Selecting Input Variables Using Mutual Information and Nonparametric Density Estimation

Brian V. Bonnlander
Department of Computer Science
and Institute of Cognitive Science
University of Colorado
Boulder, CO 80309-0430
brianb@cs.colorado.edu

Andreas S. Weigend
Department of Computer Science
and Institute of Cognitive Science
University of Colorado
Boulder, CO 80309-0430
andreas@cs.colorado.edu*

Abstract

In learning problems where a connectionist network is trained with a finite sized training set, better generalization performance is often obtained when unneeded weights in the network are eliminated. One source of unneeded weights comes from the inclusion of input variables that provide little information about the output variables. We propose a method for identifying and eliminating these input variables. The method first determines the relationship between input and output variables using nonparametric density estimation and then measures the relevance of input variables using the information theoretic concept of mutual information. We present results from our method on a simple toy problem and a nonlinear time series.

1 INTRODUCTION

Generalization performance on a fixed-size training set is closely related to the number of free parameters in a network. Selecting too many free parameters can lead to poor generalization performance (Baum & Haussler, 1989; Geman, Bienenstock, & Doursat, 1991). A common strategy in designing connectionist networks involves reducing the set of adjustable weights only to those that are relevant to the learning problem (e.g., Weigend, Huberman, & Rumelhart, 1990; Buntine & Weigend, 1991; MacKay, 1992).

One source of unneeded weights comes from input variables that provide little information about the network output to be learned. Knowing which input variables fall into this category can be difficult for problems where the relationships between inputs and outputs are not well understood. For example, in financial forecasting, it can be hard to determine which market indicators are relevant for predicting a quantity such as the exchange rate between the U.S. Dollar and the German Mark.

We propose a method for measuring the information that a group of input variables provides about the outputs. The method can be used to eliminate less useful input variables, thereby reducing the number of network weights and improving generalization performance. The method determines the relationship between inputs and outputs using nonparametric density estimation, and then the relevance of input variables is measured using a formula from information theory known as *mutual information*. We explore some properties of *equal mass binning* as a density estimation technique and show that it works well in conjunction with estimating mutual information. After illustrating some properties of the method on a toy problem, we show that the the method compares well with generalization performance on a nonlinear time series prediction task.

Section 2 introduces the mutual information criterion and justifies its use for measuring input variable relevance. Section 3 describes how we estimate the probability distributions needed for computing mutual information. Section 4 discusses the search procedure we use for finding the best input variables. Section 5 presents simulation results, and the final section outlines directions for further research.

*http://www.cs.colorado.edu/~andreas/Home.html

2 MUTUAL INFORMATION

The starting point for our method is a collection of prospective input variables labeled $\{X_1, X_2, \ldots, X_n\}$; a subset of these input variables is denoted $\mathbf{X}_i$, $i = 1 \ldots 2^n - 1$.[1] A necessary assumption, if we are planning to estimate the relevance of variables by density estimation, is that the members of $\{X_1, X_2, \ldots, X_n\}$ are *random variables*; that is there are functions $P_{\mathbf{X}_i}$ describing, in probabilistic terms, how the patterns in the training set were generated. We also assume that the output variable Y, which is to be predicted in the task, is a random variable.[2] The goal is to find the input variable subset that is most relevant for predicting the output variable Y.

Our method rests on the idea that the relevance of an input variable subset $\mathbf{X}_i$ is captured by the *mutual information* $I(\mathbf{X}_i; Y)$ between the input variables and the output variable (see also Lewis, 1962; Fraser, 1986; Battiti, 1994; Gershenfeld & Weigend, 1994). What $I(\mathbf{X}_i; Y)$ measures is the *reduction in uncertainty* of Y due to knowledge of the input variables $\mathbf{X}_i$ (Cover & Thomas, 1991). The uncertainty of a distribution is defined using the formula for entropy H. If we let $H(Y)$ be the uncertainty of P_Y and $H(Y|\mathbf{X}_i)$ be the uncertainty of the distribution $P_{Y|\mathbf{X}_i}$, then

$$I(\mathbf{X}_i; Y) = H(Y) - H(Y|\mathbf{X}_i) = -\sum_{y \in \mathcal{Y}} P(y) \log P(y) + \sum_{\mathbf{x}_i \in \mathcal{X}_i} \sum_{y \in \mathcal{Y}} P(\mathbf{x}_i, y) \log P(y|\mathbf{x}_i),$$

where $\mathcal{X}_i$ and $\mathcal{Y}$ represent the sets of support for $\mathbf{X}_i$ and Y, and the logarithm is base two. Roughly speaking, the quantity $I(\mathbf{X}_i; Y)$ is large when on average, input patterns with similar values are mapped to output patterns with similar values, and it is small when on average, input patterns with similar values are mapped to output patterns with different values.

