Weiss & Hassett 1987). The median is the value in the middle when arranging the distribution in order from the smallest to the largest value. If we divide the data into two equal groups about the median, then the IQR is the difference between the medians of these groups. The IQR contains 50% of the points. When comparing the mean and the median, both have advantages and disadvantages. The median is often preferred for distributions with outliers, however the mean takes into account the numerical value of every point whereas the median does not. For example, if a student wishes to average exam results of (5, 90, 94, 92) then the mean would be more appropriate. However, for AI machine learning performance distributions we are often interested in the distribution of the individual performance results, rather than the mean performance from a number of trials. More specifically, we may be interested in the probability that a trial will meet a given performance criterion. The median, IQR, minimum and maximum values can provide more information about the distribution of results and, consequently, the nature of the optimization process. Box-whiskers plots incorporate the median, IQR, minimum and maximum values of a distribution (Tukey 1977).

Kolmogorov-Smirnov Test We use the Kolmogorov-Smirnov test in order to test for normality of the distributions. The K-S statistic is (Press *et al.* 1992): $D = \max_{-\infty<x<\infty} |S(x) - N(x)|$ where $S(x)$ is an estimator of the cumulative distribution function of the distribution to test and $N(x)$ is the cumulative distribution function for (in this case) the normal distribution ($\text{erf}(\frac{x}{\sqrt{2}})/2+0.5$ for mean 0 and variance 1). The distribution of the K-S statistic can be approximated for the null hypothesis that the distributions are the same. We can therefore determine the significance level of a given value of D (as a disproof of the null hypothesis that the distributions are the same). The formula is:

$$P = \text{Prob}(D > \text{observed}) = Q_{KS}\left(\left[\sqrt{N} + 0.12 + 0.11/\sqrt{N}\right] D\right) \quad (1)$$

where N is the number of data points and $Q_{KS}(x)$ is:

$$Q_{KS}(x) = 2\sum_{j=1}^{\infty}(-1)^{j-1}e^{-2j^2x^2} \quad (2)$$

P ranges from 0 to 1 and small values of P indicate that the distributions are significantly different (in this case small values of P indicate that the distribution represented by $S(x)$ is significantly different from Gaussian). For the results reported here, $S(x)$ is created from the distribution of results after normalization to zero mean and unit variance.

Empirical Result Distributions

We are primarily interested in the distribution of results for practical problems, and the resulting implications for how results are presented. Therefore, we present the results of a number of experiments using problems that have been commonly used in the neural network literature. In each case, we plot and analyze the distribution of the network error for the training and test data.

Training Details Standard backpropagation was used with stochastic update (update after every training point). Except when specified, all networks are MLPs. All inputs were normalized to zero mean and unit variance. The quadratic cost function was used (Haykin 1994). The learning rate was reduced linearly to zero[1] over the training period from an initial value of 0.1. Performance is reported in terms of the percentage of examples incorrectly classified (for the classification problem) or normalized mean squared error (NMSE).

Phoneme Data These experiments use a database from the ESPRIT ROARS project. The aim of the task is to distinguish between nasal and oral vowels (Verleysen *et al.* 1995). There are 3600 training patterns, 1800 test patterns, five inputs provided by cochlear spectra, and two outputs. Using 10 hidden nodes and 250,000 iterations per trial, the distribution of results is shown in figure 1. It can be observed that the distributions are skewed towards better performance and are a) not Gaussian and b) not symmetric. The K-S test is not used in this case, because the underlying distribution (of classification error) is discrete rather than continuous.

Mackey-Glass The Mackey-Glass equation is a time delay differential equation first proposed as a model of white blood cell production (Mackey & Glass 1977):

$$\frac{dx}{dt} = \frac{ax(t-\tau)}{[1+x^c(t-\tau)]} - bx(t) \quad (3)$$

[1]We have found this to result in similar performance to the "search then converge" learning rate schedules proposed by Darken and Moody (Darken & Moody 1991).

