Communicated by Steven Nowlan

Bias/Variance Analyses of Mixtures-of-Experts Architectures

Robert A. Jacobs
Department of Brain and Cognitive Sciences, University of Rochester, Rochester, NY 14627 USA

This article investigates the bias and variance of mixtures-of-experts (ME) architectures. The variance of an ME architecture can be expressed as the sum of two terms: the first term is related to the variances of the expert networks that comprise the architecture and the second term is related to the expert networks' covariances. One goal of this article is to study and quantify a number of properties of ME architectures via the metrics of bias and variance. A second goal is to clarify the relationships between this class of systems and other systems that have recently been proposed. It is shown that in contrast to systems that produce unbiased experts whose estimation errors are uncorrelated, ME architectures produce biased experts whose estimates are negatively correlated.

1 Introduction

Many researchers recently have studied statistical models that estimate the value of a random variable by combining the estimates of other models, henceforth referred to as experts. Theoretical and experimental results have established that when the experts are unbiased estimators, combination procedures are most effective when the experts' estimates are negatively correlated; they are moderately effective when the experts are uncorrelated and only mildly effective when the experts are positively correlated.

As an example of an analysis of combinations that use unbiased experts, Clemen and Winkler (1985) quantified the utility of positively correlated, uncorrelated, and negatively correlated experts when the experts' estimates are combined via Bayes' rule. These investigators represented the statistical dependence among the experts by the dependence among their estimation errors and then considered the number of independent experts that carry the same amount of information as a given number of dependent experts. Given certain assumptions, Clemen and Winkler showed that the number of independent experts that are worth the same as an infinite number of positively correlated experts is equal to the inverse of the correlation among the experts. This limit is surprisingly low. If, for instance, the correlation among the experts is 0.5, then an infinite number of such experts carry as much information as only two independent experts. After the first expert, which is worth one independent expert, all other experts combined are worth only

© 1997 Massachusetts Institute of Technology

Downloaded from http://direct.mit.edu/neco/article-pdf/9/2/369/813588/neco.1997.9.2.369.pdf by guest on 18 October 2021

one additional independent expert. The opposite situation is found when negative dependence among the experts is considered. That is, whereas positive dependence among the experts diminishes the effectiveness of a combination procedure, negative dependence increases it.

As a consequence of these types of theoretical results, as well as of empirical studies that show that these general findings also hold in practice (Perrone 1993; Hashem 1993), several researchers have investigated architectures and training procedures for obtaining unbiased experts whose estimation errors are uncorrelated (Meir 1994; Raviv and Intrator, in press; Tresp and Taniguchi 1995). In contrast, this article studies a class of architectures, known as mixtures-of-experts (ME) architectures, that adopt a very different strategy. The analyses are conducted by estimating the bias and variance of these models under a variety of conditions. Based on a result due to Meir (1994), it is shown that the architecture's variance can be expressed as the sum of two terms: the first related to the variances of the experts that comprise the architecture and the second related to the experts' covariances. One goal of this article is to study and quantify a number of properties of ME architectures via the metrics of bias and variance. Because these systems consist of several interacting components, their performance properties can be difficult to understand in a rigorous way. The analyses presented here are useful in this regard. A second goal is to clarify the relationships between this class of architectures and other architectures that have recently been proposed. Instead of producing unbiased experts whose estimation errors are uncorrelated, ME architectures produce biased experts whose estimates are negatively correlated.

The article is organized as follows. Section 2 briefly overviews the ME architecture and its associated learning procedure. Section 3 presents the equations for computing the bias and variance of these architectures. Section 4 presents the results of estimating the biases and variances of ME architectures operating with different numbers of components, operating with different parameter settings, and operating in different noise environments.

2 Mixtures-of-Experts Architectures

The architectures studied in this article are members of the ME family of architectures. This family is of interest on both theoretical and empirical grounds. From a theoretical viewpoint, the architectures combine aspects of finite mixture models and generalized linear models, two well-studied statistical frameworks (Jordan and Jacobs 1994; McCullagh and Nelder 1989; McLachlan and Basford 1988). From an empirical viewpoint, they have been shown to be capable of comparatively fast learning and good generalization on a wide variety of regression and classification tasks (Jacobs *et al.* 1991; Jordan and Jacobs 1994; Nowlan and Hinton 1991; Waterhouse and Robinson 1994).

Downloaded from http://direct.mit.edu/neco/article-pdf/9/2/369/813588/neco.1997.9.2.369.pdf by guest on 18 October 2021

ME architectures are multinetwork, or modular, architectures that combine aspects of competitive and associative learning. Different architectures within this framework may be formed by placing the networks in different structural arrangements. A probabilistic interpretation exists such that for each arrangement of networks, there is a corresponding likelihood function that characterizes an architecture's performance. Learning occurs by maximizing the likelihood function.