Another view of mutual information is that it measures the degree to which $\mathbf{X}_i$ and Y are not independent. With the identity $P(\mathbf{x}_i, y) = P(y|\mathbf{x}_i)P(\mathbf{x}_i)$, the expression for mutual information can also be written as

$$I(\mathbf{X}_i; Y) = \sum_{\mathbf{x}_i \in \mathcal{X}_i} \sum_{y \in \mathcal{Y}} P(\mathbf{x}_i, y) \log \frac{P(\mathbf{x}_i, y)}{P(\mathbf{x}_i)P(y)}.$$

When $\mathbf{X}_i$ and Y are statistically independent, then $P(\mathbf{X}_i, Y) = P(\mathbf{X}_i)P(Y)$, causing the fraction in the formula to equal unity, and thus the value of $I(\mathbf{X}_i; Y)$ is zero. The value of $I(\mathbf{X}_i; Y)$ grows as $\mathbf{X}_i$ and Y become more dependent. The more dependent Y is on $\mathbf{X}_i$, the more information one gains about Y once $\mathbf{X}_i$ is known, and therefore the less uncertain Y is when $\mathbf{X}_i$ is known. Mutual information is equivalent to the Kullback-Leibler distance, or cross-entropy, between the joint distribution $P_{\mathbf{X}_i, Y}$ and the product of the marginal distributions $P_{\mathbf{X}_i}$ and P_Y.[3]

We identify the most relevant input variable subset as the one with the highest mutual information estimate; Section 3 discusses how the probability distributions are created for this estimate.

3 NONPARAMETRIC DENSITY ESTIMATION

The three probability distributions $P_{\mathbf{X}_i}$, P_Y, and $P_{\mathbf{X}_i, Y}$, together describe, in probabilistic terms, the relationship between a set of input variables $\mathbf{X}_i$ and the output Y. The goal of density estimation is to approximate these distributions from a finite set of examples. In this paper, we focus on *nonparametric density estimation*. In contrast to nonparametric density estimation, neural networks do not use a kernel for each data point. Additionally, the parameters in a neural network are more flexible. The advantage of a nonparametric approach is the model's expressive power; it has been shown that many density estimation methods are capable of approximating large classes of distributions, given enough data (Härdle, 1989). There is an extensive literature on density estimation, which includes treatment of discrete nonparametric estimates using histogram based techniques (e.g., Scott, 1992), continuous nonparametric estimates using continuous

[1] The presence or absence of n input variables in a subset can be represented by an n bit binary vector, where a 1 in the vector represents the presence of a variable, and a 0 represents the absence of a variable. Since there are 2^n binary numbers of length n and only one of these numbers is zero, there are $2^n - 1$ subsets containing at least one element.

[2] For clarity of presentation, we only consider the case of a single output value in this paper. The ideas generalize straightforwardly to multiple output values.

[3] The Kullback-Leibler distance between two distributions $p(x)$ and $q(x)$ is equal to $\sum_{x \in \mathcal{X}} p(x) \log (p(x)/q(x))$.