Figure 1. The distribution of classification performance for the networks trained on the phoneme problem. The left hand graph shows the training distribution and the right hand graph shows the test distribution. The abscissa corresponds to the percentage of examples correct and the ordinate represents the percentage of individual results falling within each section of the histogram. The distribution is created from 200 individual simulations with random starting points. Note that the scales change between the graphs.

where the constants are commonly chosen as $a = 0.2$, $b = 0.1$, and $c = 10$. The delay parameter τ determines the behavior of the system (Farmer 1982). For $\tau < 4.53$ there is a stable fixed point attractor. For $4.53 < \tau < 13.3$ there is a stable limit cycle attractor. Period doubling begins at $\tau = 13.3$ and continues until $\tau = 16.8$. For $\tau > 16.8$ the system produces a chaotic attractor. For the experiments reported here, we have chosen $\tau = 30$, and subsampled the series using $\Delta T = 6$. We consider predicting the series one step ahead. For this problem, the results of a number of architectures are compared: MLP, FIR MLP, and IIR MLP (Back 1992). The FIR and IIR MLP networks are similar to the standard MLP except each synapse is replaced by FIR and IIR[2] filters respectively. The FIR and IIR filters in the first layer synapses contained 6 taps (the second layer synapses did not contain FIR/IIR filters) and the MLP networks used an input window of 6. Each network had 5 hidden nodes and was trained for 200,000 updates. There were 1000 training patterns and 1000 test patterns. The FIR and IIR networks were tested both with and without synaptic gains (Back 1992). It is interesting to observe the difference in the distribution of results in this case. When using synaptic gains an extra parameter is inserted into each synapse which multiplies the weighted sum of the individual filter outputs. Altering a synaptic gain is equivalent to altering all of the weights corresponding to the filter taps. The addition of synaptic gains does not affect the representational power of the networks, however it does affect the error surface and the extra degrees of freedom may make optimization easier (Back 1992).

Figure 2 shows the distribution of the normalized mean squared error (NMSE) results. It can be observed that the distribution varies significantly across the various models and that the distributions are often highly skewed and several are multimodal. Figure 3 shows box-whiskers plots and the usual mean and standard deviation plots for these models. The mean minus one standard deviation is actually lower than the best individual error for the two IIR test cases. An observer interpreting the results using the mean and standard deviation along with the assumption that the distributions are approximately Gaussian may be under the impression that a percentage of networks obtained performance better than the mean minus one standard deviation points. However, none of the 100 trials results in such performance for the two IIR test cases. When considering the FIR and IIR synaptic gains networks, significant differences are evident from the distributions and the box-whiskers plots (all distributions are multimodal, however the IIR case is more significantly skewed towards better performance). However, these differences are not clear in the mean and standard deviation which is similar for these two cases. Also interesting are the significantly different distributions for the FIR and IIR MLP networks with and without synaptic gains. As expected, it can be observed that, in general, the box-whiskers plots are be more informative than the mean plus standard deviation plots, but are not as informative as the actual distributions.

The K-S values (D,P) for the training and test sets respectively are FIR no gains: (0.34,1.3×10^{-10}), (0.39,6.8×10^{-14}), FIR gains: (0.18,0.0035), (0.17,0.0054), IIR no gains: (0.5,≈ 0), (0.5,≈ 0), IIR gains: (0.28,2.5×10^{-7}), (0.28,2.2×10^{-7}), and MLP: (0.066,0.76), (0.052,0.95). All distributions are significantly different from Gaussian except for the MLP case.

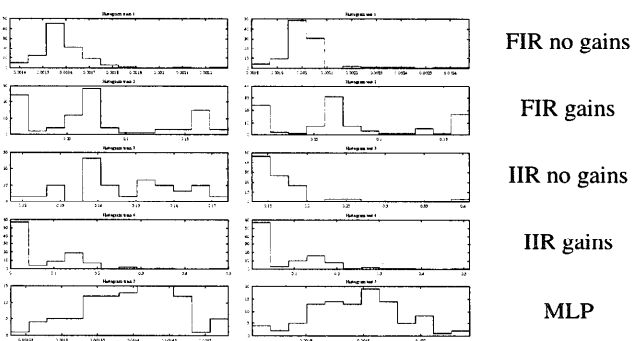

Figure 2. The distribution of the NMSE results for the MLP, FIR MLP, and IIR MLP networks trained on the Mackey-Glass problem. The left hand graphs show the distribution of training errors and the right hand graphs shows the distribution of test errors. The abscissa corresponds to the mean squared error and the ordinate represents the percentage of individual results falling within each section of the histogram. Each distribution is created from 100 individual simulations. The scales are too small to distinguish – see figure 3.