ME architectures attempt to solve problems using a "divide-and-conquer" strategy; that is, complex problems are decomposed into a set of simpler subproblems. It is assumed that the data can be adequately summarized by a collection of functions, each defined over a local region of the input space. ME architectures adaptively partition the input space into possibly overlapping regions and allocate different networks to summarize the data located in different regions. This section briefly overviews the ME architectures used in this article. More extensive discussions of ME architectures can be found in Jacobs *et al.* (1991), Jacobs and Jordan (1991, 1993), Jordan and Jacobs (1994), Jordan and Xu (1993), Nowlan and Hinton (1991), and Peng *et al.* (1996).

For the purposes of this article, it is assumed that the data are generated by a number of different probabilistic rules. Let $D = \{(\mathbf{x}^{(t)}, y^{(t)})\}$ denote the collection of training data, where $\mathbf{x}$ is an input vector, y is a target output, and t is a time index. At each time step, a rule is selected from a conditional multinomial distribution with probability $g_i^{(t)} = p(i|\mathbf{x}^{(\mathbf{t})}, V)$, where i indexes a rule and $V = [\mathbf{v}_1, \ldots, \mathbf{v}_I]$ is the matrix of parameters underlying the distribution. The selected rule generates an output y with probability $p(y^{(t)}|\mathbf{x}^{(t)}, U_i, \Phi_i)$, where U_i is a parameter matrix and Φ_i represents other (possibly nuisance) parameters. In the case of regression, each rule is characterized by a statistical model of the form $y = f_i(\mathbf{x}) + \epsilon_i$, where $f_i(\mathbf{x})$ is a fixed linear function of the input vector, and ϵ_i is a random variable. If it is assumed that ϵ_i is gaussian with a mean of zero and a variance of σ_i^2, and if it is assumed that the data are independent and identically distributed, then the likelihood of generating the data is proportional to the finite mixture density

$$L(\mathbf{\Theta}|D) \propto \prod_t \sum_i p(i|\mathbf{x}^{(t)}, V)p(\mathbf{y}^{(t)}|\mathbf{x}^{(t)}, U_i, \Phi_i) \tag{2.1}$$

$$\propto \prod_t \sum_i g_i^{(t)} \, \frac{1}{\sigma_i} e^{-\frac{1}{2\sigma_i^2}(y^{(t)}-\mu_i^{(t)})^2} \tag{2.2}$$

where $\mu_i^{(t)} = f_i(\mathbf{x}^{(t)})$, and $\mathbf{\Theta} = [\mathbf{v}_1, \ldots, \mathbf{v}_I, U_1, \ldots, U_I, \Phi_1, \ldots, \Phi_I]^T$ is the matrix of all parameters.

The ME architectures studied here consist of two types of networks: a gating network and a number of expert networks. The gating network models the input-dependent multinomial distribution used to select a rule. The

Downloaded from http://direct.mit.edu/neco/article-pdf/9/2/369/813588/neco.1997.9.2.369.pdf by guest on 18 October 2021

expert networks model the input-dependent statistical models associated with the different rules. At each time step, all networks receive the vector **x** as input. The output of expert network i, denoted μ_i, is a linear function of its input. The outputs of the gating network are computed using the "softmax" function (Bridle 1989); specifically, the activation of the ith output unit of the gating network, denoted g_i, is

$$g_i = \frac{e^{s_i/\tau}}{\sum_{j=1}^{I} e^{s_j/\tau}}, \tag{2.3}$$

where τ is a temperature parameter, s_i denotes the weighted sum of unit i's inputs, and I denotes the number of expert networks. The output of the architecture as a whole, given by

$$\mu = \sum_{i=1}^{I} g_i \, \mu_i, \tag{2.4}$$

is a convex combination of the experts' outputs. The parameters of the gating and experts networks are adapted so as to maximize the likelihood function given in equation 2.2. This maximization is performed using the expectation-maximization (EM) algorithm (see Jordan and Jacobs 1994 for the EM equations).

3 Bias/Variance Measures

The expected squared error of an estimator on a particular data item may be expressed as the sum of two terms,

$$E[(y-\mu)^2|\mathbf{x}, D] = E[(y - E[y|\mathbf{x}])^2 \mid \mathbf{x}, D] + (\mu - E[y \mid \mathbf{x}]|\mathbf{x}, D)^2, \tag{3.1}$$

where **x** is the input vector, y is the target output, μ is the estimator's approximation, D is the set of training data used to train the estimator, and E is the expectation operator taken with respect to the probability distribution $p(y|\mathbf{x})$. Because the first term does not depend on the data, only the second term is considered below. The expected value of the second term with respect to the data can also be written as the sum of two terms,

$$E_D[(\mu - E[y|\mathbf{x}])^2] = (E_D[\mu] - E[y|\mathbf{x}])^2 + E_D[(\mu - E_D[\mu])^2], \tag{3.2}$$

where E_D is the expectation operator taken with respect to the data. The first term is the square of the bias of an estimator, and the second term is the estimator's variance.