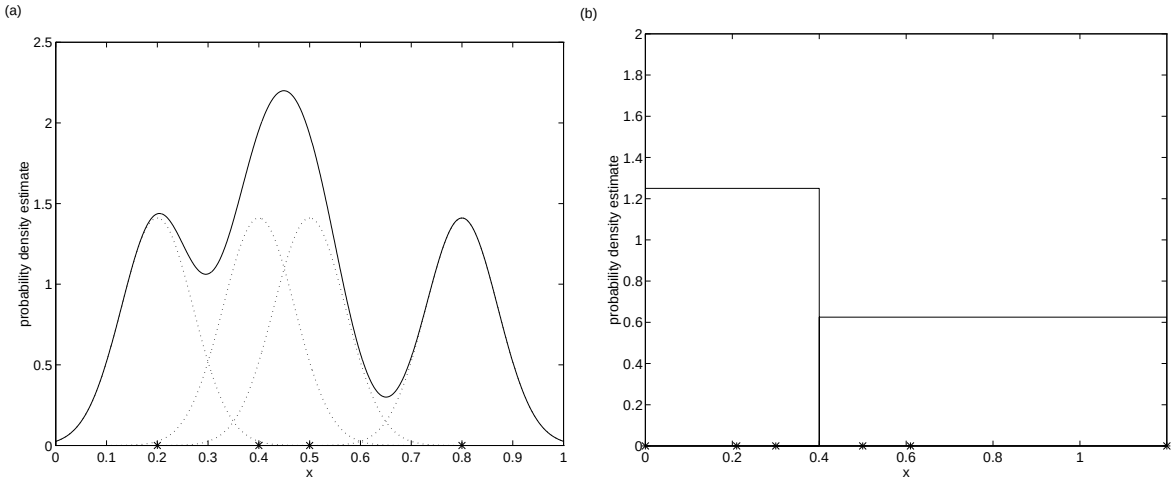

Figure 1: Nonparametric density estimation. (a) Continuous kernels for a simple one dimensional example. A kernel function is centered over every data point (represented by stars in the graph), and the contributions of all kernels are added together to estimate the probability density at any point. (b) Equal mass binning. Bin partitions are chosen to split the data points (stars in the graph) as evenly as possible, and bin heights are found by multiplying the relative number of points in a bin by the bin size. In the example, the area under each bin height is 0.5, since each bin contains half of the total number of data points.

kernels (Härdle, 1989), and parametric estimates using connectionist networks (Bishop, 1994; Srivastava & Weigend, 1994).

We are interested in identifying nonparametric methods that attribute high mutual information to those input variable subsets that result in high prediction accuracy when used in connectionist learning. There is currently little theory to suggest a specific nonparametric density estimation method for estimating mutual information. We explore two approaches in detail: a continuous approach based on kernel density estimation, and a histogram based approach similar to a method proposed by Fraser (1986). One attractive quality of these two particular density estimation methods is an insensitivity to the choice of kernel or bin width, which is often difficult to determine ahead of time without detailed knowledge of the problem.

3.1 KERNEL DENSITY ESTIMATION

One nonparametric method we explored was using continuous kernels to estimate the densities needed for computing mutual information. The estimates are formed by placing a continuous "bump", or kernel function, over each data point sampled from the distribution (Figure 1a). We use a kernel function known as a multi-dimensional Epanechnikov product kernel (Epanechnikov, 1969). If each data point $\mathbf{x}$ has d elements, then

$$K(\mathbf{x}) = \begin{cases} \prod_{i=1}^{d} \frac{3}{4}(1 - x_i^2) & \text{for } ||\mathbf{x}||_\infty < 1 \\ 0 & \text{otherwise} \end{cases}$$

One reason for using this kernel instead of other kernels, such as a Gaussian, is computational speed: only nearby data points can contribute to a density estimate, as opposed to having all data points involved for every estimated probability density. We use an "all nearest neighbors" algorithm proposed by Gabow, Bentley, & Tarjan (1984) to find the nearest points for estimating a probability density, enabling us to compute a mutual information estimate in total time $O(Nb \log b)$.[4] Here, N is the number of data points, and b is the number of bits used to represent a data point.

The probability density for data point k, $P(\mathbf{x}_i(k))$, is estimated as

$$P(\mathbf{x}_i(k)) = \frac{1}{N} \frac{1}{h^d} \sum_{j=1}^{N} K\left(\frac{1}{h}(\mathbf{x}_i(k) - \mathbf{x}_i(j))\right),$$

[4] C Code for this algorithm is available through `http://www.cs.colorado.edu/Time-Series/Code`

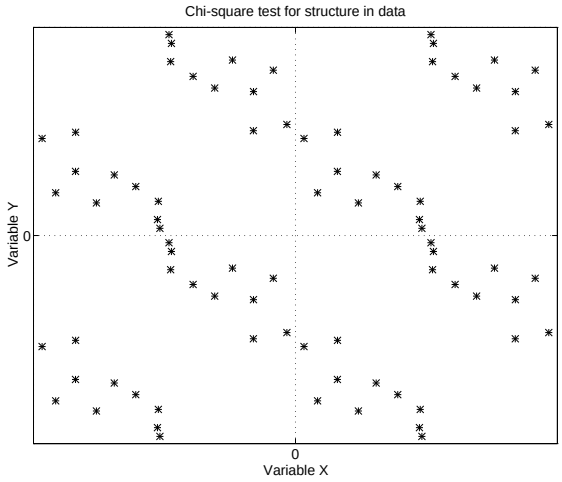

Figure 2: Fraser's χ^2 test for bin splitting. The proposed split along the dotted lines would not be performed because the number of points in each resulting bin is equal. If the split were to be performed, however, a χ^2 test on splitting any of the four resulting bins would indicate that further splitting is needed.

where N is the size of the training set, and h is the "window width", or spread, chosen for the kernel functions. This formula is used to estimate the distributions $P_{\mathbf{X}_i}$, P_Y, and $P_{\mathbf{X}_i,Y}$. The choice of kernel width h cannot be determined *a priori*; its value depends on how "smooth" the true distribution is assumed to be.