Artificial Task In order to conduct a controlled experiment where we vary the complexity of the target function, we used the following artificial task[3]:

1. An MLP with 5 input nodes, 5 hidden nodes, and 1 output node is initialized with random weights, uniformly selected within a specified range, i.e., w_i in the range $-K$ to K, where w_i are the weights of the network except the biases, and K is a constant. The bias weights are initialized to small random values in the range $(-0.01, 0.01)$.

[2]FIR: Finite Impulse Response, IIR: Infinite Impulse Response.

[3]The task is similar to the procedure used in (Crane *et al.* 1995).

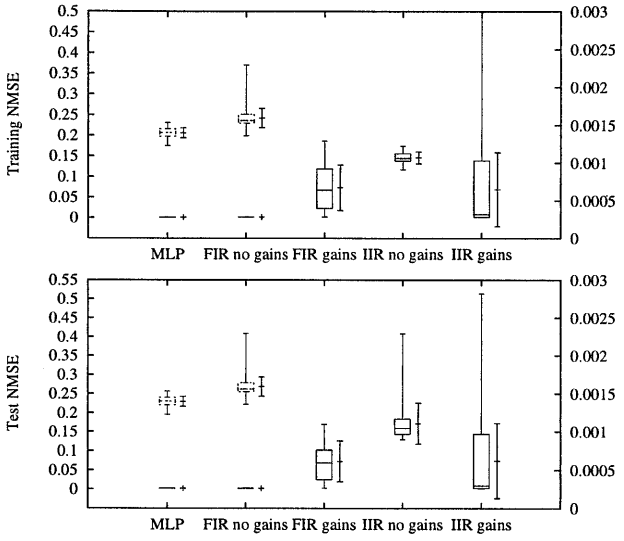

Figure 3. Box-whiskers plots (Tukey 1977) (on the left in each case) for the Mackey-Glass task shown with the mean plus or minus one standard deviation (on the right in each case). For the box-whiskers plots the box corresponds to the IQR, the bar represents the median, and the whiskers extend to the minimum and maximum values. The MLP and FIR no gains cases are compressed in this graph due to the relative poor performance of the other cases. Therefore, the results for these cases have been plotted again using the right hand y scale. These plots can be distinguished with the dotted line used for the box. Notice that a) although the means for the FIR and IIR synaptic gains cases are similar, the median for the IIR MLP networks is much lower, and b) the mean minus one standard deviation for the IIR MLP networks is lower than the best individual networks and actually lower than zero for the synaptic gains case.

In general, as K is increased, the "complexity" of the function mapping is increased.

2. n_{tr} data points are created by selecting random inputs with zero mean and unit variance and propagating them through the network to find the corresponding outputs. This dataset $\mathcal{S}$ forms the training data for subsequent simulations. The procedure is repeated to create a test dataset with n_{te} points. n_{tr} is 1000 and n_{te} is 5000.
3. The training data set $\mathcal{S}$ is used to train new MLPs. The initial weights of these new networks are set using standard procedures (i.e. they are not equal to the weights in the network used to create the dataset). They are initialized on a node by node basis as uniformly distributed random numbers in the range $(-2.4/F_i, 2.4/F_i)$ where F_i is the fan-in of neuron i (Haykin 1994). Each network was trained for 200,000 updates.

Figure 4 shows histograms of the distribution of results for the following four cases: $K = 1, 5, 10, 15$. It can be observed that the distribution of performance is skewed towards better performance for smoother target functions (lower K) and skewed towards worse performance for more complex target functions (higher K), i.e. a general trend

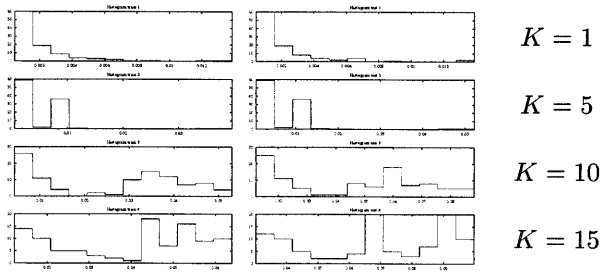

Figure 4. The distribution of errors for the networks trained on the artificial task. From top to bottom, the graphs correspond to K values of 1, 5, 10, and 15. The left hand graphs show the distribution of training errors and the right hand graphs shows the distribution of test errors. The abscissa corresponds to the mean squared error and the ordinate represents the percentage of individual results falling within each section of the histogram. Each distribution is created from 100 individual simulations. The scales are too small to distinguish – see figure 5.