Downloaded from http://direct.mit.edu/neco/article-pdf/9/2/369/813588/neco.1997.9.2.369.pdf by guest on 18 October 2021

ME architectures produce an estimate of a target output by linearly combining the estimates of several experts. Consequently, the bias and variance of an ME architecture can be written in terms of the gating network outputs (the linear coefficients) and the expert network outputs. This leads to an interesting decomposition in the case of the variance of an ME architecture. Based on a result due to Meir (1994), the variance may be expressed as the sum of two terms,

$$E_D[(\mu - E_D[\mu])^2] = E_D\left[\sum_i (g_i\,\mu_i - E_D[g_i\,\mu_i])^2\right] + E_D\left[\sum_i \sum_{i \neq j} (g_i\,\mu_i - E_D[g_i\,\mu_i])\,(g_j\,\mu_j - E_D[g_j\,\mu_j])\right], \tag{3.3}$$

where the first term is the variance of the weighted outputs of the individual experts, and the second term is the covariance of the experts' weighted outputs.

Combining equations 3.2 and 3.3, the expected squared difference between the estimate of an ME architecture and the expected value of a regression function is the sum of the architecture's squared bias and the variance and covariance of the experts' weighted outputs. It is possible to gain insight into the performance characteristics of ME architectures by analyzing these systems in terms of these quantities.

4 Simulation Results

This section reports the results of estimating the bias of ME architectures and the variance and covariance of experts' weighted outputs on a regression task. The regression function is

$$f(\mathbf{x}) = \frac{1}{13}\left[10\,\sin(\pi x_1 x_2) + 20\left(x_3 - \frac{1}{2}\right)^2 + 10\,x_4 + 5\,x_5\right] - 1\,, \tag{4.1}$$

where $\mathbf{x} = [x_1, \ldots, x_5]$ is an input vector whose components lie between zero and one. The value of $f(\mathbf{x})$ lies in the interval $[-1, 1]$. Target outputs are created by adding noise sampled from a gaussian distribution with a mean of zero and a variance of σ^2 to the function f. This regression task is a linearly scaled version of a task that has been used by other investigators to evaluate statistical estimators (e.g., Friedman 1991; Friedman *et al.* 1983).

Twenty-five training sets were created. Each set consisted of 500 input-output patterns in which the components of the input vectors were independently sampled from a uniform distribution over the interval (0, 1). A test set of 1024 input-output patterns was also created. For this set, the input vectors were uniformly spaced in the five-dimensional input space, and the target

Downloaded from http://direct.mit.edu/neco/article-pdf/9/2/369/813588/neco.1997.9.2.369.pdf by guest on 18 October 2021

outputs were not corrupted by noise. Twenty-five simulations of each architecture were conducted. In each simulation, the architecture was trained on a different training set. Simulations lasted for 60 epochs, and architectures were evaluated on the test set at every fifth epoch. The variance parameter associated with each expert network was set to the variance of the noise added to the training data. An architecture was initialized with the same set of small, random weights at the start of each simulation. Consequently, different simulations of an architecture yielded different performances solely due to the use of different training sets.

The equations for estimating the quantities of interest are closely related to the equations for estimating the bias and variance of neural networks presented by Geman *et al.* (1992). The average output of an architecture on pattern t from the test set, denoted $\overline{\mu}^{(t)}$, is given by

$$\overline{\mu}^{(t)} = \frac{1}{N} \sum_{n=1}^{N} \mu^{(t,n)}, \tag{4.2}$$

where $\mu^{(t,n)}$ is the output on the nth simulation and N is the number of simulations. Similarly, the average weighted output of expert network i on pattern t, denoted $\overline{g_i^{(t)} \mu_i^{(t)}}$, is given by

$$\overline{g_i^{(t)} \mu_i^{(t)}} = \frac{1}{N} \sum_{n=1}^{N} g_i^{(t,n)} \mu_i^{(t,n)}, \tag{4.3}$$

where $g_i^{(t,n)}$ is the activation of the ith unit of the gating network on simulation n, and $\mu_i^{(t,n)}$ is the output of expert network i on simulation n. Define the integrated bias, meaning the squared bias averaged over the set of input-output patterns in the test set, to be

$$\text{Integrated bias} \equiv \frac{1}{T} \sum_{t=1}^{T} |\overline{\mu}^{(t)} - y^{(t)}|^2, \tag{4.4}$$

where $y^{(t)}$ is the target output on pattern t and T is the total number of patterns. The integrated variance and integrated covariance of the experts' weighted outputs are defined in analogous ways:

$$\text{Integrated variance} \equiv \sum_i \frac{1}{T} \sum_{t=1}^{T} \frac{1}{N} \sum_{n=1}^{N} (g_i^{(t,n)} \mu_i^{(t,n)} - \overline{g_i^{(t)} \mu_i^{(t)}})^2 \tag{4.5}$$

$$\text{Integrated covariance} \equiv$$

$$\sum_i \sum_{j \neq i} \frac{1}{T} \sum_{t=1}^{T} \frac{1}{N} \sum_{n=1}^{N} (g_i^{(t,n)} \mu_i^{(t,n)} - \overline{g_i^{(t)} \mu_i^{(t)}})\,(g_j^{(t,n)} \mu_j^{(t,n)} - \overline{g_j^{(t)} \mu_j^{(t)}})\,. \tag{4.6}$$

Downloaded from http://direct.mit.edu/neco/article-pdf/9/2/369/813588/neco.1997.9.2.369.pdf by guest on 18 October 2021

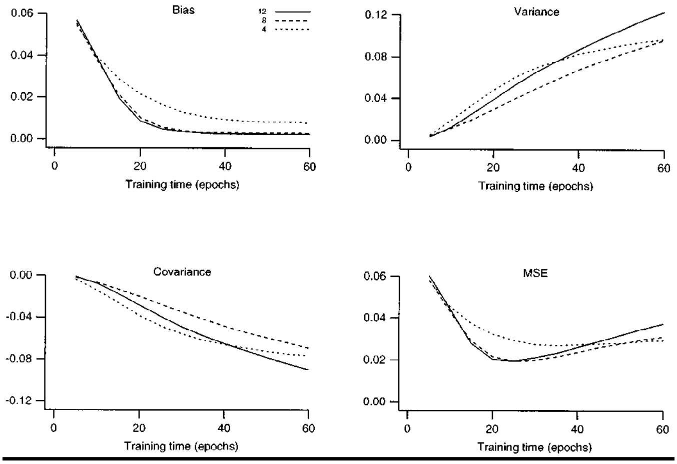

Figure 1: Integrated bias, variance, covariance, and mean-squared error of architectures with 4, 8, or 12 expert networks.

It is also useful to define the integrated mean-squared error (MSE):

$$\text{Integrated MSE} \equiv \frac{1}{T}\sum_{t=1}^{T}\frac{1}{N}\sum_{n=1}^{N}|\mu^{(t,n)} - y^{(t)}|^2. \qquad (4.7)$$

Equivalently, the integrated MSE can be expressed as the sum of the integrated bias, variance, and covariance.

The first experiment that we conducted evaluated ME architectures with different numbers of expert networks. Systems with 4, 8, and 12 experts were used. The temperature parameter in the softmax activation function used by the gating network (see equation 2.3) was set to 1, and the variance of the noise added to the target function was set to 0.1. The results are presented in Figure 1. The training time is given by the horizontal axis of each graph in this figure. The integrated bias is given by the vertical axis of the upper-left graph; the integrated variance is given by the vertical axis of the upper-right graph; the lower-left graph gives the integrated covariance; and the lower-right graph gives the integrated MSE. In all four graphs, the solid line gives the results for the architecture with 12 experts; the dense dashed line and the sparse dashed line give the results for the architectures with 8 and 4 experts, respectively (a legend is shown in the upper-left graph).

As expected, the integrated bias declined with training time for all architectures. The 12-expert and 8-expert systems have more computational resources than the 4-expert system and thus achieved smaller bias. The in-

Downloaded from http://direct.mit.edu/neco/article-pdf/9/2/369/813588/neco.1997.9.2.369.pdf by guest on 18 October 2021

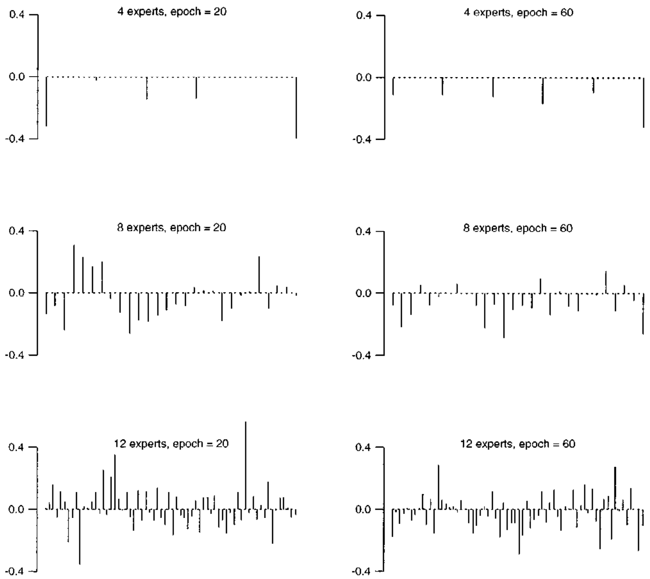

Figure 2: Correlations among the expert networks' outputs for architectures with 4, 8, and 12 experts at epochs 20 and 60.

tegrated variance for all architectures increased with training time. This is particularly true for the system with 12 experts. In contrast, the covariance for all architectures decreased as training progressed. Because the variance increased significantly faster than the covariance decreased, the systems eventually overfit the training data as evidenced by the integrated MSE. Overfitting was most severe for the 12-expert system.