3.2 EQUAL MASS BINNING

We explore a nonparametric, histogram based method known as *equal mass binning*, which is a variant of the standard histogram technique. In the standard histogram technique, a density estimate for $P(\mathbf{x})$, where $\mathbf{x}$ is a collection of d random variables, is formed by dividing the range of sampled data for each variable into k equally sized segments. This creates k^d bins of equal volume. A distribution estimate is formed by counting the number of points that fall into a bin and assigning to that bin a probability equal to the number of points it contains divided by the total number of points.

In equal mass binning, rather than forming bins of equal width (or volume), one forms bins of equal mass (equal number of points in each bin). The probability associated with a bin is then given by the relative number of points (roughly equal) divided by the size of the bin (Figure 1b). One version of this idea has been explored by Fraser (1986), where the decision to split a bin is based on a χ^2 test. Splitting is performed whenever the resulting bins would provide significantly different probability estimates compared to not splitting. Unfortunately, this approach can sometimes fail to split even when structure is present in the data (Figure 2). We explore an alternative method where the total number of splits to perform is chosen ahead of time, bin partitions are selected randomly, and mutual information estimates are "bootstrapped" over many runs of the splitting algorithm. The only provision in splitting bins is that bins with many points are chosen before bins with few points; that is, a bin is not considered for splitting until all bins have a comparable number of points in them.

The splitting algorithm is performed for the joint density function $P_{\mathbf{X}_i,Y}$. Once this estimate is computed, we obtain $P_{\mathbf{X}_i}$ and P_Y by summing over bin probabilities, such that marginal probability estimates are obtained.

4 INPUT VARIABLE SEARCH PROCEDURE

In general, it is not possible to assume that the variables in $\{X_1, X_2, \ldots, X_n\}$ are independent. This has an important implication: finding the most relevant subset may require that all $2^n - 1$ subsets be examined. Search procedures discussed in the linear regression literature, such as forward selection, backward elimination, or stepwise selection, where input variables are added or removed one at a time, work well when the input variables are nearly independent (Miller, 1990). We illustrate with two examples why all input subsets

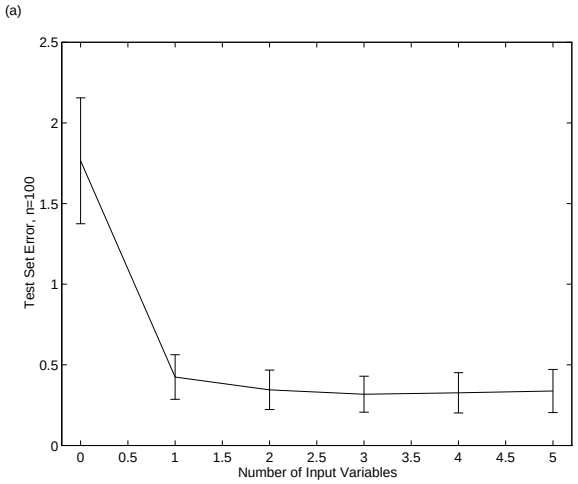

Figure 3: Redundant information in the input values. The graph shows test set errors for a linear network with five possible input units. For each pattern, every input variable carries the same value corrupted by a small amount of Gaussian noise. Although all five input variables appear relevant individually (all have correlations between 0.94 and 0.96 with the output), adding more than one input to the network fails to improve the test set error substantially.

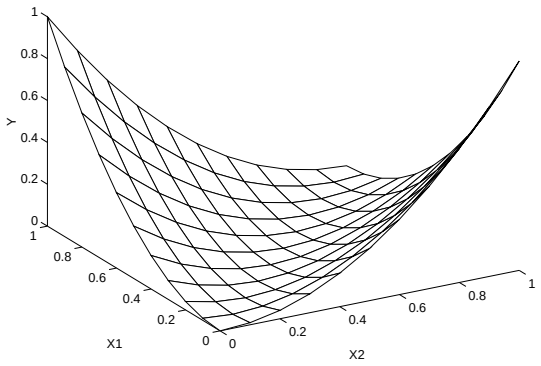

Figure 4: Simple toy problem. A desired input-output mapping $Y = (X_2 - X_1)^2$ is shown. The variables X_1 and X_2 are random variables with values chosen from a uniform distribution on the interval $(0, 1)$. Two other possible input variables also exist; $X_3 = (X_2 - X_1)$, and X_4, which has no bearing on the desired output value.

must potentially be considered when independence among inputs cannot be assumed.