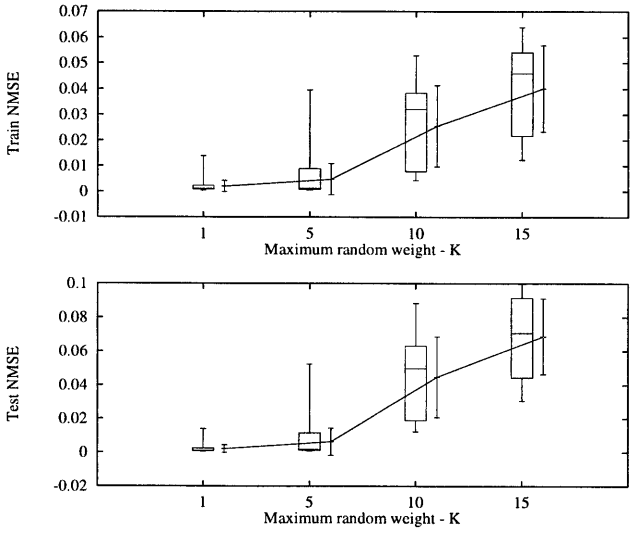

Figure 5. Box-whiskers plots for the artificial task (left in each case) together with the mean and plus or minus one standard deviation (right in each case). From left to right, $K = 1, 5, 10, 15$. We observe that a) the mean minus one standard deviation for $K = 1$ and 5 is lower than the best individual networks, and b) the median moves from being below the mean to being above the mean as K is increased.

can be observed where a higher frequency of the trials resulted in relatively worse performance as K was increased. Note that there is significant multimodality for high K. Figure 5 shows box-whiskers plots and the usual mean and standard deviation plots for these four cases. Note that the mean minus one standard deviation for $K = 1$ and $K = 5$ is actually lower than the best individual error (from 100 trials).

The K-S values (D,P) for the training and test sets respectively are $K = 1$: (0.25,8.3 $\times$ 10^{-6}), (0.25,6.3 $\times$ 10^{-6}),

$K = 5$: (0.26,1.9 $\times$ 10^{-6}), (0.25,3.3 $\times$ 10^{-6}), $K = 10$: (0.17,0.0045), (0.17,0.0071), and $K = 15$: (0.19,0.0015), (0.12,0.088). Hence, all of these distributions are significantly different from Gaussian.

Data Snooping

The previous section presents a case for providing more information about the distribution of results and taking that information into account when drawing conclusions. This section presents a case for providing more information about the experimental process, and using more information when formulating the experimental process, than is commonly done.

Researchers are aware of many forms of experimental bias, e.g. ceiling and floor effects, regression effects, and order effects (Cohen 1995). However, investigation of the machine learning AI literature shows that "data snooping" biases are often ignored. Data snooping commonly refers to the practice of assigning meaning to spurious correlations or patterns. For example, running a simulation with many different parameters and reporting only the best result on an out-of-sample test set, even though this result may be partially or completely due to chance.

In order to demonstrate data snooping, we created a purely random problem, with five uniformly distributed random inputs, and a random two class classification as the output. Training details are as follows: the training and test sets consisted of 100 points, the networks contained three hidden nodes, stochastic backpropagation was used for 100,000 updates, and the learning rate was reduced linearly to zero from an initial value of 0.1 with a linearly reducing learning rate. Ten simulations were performed for each setting of parameters.