For our purposes, it is important to note that the integrated covariance became negative during the course of training. Although this covariance is based on the expert networks' weighted outputs (as weighted by the gating network), its negative value suggests that it is likely that the experts' unweighted outputs also became negatively correlated. In order to evaluate this hypothesis, we measured the correlations among these outputs (averaged over the 25 simulations of each architecture) at epoch 20 (prior to overfitting) and at epoch 60 (overfitting has occurred). The results are shown in Figure 2. The three rows of bar graphs correspond to the systems with 4, 8, and 12 expert networks; the two columns correspond to epochs 20 and 60. The vertical axis of each graph gives the value of a correlation. A correlation is represented by a vertical bar with one end point at zero and the other end point at the value of the represented correlation.

Downloaded from http://direct.mit.edu/neco/article-pdf/9/2/369/813588/neco.1997.9.2.369.pdf by guest on 18 October 2021

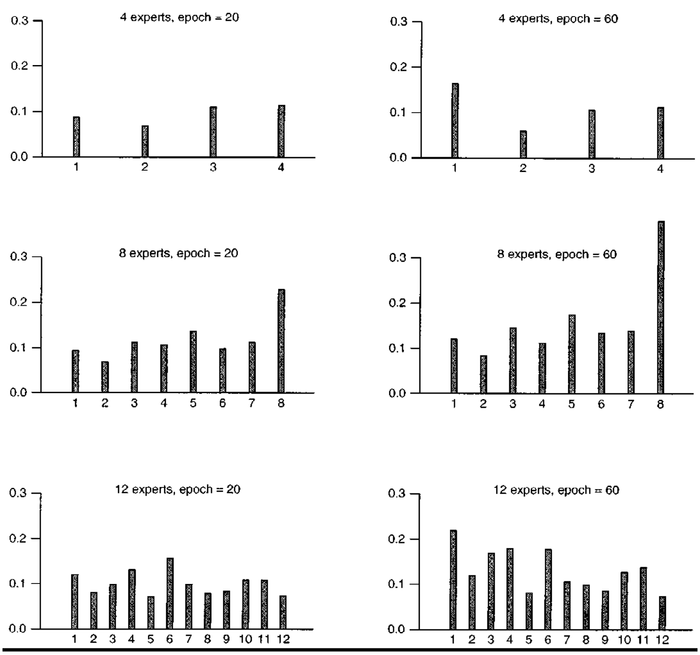

Figure 3: Squared biases of the individual expert networks that comprise architectures with 4, 8, and 12 experts at epochs 20 and 60.

The results suggest that the ME training procedure leads to negatively correlated experts. In addition, the correlations become more negative as training proceeds. The negative correlations stem from the fact that each ME system adaptively partitions the input space into regions such that the target function has different properties in each region. Different experts learn the data located in different regions. Because the correlations for the 4-expert system, which has relatively few computational resources, were all negative, it may be said that the ME architecture was efficient in the sense that it adapted the experts so that different experts provided informationally different "basis" functions. Although some of the correlations of the 8- and 12-expert systems were positive in value, the majority of the correlations were negative.

The ME architecture tends to produce negatively correlated experts and thus should be preferable to systems that produce uncorrelated and unbiased experts only if its experts are also unbiased. To evaluate this possibility, we estimated the squared bias of the individual experts. The results are shown in Figure 3. The bar graphs in this figure correspond to the sys-

Downloaded from http://direct.mit.edu/neco/article-pdf/9/2/369/813588/neco.1997.9.2.369.pdf by guest on 18 October 2021

tems with 4, 8, and 12 expert networks evaluated at epochs 20 and 60. The vertical axis of each graph gives the estimated bias of an individual expert, and different bars correspond to different experts. Despite the fact that the bias of an ME architecture was nearly zero, the squared biases of the individual experts comprising the architecture were positive. The biases of experts in larger architectures were generally greater than those of experts in smaller architectures. Furthermore, the expert biases increased with additional training.

The experiments reported here help clarify the relationships between the performance characteristics of ME architectures and those of some systems studied by other researchers. As noted above, several investigators are seeking training methods for achieving unbiased experts whose estimation errors are uncorrelated (Meir 1994; Raviv and Intrator, in press; Tresp and Taniguchi 1995). As just one example, Raviv and Intrator (in press) use a noisy sampling technique (bootstrapping) in order to create sets of independent training samples. Uncorrelated experts are produced by training different experts on different training sets. In contrast, ME architectures and their associated training procedures adopt a very different strategy. They tend to produce experts that are biased and negatively correlated.