The only alternative to considering all input subsets is to consider some fraction of them and then, from the resulting information, deduce something about the relevance of the other subsets. The first example shows that it is possible to overestimate the relevance of an input variable subset from its components. Figure 3 refers to a situation where five input variables all carry similar information about the output: the input variable values are always similar to each other, and each is highly predictive of the output. Using any standard test of input variable relevance, such as correlation, indicates that each individual input variable is highly relevant; however, using more than one variable for training does not substantially improve prediction performance.

A second example shows that it is possible to underestimate the relevance of an input variable subset from its components. In the XOR problem, neither input variable alone can determine the correct output value; individually, the inputs cannot predict the output value any better than chance. However, it is clear that the combined information of both inputs suffices to correctly determine the value of the output.

The conclusion of our argument is that searching for the best input variable subset is difficult, and the only guaranteed, general method for finding it is to exhaustively test the relevance of each possible subset of input variables. Possible alternatives to exhaustive search are branch and bound techniques (for the linear case, see Furnival & Wilson, 1973), heuristic search methods, and additional problem assumptions, such as independence among input variables (Lewis, 1962).

5 RESULTS

Figure 4 shows the desired input-output mapping of a simple toy problem. The problem is constructed so that input variable X_3 carries information that is redundant with the input variables X_1 and X_2. The two variables X_1 and X_2 are designed to be nearly useless individually, but to be useful together. A fourth input variable X_4 was designed to have no bearing on the desired output value.

We created a training set consisting of 1000 patterns. For each random variable, the sampled values were scaled to have mean zero and variance one. Figure 5 demonstrates how using different density estimation