Figure 6 shows the results of repeating the problem 20 times on different random training and test sets. Overall, we observe that the performance varies around 50% correct classification, as expected. If we compare each of the results against the null hypothesis of a Gaussian distribution with mean 50 and the same variance using the commonly recommended t-test, then we find that 30% of the results are significantly different at p = 1%, 45% at p = 5% and 50% at p = 10%, i.e. if we only ever ran one of these simulations, there would be a 45% chance that the results would be significantly different from random at the 5% level of significance! Note that this is different from doing all of these tests and selecting the most significant one which is of course not valid – comparing multiple groups requires different tests, e.g. ANOVA and Tukey's HSD. What is wrong? The t-tests are not always appropriate – the distributions are significantly different from normal 30% of the time according to Shapiro-Wilks tests, and the pairwise variance of the distributions varies significantly 25% of the time according to f-tests. Clearly, care must be taken in interpreting the results and formulating tests.

Figure 7 shows the results of testing a range of learning rates, a procedure which is often done and seen summarized in the literature as "We optimized the learning rate". Clearly, selection of the optimal value here would allow us to present results which are significantly better than chance. This is a prime example of data snooping and the principle recommendation (well known to some) is that the result of the optimization (the "optimal" learning rate) should be tested on an additional test set of unseen data, i.e. it is possible to tune the learning algorithm to the first out-of-sample test set.

Figure 8 shows the results of a comparison study which compares the performance of different neural networks. In this case, we modify the random task slightly by assuming that the problem is temporal and we perform training with a variety of recurrent networks as well as the standard TDNN network (the order of the networks is 5, however details of the models are not important here). Once again, there is an overabundance of "statistically significant" differences between the algorithms, e.g. the Gamma network is "better" than the Elman, NP, WZ, and FIR at the 1% level of significance according to t-tests. Problems with drawing a conclusion such as "The Gamma network is better than..." include the fact that we are concentrating on a particular comparison out of a number of possible comparisons, and the potential inappropriateness of t-tests as above.

Precise description of experimental techniques can therefore be very important to the evaluation of results, and we need to be aware of potential data snooping biases when formulating these experimental techniques. Additionally, it is important to only rely on appropriate statistical tests (e.g. ANOVA and Tukey's HSD when comparing multiple groups) and to ensure that any assumptions made in the tests are valid (e.g. normality of the distribution).

Note that the observed difficulties in the example above can be reduced greatly by using a larger test set size.

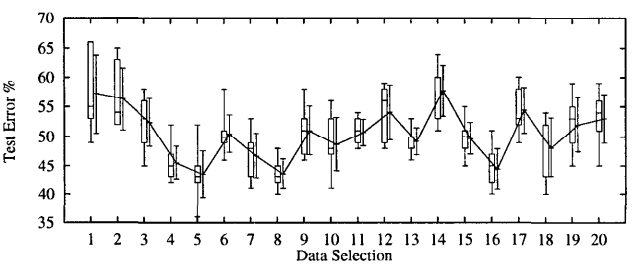

Figure 6. Results on the random problem as different random data sets are used.

Conclusions

Publications commonly report the performance of neural networks using the mean and variance of a number of simulations with different starting conditions. Other papers recommend reporting confidence intervals using Gaussian or t-distributions and testing the significance of comparisons using the t-test (Flexer 1995). However, these as-

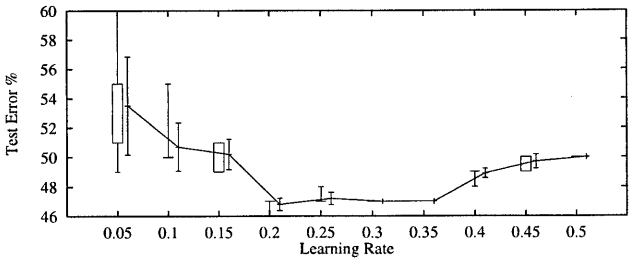

Figure 7. Results on the random problem as the learning rate is varied.

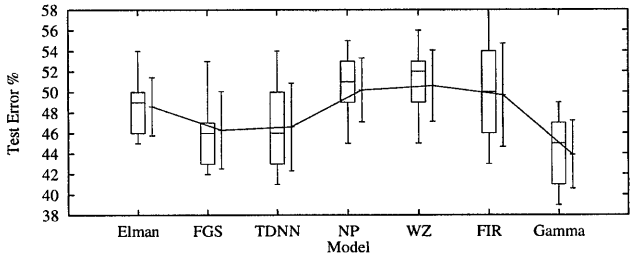

Figure 8. Results on the random problem for different network types.