The next experiment studied the effects of varying the temperature parameter in the softmax function used by the gating network. Jordan and Jacobs (1994) claimed that an advantage of ME architectures over other tree-structured statistical estimators, such as CART (Breiman *el al.* 1984) or C4.5 (Quinlan 1993), is that they use soft splits of the input space instead of hard splits, meaning that regions of the input space defined by the architecture can overlap and that data items can lie simultaneously in multiple regions. Estimators that use hard splits are likely to have a larger variance than estimators that use soft splits. In order to verify and quantify this claim, we compared the performances of an ME architecture when the temperature was small ($\tau = 0.025$) to that when the temperature was large ($\tau = 1.0$). The splits of an architecture with a near-zero temperature are more like hard splits, especially during the early stages of training when the gating network's weights are small. The architecture that was used had 8 expert networks. The variance of the noise added to the target function was 0.1.

The results are presented in Figure 4 (this figure has the same format as Fig. 1). The solid line gives the performance of the architecture with a large temperature; the dashed line is for the case of a small temperature. The integrated bias of the large-temperature architecture decreased more slowly than that of the small-temperature architecture during the early stages of training but eventually reached a lower value. Its integrated variance was significantly lower, and its integrated covariance was only moderately higher. Thus, these outcomes are consistent with the claim of Jordan and Jacobs (1994). The MSE of the large-temperature system was smaller. Overall, the results suggest that the system with a low temperature tended to commit early in the training process to a particular partition of the input

Downloaded from http://direct.mit.edu/neco/article-pdf/9/2/369/813588/neco.1997.9.2.369.pdf by guest on 18 October 2021

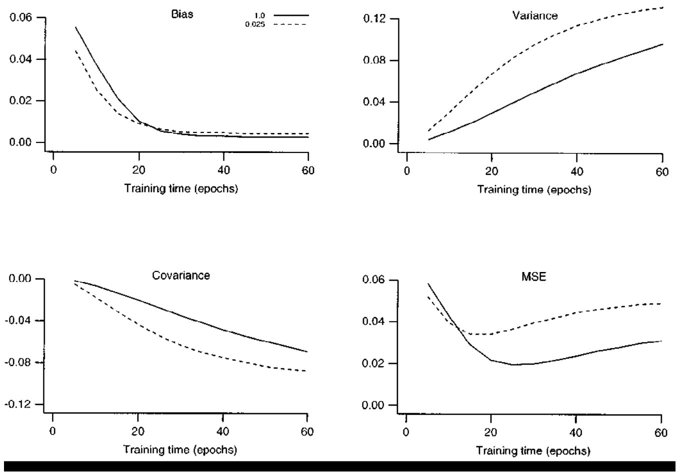

Figure 4: The integrated bias, variance, covariance, and mean-squared error for architectures when the temperature parameter τ was set to 1.0 or 0.025.

space. Because the architecture at this stage of training is highly sensitive to the idiosyncrasies of both its training data and its initial weight settings, the selected partitions frequently lead to poor generalization performance.

We have also analyzed architectures when the variance of the noise added to the target function was varied. Large noise ($\sigma^2 = 0.2$), moderate noise ($\sigma^2 = 0.1$), and small noise ($\sigma^2 = 0.05$) conditions were studied. The ME architecture had 8 expert networks. The temperature τ was set to 1. The results are shown in Figure 5. The solid line is for the large noise condition; the dense and sparse dashed lines correspond to the moderate noise and small noise conditions, respectively. As expected, the integrated bias of the architecture decreased more slowly when the noise was relatively large. The architecture's integrated variance grew slowly during early stages of training and more rapidly during later stages under this condition. Similarly, its covariance decreased slowly initially but then eventually decreased rapidly in the large noise case. The integrated MSE is smallest for the low noise case. In sum, performance was best when the signal-to-noise ratio of the training data was high and degraded gracefully with decreasing values of this ratio.