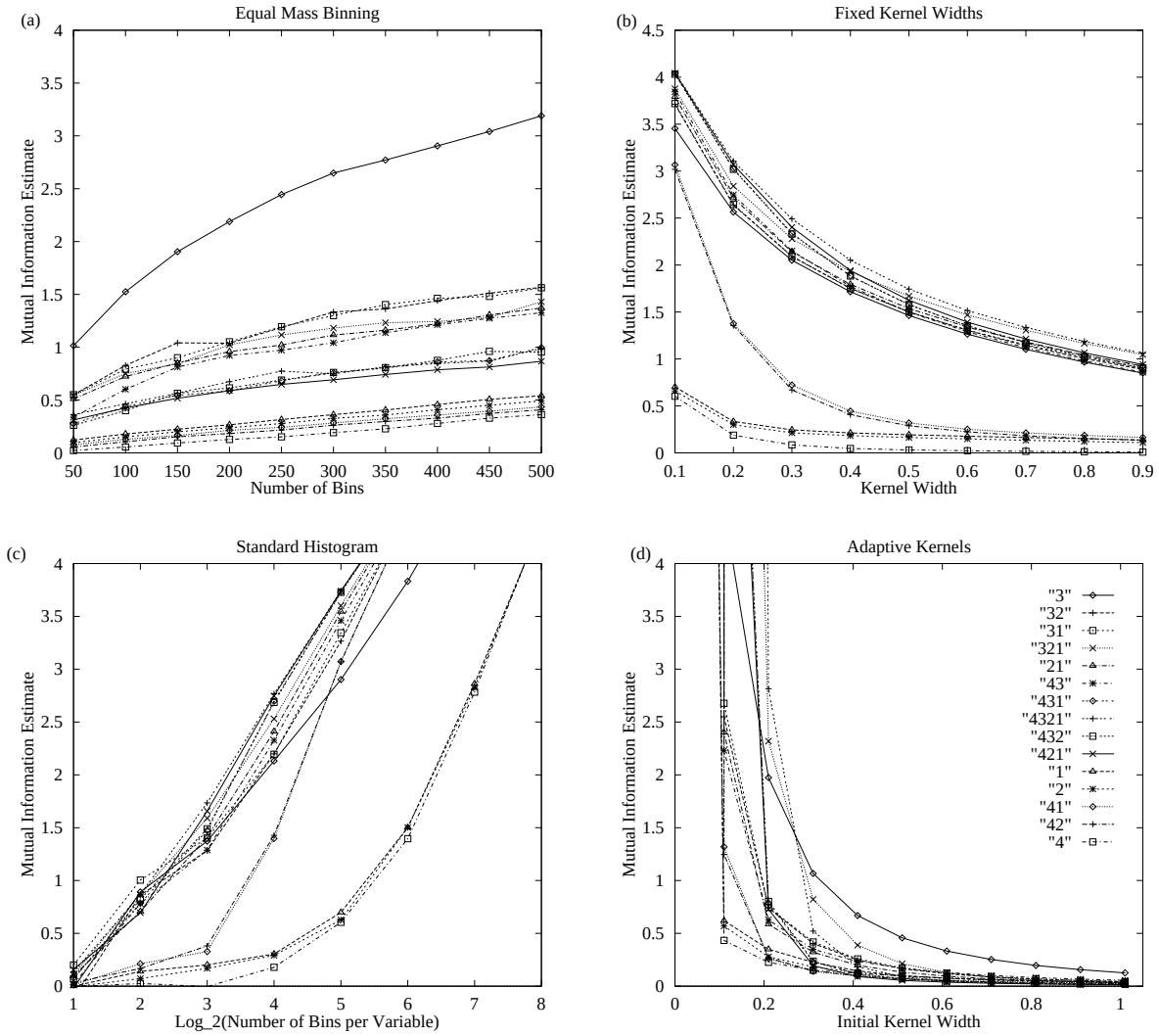

Figure 5: Mutual information estimates using different density estimation methods. The key indicates the subsets; they are ordered according to the importance obtained from method (a).
(a) Equal mass binning. Mutual information estimates were obtained by averaging over 100 runs. The ranking of mutual information estimates remains unaffected by changing the number of bins.
(b) Kernel density estimation using fixed kernel widths. The separation between predictive and nonpredictive subsets is distinct for most kernel width choices. Small kernel widths may tend to confuse information with randomness; subsets $\{X_1, X_4\}$ and $\{X_2, X_4\}$ have the greatest data variance over all subsets with two input variables, and they receive high mutual information scores for small kernel widths, even though they are not predictive of the output variable.
(c) Fixed width histograms. The number of bins is plotted on a logarithmic scale. As the number of bins increase, the subset $\{X_3\}$ receives a lower score relative to other subsets; there is a bias toward large input variable subsets, regardless of how much information about the output they contain.
(d) Adaptive kernel widths. An initial kernel width (plotted on the x-axis) is chosen for all data points, but then widths are adapted: in areas where data points are highly clustered, widths are reduced to capture local structure in the data. In areas where there are few points, kernel widths are increased to prevent the invention of structure in the data. Mutual information estimates based on adaptive kernels appear to depend strongly on the choice of initial kernel width.

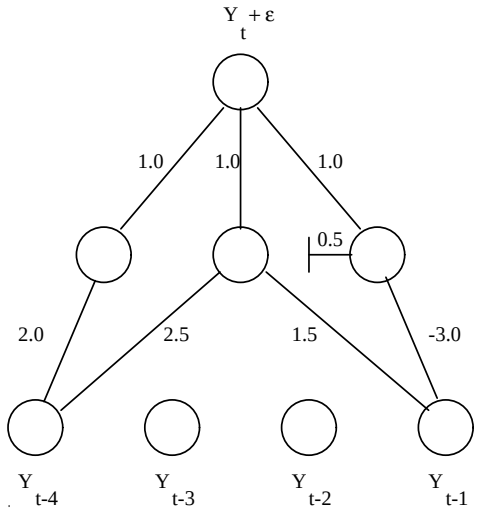

Figure 6: Network used to generate a nonlinear time series. The hidden units use the *tanh* activation function, and the output is linear. The noise added to the output was normal with mean zero and variance 0.1. The input values used to begin generation of the series were all equal to 0.5.

techniques affects mutual information estimates. As Figure 5a indicates, when equal mass bins or fixed width kernels are used, subsets with the ability to predict the output are given the highest mutual information ranking across many width settings.

It is also apparent that in the equal mass binning case, there is a bias toward small input variable subsets: the highest score was given to $\{X_3\}$, which is the smallest input variable subset that can be used to predict the output accurately. In the fixed width kernel method, the situation is reversed: for subsets containing equal information about the output variable, such as $\{X_1, X_2, X_3, X_4\}$ and $\{X_3\}$, a higher score is given to the first subset. This difference emphasizes an important issue in the task of input subset selection: how to compare input variable subsets of different sizes. We are currently working on some modifications to our techniques for addressing this issue.

5.1 COMPARING MUTUAL INFORMATION TO TEST SET ERROR

We also compare our input variable relevance measure with the test set error achieved using the different input variable subsets on a nonlinear time series. A series with twenty thousand points was generated using the network depicted in Figure 6. The last one thousand points were withheld for testing, and the remaining points were randomly split into two equally sized training and cross-validation sets.

There are four inputs to the network, resulting in 15 input variable subsets. For each subset, we trained a network ten times using different initial weights and averaged the test set errors. We also ran the equal mass binning with 200 bins and fixed kernel width algorithm with width 0.1 on the combined training and cross validation data. Different density estimation parameters were not explored for this problem.