sume symmetric distributions. The distribution of results for neural network simulations can vary widely depending on the architecture, data, and training algorithm. Comparisons based on the mean and standard deviation of simulation results can therefore be misleading if the observer assumes the distributions are Gaussian. Alternative means of presenting results can be more informative. For example, it is possible to obtain an indication of how often a particular network and algorithm will produce an acceptable result. In a practical situation, the distribution of results can affect the desirable number of trials, e.g. if the results of multiple trials do not vary greatly, then it may be reasonable to use a smaller number of trials. Our recommendations are:

1. Plot the distribution of results for visual inspection. Distributions can be significantly multimodal and neither the mean plus standard deviation or box-whisker plots show the complete picture.
2. Use the median, interquartile range, minimum and maximum values as well as the mean and standard deviation for interpreting results. When plotting results, use box-whiskers plots.
3. In certain cases it may be possible to approximate a normal distribution by removing outliers. For the case where a relatively small number of trials result in comparatively poor convergence, the practice of removing those trials from the statistics and reporting the percentage of "failed" trials would appear reasonable.

It may sometimes be difficult to perform enough simulations in order to accurately characterize the distribution of performance within a reasonable time. Therefore it may not always be possible to follow these recommendations.

Even if the distribution of results is taken into account when presenting and analyzing results, there are still many avenues for the presentation of misleading results or inaccurate conclusions. One that is often not considered is the possibility of data snooping biases. We demonstrated that common procedures for performing experiments and analyzing results can result in incorrect conclusions due to data snooping. We suggest that precise description of experimental techniques can be very important to the evaluation of results, and that we need to be aware of possible data snooping biases when formulating these experimental techniques. Additionally, it is important to only rely on appropriate statistical tests and to ensure that any assumptions made in the tests are valid (e.g. normality of the distribution).

References

Back, A. 1992. *New Techniques for Nonlinear System Identification: A Rapprochement Between Neural Networks and Linear Systems*. Ph.D. Dissertation, Department of Electrical Engineering, University of Queensland.

Blum, A., and Rivest, R. 1992. Training a 3-node neural network is NP-complete. *Neural Networks* 5(1):117–127.

Cohen, P. 1995. *Empirical Methods for Artificial Intelligence*. Cambridge, Massachussets: MIT Press.

Crane, R.; Fefferman, C.; Markel, S.; and Pearson, J. 1995. Characterizing neural network error surfaces with a sequential quadratic programming algorithm. In *Machines That Learn*.

Darken, C., and Moody, J. 1991. Note on learning rate schedules for stochastic optimization. In Lippmann, R.; Moody, J.; and Touretzky, D. S., eds., *Advances in Neural Information Processing Systems*, volume 3. San Mateo, CA: Morgan Kaufmann. 832–838.

Faragó, A., and Lugosi, G. 1993. Strong universal consistency of neural network classifiers. *IEEE Transactions on Information Theory* 39(4):1146–1151.

Farmer, J. D. 1982. Chaotic attractors of an infinite-dimensional dynamical system. *Physica* 4D:366.

Flexer, A. 1995. Statistical evaluation of neural network experiments: Minimum requirements and current practice. Technical Report OEFAI-TR-95-16, The Austrian Research Institute for Artificial Intelligence, Schottengasse 3, A-1010 Vienna, Austria.

Haykin, S. 1994. *Neural Networks, A Comprehensive Foundation*. New York, NY: Macmillan.

Mackey, M., and Glass, L. 1977. Oscillation and chaos in physiological control systems. *Science* 197:287.

Prechelt, L. 1996. A quantitative study of experimental evaluations of neural network learning algorithms. *Neural Networks* 9:457–462.

Press, W.; Teukolsky, S.; Vetterling, W.; and Flannery, B. 1992. *Numerical Recipes*. Cambridge: Cambridge University Press, second edition.

Tukey, J. 1977. *Exploratory Data Analysis*. Reading, MA: Addison-Wesley.

Verleysen, M.; Voz, J.; Thissen, P.; and Legat, J. 1995. A statistical neural network for high-dimensional vector classification. In *Proceedings of the IEEE International Conference on Neural Networks, ICNN 95*. Perth, Western Australia: IEEE.

Weiss, N., and Hassett, M. 1987. *Introductory Statistics*. Reading, Massachusetts: Addison-Wesley, second edition.