5 Conclusions

This article has investigated the bias and variance of several ME architectures. It was shown that the variance of an ME architecture can be expressed as the sum of two terms: the first is related to the variances of the experts that

Downloaded from http://direct.mit.edu/neco/article-pdf/9/2/369/813588/neco.1997.9.2.369.pdf by guest on 18 October 2021

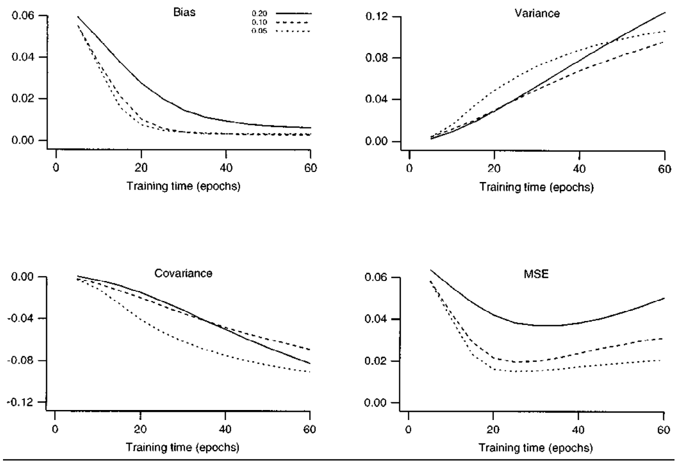

Figure 5: The integrated bias, variance, covariance, and mean-squared error for architectures when the variance of the noise added to the target function was 0.2, 0.1, or 0.05.

comprise the architecture, the second is related to the experts' covariances. One goal of this article was to study and quantify a number of properties of ME architectures. A second goal was to clarify the relationships between this class of systems and other systems that have recently been proposed. In contrast to systems that produce unbiased experts whose estimation errors are uncorrelated, ME architectures produce biased experts whose estimates are negatively correlated.

The fact that ME architectures produce biased experts may be seen as a disadvantage by some readers. My view is that this property is of no consequence so long as the overall ME architecture is unbiased at the end of training. This appears to be the case, as is evidenced in Figures 1, 4, and 5. Nonetheless, it may be tempting to seek ways to modify the architecture or training procedure so as to produce individual experts with less bias. The most obvious possibility is to add hidden units to the expert networks so that these networks could perform nonlinear regressions. Although this practice would reduce the bias of the individual experts, I do not necessarily recommend it. One drawback of adding hidden units to the experts is that the EM algorithm, extremely efficient for optimizing likelihood functions, could no longer be easily used during training. A second drawback of adding hidden units is that this would result in increases in the variances of the individual experts. The statistical community has generally regarded nonzero variance as a more serious problem than nonzero bias as evidenced

Downloaded from http://direct.mit.edu/neco/article-pdf/9/2/369/813588/neco.1997.9.2.369.pdf by guest on 18 October 2021

by the community's focus on the overfitting problem. It is often found that the addition of a small amount of bias to an estimator can result in a larger decrease in an estimator's variance.

The analyses presented here help in understanding a number of regularization procedures that have recently been applied to ME architectures in order to ameliorate the problem of overfitting. One way that overfitting can be lessened is through the use of methods that decrease the variances of the experts. Waterhouse *et al.* (1996) pursued this approach in a Bayesian framework by assigning prior distributions to the weights of each expert. The parameter values for each prior distribution were estimated from the data. This strategy aims to decrease the experts' variances without overly increasing the architecture's bias. Simulation results on two artificial data sets and on a sunspot prediction task showed that the method can be effective in eliminating overfitting. The experts' variances can also be decreased by methods that reduce the number of free parameters in an ME architecture. Jacobs *et al.* (1996) used Bayesian sampling techniques in order to detect and eliminate unnecessary expert networks. Simulation results on a speech recognition task and a breast cancer classification task showed that the method led to improved generalization performances. Waterhouse and Robinson (1996) proposed an algorithm that both adds and deletes networks from an ME architecture during the course of training. A different approach to ameliorating overfitting is to increase the degree to which experts are negatively correlated. Although we are not aware of any studies that have pursued this approach, simulation results of Jacobs and Kosslyn (1994) suggest that it might be achieved through the use of expert networks that each have a different topology or each receive a different set of input variables.

Acknowledgments

I thank M. Jordan for commenting on an earlier version of this article. This work was conducted while I was a visiting scholar at Stanford University. I am grateful to D. Rumelhart and the members of his research laboratory for many interesting discussions. This work was supported by NIH research grant R29-MH54770.

References

Breiman, L., Friedman, J. H., Olshen, R. A., and Stone, C. J. 1984. *Classification and Regression Trees*. Wadsworth International Group, Belmont, CA.

Bridle, J. 1989. Probabilistic interpretation of feedforward classification network outputs, with relationships to statistical pattern recognition. In *Neurocomputing: Algorithms, Architectures, and Applications*, F. Fogelman-Soulie and J. Hérault, eds. Springer-Verlag, New York.

Clemen, R. T., and Winkler, R. L. 1985. Limits for the precision and value of information from dependent sources. *Operations Research* **33**, 427–442.

Downloaded from http://direct.mit.edu/neco/article-pdf/9/2/369/813588/neco.1997.9.2.369.pdf by guest on 18 October 2021

Friedman, J. H. 1991. Multivariate adaptive regression splines. *Annals of Statistics* **19**, 1–141.