Figure 5.1a shows the rank ordering obtained for the fixed width kernel method against the test set error. The correlation in the rank ordering is 0.793; an identical ranking between the two methods would have a correlation of unity. Note that the subset consisting of all input variables gets high rankings from both methods, whereas the true model consisting of time lags y_{t-4} and y_{t-1} gets lower rankings. For the equal mass binning algorithm, the correlation was 0.725 (Figure 5.1b), but the subset of input variables that were actually used to produce the output in the series is ranked highest by the method. This highlights the difference between the two methods on comparisons of different sized input subsets: the equal mass binning algorithm biases toward smaller subsets, while the fixed kernel width algorithm biases toward larger subsets.

One possible reason for the slightly lower overall correlation for the equal mass binning algorithm is illustrated in Figure 8: mutual information estimates involving many variables tend to have a larger variance than those involving just a few variables. One solution to this problem is to run the splitting algorithm until differences in mutual information estimates become significant; this has been termed *racing* (Maron & Moore, 1994).

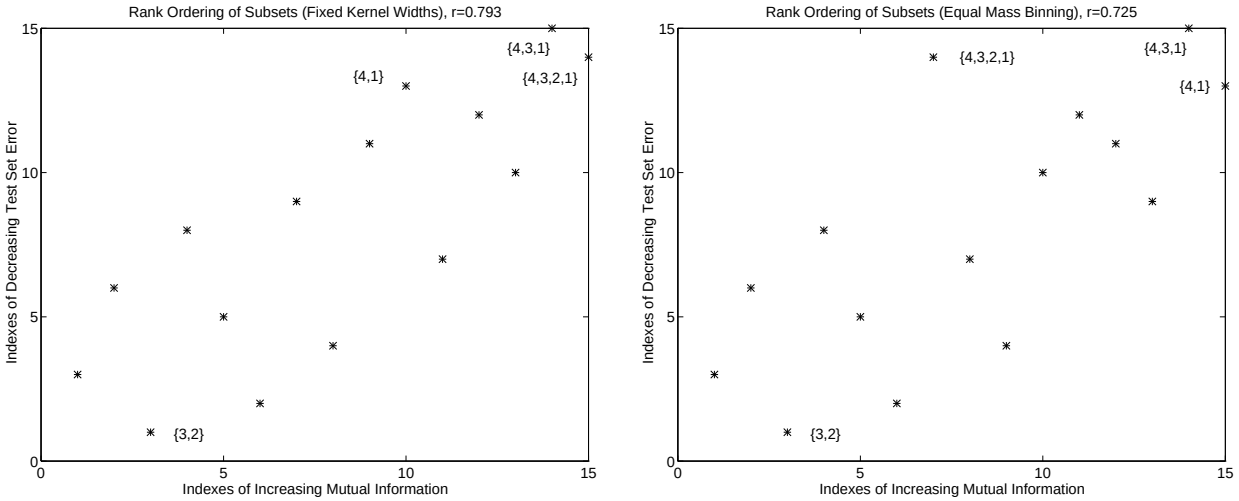

Figure 7: Rank orderings against test set error. (a) Fixed width kernel density estimation. Selected points associated with certain subsets are labeled with their corresponding time lags; that is, the label $\{4, 1\}$ indicates the input variable subset formed from the time lags y_{t-4} and y_{t-1}. (b) Equal mass binning.

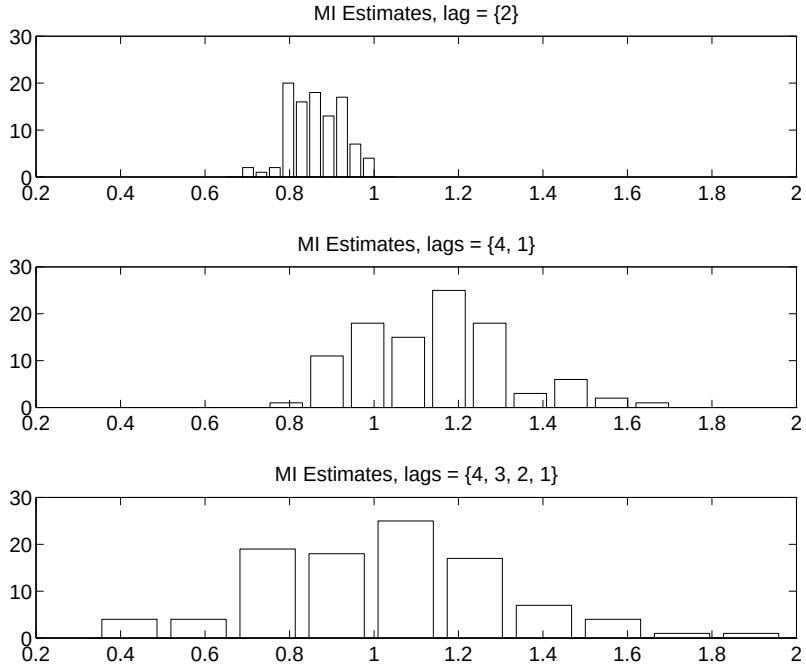

Figure 8: Histograms of mutual information estimates for different input variable subsets. The top histogram, involving just the one input variable corresponding to y_{t-2}, has the least variance, while the bottom histogram, involving all four input variables, has the largest variance.

6 FUTURE RESEARCH DIRECTIONS

One goal in our current research is to combine the best qualities of the two density estimation methods: the smoothness of kernel density estimation with the insensitivity to width choice of equal mass binning. Another goal is to design a principled method for comparing input variable subsets of different sizes using a bootstrap technique.

At present, we are applying this technique to real-world data, including financial time series. Our objective is to understand the class of problems for which it works well; it appears that the technique is particularly well suited for relatively large data sets with nonlinear structure.

Finally, we will relate the nonparametric approach presented in this paper to connectionist approaches for improving generalization performance, such as pruning, weight-elimination, Bayesian methods, and others. Knowing the "best" subset of inputs is also valuable for understanding the underlying system.

Acknowledgements

The authors acknowledge support from the National Science Foundation under Grant No. RIA ECS-9309786. We thank Hal Gabow for pointing out the all nearest neighbors algorithm, Thomas Koetter for discussions on kernel density estimation, as well as Robert Dodier and the other members of the Boulder Connectionist Research group for helpful discussions. We are also grateful for the generous hospitality of the Institut für Wirschaftsinformatik, Humboldt Universität zu Berlin, over the summer of 1994.

References

R. Battiti. (1994) Using mutual information for selecting features in supervised neural net learning. *IEEE Transactions on Neural Networks* **5**(4):537-550.

E. B. Baum & D. Haussler. (1989) What size net gives valid generalization? *Neural Computation* **1**(1):151-160.

C. M. Bishop. (1994) Mixture Density Networks. Unpublished report, Aston University.

W. Buntine & A. S. Weigend. (1991) Bayesian back-propagation. *Complex Systems* **5**:603-643.

T. M. Cover & J. A. Thomas. (1991) *Elements of Information Theory.* New York: John Wiley & Sons.

A. M. Fraser & H. L. Swinney. (1986) Independent coordinates for strange attractors from mutual information. *Physical Review A* **33**(2):1134-1140.

G. M. Furnival & R. W. Wilson. (1974) Regressions by leaps and bounds. *Technometrics*, **16**(4):499-511.

H. N. Gabow, J. L. Bentley, & R. E. Tarjan. (1984) Scaling and Related Techniques for Geometry Problems. In *STOC '84: Proceedings of the Sixteenth Annual ACM Symposium on the Theory of Computing*, 135-140.

S. Geman, E. Bienenstock & R. Doursat. (1992) Neural networks and the bias/variance dilemma. *Neural Computation* **4**(1):1-58.

N. A. Gershenfeld & A. S. Weigend. (1994) The future of time series: learning and understanding. In A. S. Weigend & N. A. Gershenfeld (eds.), *Time Series Prediction: Forecasting the Future and Understanding the Past.* Reading, MA: Addison-Wesley, 1-70.

W. Härdle. (1989) *Applied Nonparametric Regression.* Cambridge, MA: Cambridge University Press.

P. M. Lewis. (1962) The characteristic selection problem in recognition systems. *IRE Transactions on information theory*, 171-178.

D. J. C. MacKay. (1992) A practical Bayesian framework for backpropagation networks. *Neural Computation* **4**(3):448-472.

A. J. Miller. (1990) *Subset Selection in Regression.* London: Chapman and Hall.

O. Maron & A. Moore. (1994) Hoeffding races: accelerating model selection search for classification and function approximation. In *Advances in Neural Information Processing Systems 6.* Morgan Kaufmann, 59-66.

D. W. Scott. (1992) *Multivariate Density Estimation: Theory, Practice, and Visualization.* John Wiley and Sons.

A. N. Srivastava & A. S. Weigend. (1994) Computing the probability density in connectionist regression. In *Proceedings of the 1994 International Symposium on Artificial Neural Networks (ISANN'94),* Tainan, Taiwan.

A. S. Weigend, B. A. Huberman, & D. E. Rumelhart. (1990) Predicting the future: a connectionist approach. *International Journal of Neural Systems*, **1**(3):193-209.