Friedman, J. H., Grosse, E., and Stuetzle, W. 1983. Multidimensional additive spline approximation. *SIAM Journal of Scientific Computing* **4**, 291–301.

Geman, S., Bienenstock, E., and Doursat, R. 1992. Neural networks and the bias / variance dilemma. *Neural Computation* **4**, 1–58.

Hashem, S. 1993. *Optimal linear combinations of neural networks*. Tech. Rep. SMS 94–4, School of Industrial Engineering, Purdue University.

Jacobs, R. A., and Jordan, M. I. 1991. A competitive modular connectionist architecture. In *Advances in Neural Information Processing Systems 3,* R. P. Lippmann, J. E. Moody, and D. S. Touretzky, eds. Morgan Kaufmann, San Mateo, CA.

Jacobs, R. A., and Jordan, M. I. 1993. Learning piecewise control strategies in a modular neural network architecture. *IEEE Transactions on Systems, Man, and Cybernetics* **23**, 337–345.

Jacobs, R. A., Jordan, M. I., Nowlan, S. J., and Hinton, G. E. 1991. Adaptive mixtures of local experts. *Neural Computation* **3**, 79–87.

Jacobs, R. A., and Kosslyn, S. M. 1994. Encoding shape and spatial relations: The role of receptive field size in coordinating complementary representations. *Cognitive Science* **18**, 361–386.

Jacobs, R. A., Peng, F., and Tanner, M. A. 1996. A Bayesian approach to model selection in hierarchical mixtures-of-experts architectures. *Neural Networks*. In press.

Jordan, M. I., and Jacobs, R. A. 1994. Hierarchical mixtures-of-experts and the EM algorithm. *Neural Computation* **6**, 181–214.

Jordan, M. I., and Xu, L. 1993. *Convergence results for the EM approach to mixtures-of-experts architectures*. Tech. Rep. 9303, Department of Brain and Cognitive Sciences, MIT.

McCullagh, P., and Nelder, J. A. 1989. *Generalized Linear Models*. Chapman and Hall, London.

McLachlan, G. J., and Basford, K. E. 1988. *Mixture Models: Inference and Applications to Clustering*. Marcel Dekker, New York.

Meir, R. 1994. *Bias, variance, and the combination of estimators: The case of linear least squares*. Tech. Rep. 922, Department of Electrical Engineering, Technion, Haifa, Israel.

Nowlan, S. J., and Hinton, G. E. 1991. Evaluation of adaptive mixtures of competing experts. In *Advances in Neural Information Processing Systems 3*, R. P. Lippmann, J. E. Moody, and D. S. Touretzky, eds. Morgan Kaufmann, San Mateo, CA.

Peng, F., Jacobs, R. A., and Tanner, M. A. 1996. Bayesian inference in mixtures-of-experts and hierarchical mixtures-of-experts models with an application to speech recognition. *Journal of the American Statistical Association*, in press.

Perrone, M. P. 1993. *Improving regression estimation: Averaging methods for variance reduction with extensions to general convex measure optimization*. Ph.D. thesis, Brown University.

Quinlan, J. R. 1993. *C4.5: Programs for Machine Learning*. Morgan Kaufmann, San Mateo, CA.

Downloaded from http://direct.mit.edu/neco/article-pdf/9/2/369/813588/neco.1997.9.2.369.pdf by guest on 18 October 2021

Raviv, Y., and Intrator, N. In press. Bootstrapping with noise: An effective regularization technique.

Tresp, V., and Taniguchi, M. 1995. Combining estimators using non-constant weighting functions. In *Advances in Neural Information Processing Systems 7*, G. Tesauro, D. S. Touretzky, and T. K. Leen, eds. MIT Press, Cambridge, MA.

Waterhouse, S. R., MacKay, D., and Robinson, T. 1996. Bayesian methods for mixtures of experts. In *Advances in Neural Information Processing Systems 8*, D. S. Touretzky, M. C. Mozer, and M. E. Hasselmo, eds. MIT Press, Cambridge, MA.

Waterhouse, S. R., and Robinson, A. J. 1994. Classification using hierarchical mixtures of experts. In *Proceedings of the IEEE Workshop on Neural Networks for Signal Processing*, J. Vloutzos, J.-N. Hwang, and E. Wilson, eds. IEEE Press, New York.

Waterhouse, S. R., and Robinson, A. J. 1996. Constructive algorithms for hierarchical mixtures of experts. In *Advances in Neural Information Processing Systems 8*, D. S. Touretzky, M. C. Mozer, and M. E. Hasselmo, eds. MIT Press, Cambridge, MA.

Received September 5, 1995; accepted April 24, 1996.

Downloaded from http://direct.mit.edu/neco/article-pdf/9/2/369/813588/neco.1997.9.2.369.pdf by guest on 18 October 2021