**In: Advances in Intelligent Data Analysis (AIDA),
Computational Intelligence Methods and Applications (CIMA),
International Computer Science Conventions
Rochester New York, 1999**

Feature Selection Based on Joint Mutual Information

Howard Hua Yang and John Moody

Department of Electrical and Computer Engineering
Department of Computer Science and Computer Engineering
Oregon Graduate Institute
20000 NW, Walker Rd., Beaverton, OR97006, USA
e-mail: {hyang,moody}@cse.ogi.edu

Abstract

A feature/input selection method is proposed based on *joint mutual information*. The new method is better than the existing methods based on mutual information in eliminating redundancy in the inputs. It is applied in a real world application to find 2-D viewing coordinates for data visualization and to select inputs for a neural network classifier. The result shows that the new method can find many good 2-D projections which cannot be found by the existing methods.
Keywords: feature selection, joint mutual information, visualization, classification.

1 INTRODUCTION

The goal of statistical modeling is to find a functional relationship between a target variable and a set of feature/input variables or explanatory variables. The statistical modeling is a process of construction and verification of hypotheses. It consists of three basic steps: model specification, model estimation and model selection. These three steps form a common framework for regression, classification and prediction. A family of models are identified in the first step based on the prior knowledge about the underlying problem and the empirical distribution of the data gathered before the modeling. Principles such as maximum likelihood and Bayes rule are applied in the second step to fit the models to the data. Finally, the best model is selected in the third step based on a generalization error on a test data set, cross-validation or a criterion like BIC.

Input variable selection is the most important part of the model selection process, because it interprets the the data modeling problem by specifying those explanatory variables most relevant to the target variables. However, exhaustive search for a set of optimal input variables is exponentially complex. Some heuristic search strategies are needed to select a set of suboptimal input variables. Three search strategies frequently used in selecting regressors for linear models are Forward Selection, Backward Elimination and Stepwise Regression [4]. The same strategies can be applied to select inputs for nonlinear models. To guide the search, we need a *saliency criterion* to rank the input variables according to their relevance to the target variables. We also need a *selection criterion* to evaluate the relevance for a set of selected input variables. The saliency and selection criteria are often different.

Selecting input variables after model specification is a model-dependent approach[7], the learning of a regression or classification model and the variable selection are performed iteratively, leading to the improvement of the model. The saliency criterion is the sensitivity of each input variable. The selection criteria in this approach can include the generalization error, Akaike's Information Criterion, the Bayesian Information Criterion and the related Minimum Description Length. The generalization error can be approximated by the prediction error on a sufficiently large test sample. When there is not enough data for a separate test sample, one can use resampling techniques or algebraic estimates to approximate the generalization error. One resampling technique is a non-linear refinement of n-fold cross-validation [7].

Other model-dependent approaches are based on hierarchical Bayes formulation using a hyperparameter for each input variable. The relevance of the input variables is indicated by the posterior of the hyperparameters [5, 6].

When a target variable is chosen, a set of input variables can be selected as explanatory variables by the prior knowledge. However, many irrelevant input variables cannot be ruled out by the prior knowledge. Too many input variables irrelevant to the target variable will not only severely complicate the model estimation and the model selection process but also damage the performance of the final model. To reduce the computational burden in the estimation

and selection processes, we need model-independent approaches to select input variables before model specification. One such approach is δ-Test [8]. Other approaches are based on the *mutual information* (MI) [1, 2, 3]. MI is very effective in evaluating the relevance of each input variable, but it fails to eliminate redundant variables.

In this paper, we focus on the model-independent approach for input variable selection based on joint mutual information (JMI). The increment from MI to joint MI is the conditional mutual information. Although the conditional mutual information was used in [3] to show the monotonic property of the MI, it was not used for input selection. We shall discuss how to use the conditional mutual information to compute JMI of two input variables and one target variable.

2 MI and conditional MI

Let Y be a target variable and X_i's are inputs. The relevance of a single input is measured by the MI

$$I(X_i;Y) = K[p(x_i,y)\|p(x_i)p(y)]$$

where $K[p\|q]$ is the Kullback-Leibler divergence of two probability functions p and q defined by $K[p(x)\|q(x)] = \sum_x p(x)\log\frac{p(x)}{q(x)}$. For continuous random variables, the summation becomes an integral.

The relevance of a set of inputs is defined by the *joint mutual information*

$$I(X_i,\cdots,X_k;Y) = K[p(i,\cdots,k,y)\|p(i,\cdots,k)p(y)]$$

where

$$\begin{aligned} p(i,k) &= p(i,\cdots,k)\\ p(i,k,y) &= p(x_i,\cdots,x_k,y). \end{aligned}$$

Given two previously selected inputs x_j and x_k, the *conditional MI* is defined by

$$I(X_i;Y|X_j,X_k) = \sum_{x_j,x_k} p(j,k)K[p(i,y|j,k)\|p(i|j,k)p(y|j,k)]$$

where

$$\begin{aligned} p(i,y|j,k) &= p(x_i,y|x_j,x_k),\\ p(i|j,k) &= p(x_i|x_j,x_k),\\ p(y|j,k) &= p(y|x_j,x_k). \end{aligned}$$

Similarly, we can define the conditional mutual information $I(X_i;Y|X_j,\cdots,X_k)$, which is a mutual information conditioned on more than two variables. The conditional MI is always non-negative since it is a weighted average of the Kullback-Leibler divergence.

In the model-independent approach, we can still use the previously mentioned heuristic search strategies to select inputs. In contrast to the model-dependent approach, we select inputs before model specification. The MI $I(X_i;Y)$ is the saliency criterion to rank the relevance of the inputs. When the input X_k is chosen, the conditional mutual information $I(X_i;Y|X_k)$ is a conditional saliency used to rank the rest of the inputs. The joint mutual information $I(X_i,\cdots,X_k;Y)$ is a selection criterion.

It is easy to show that

$$\begin{aligned} &I(X_1,\cdots,X_{n-1},X_n;Y) - I(X_1,\cdots,X_{n-1};Y)\\ &= I(X_n;Y|X_1,\cdots,X_{n-1}) \geq 0. \end{aligned}$$

Therefore,

$$I(X_1,\cdots,X_{n-1},X_n;Y) \geq I(X_1,\cdots,X_{n-1};Y),$$

which means that adding the variable X_n will always increase the mutual information. The information gained by adding a variable is measured by the conditional MI.

When X_n and Y are conditionally independent given $X_1,\cdots,X_{n-1}$, the conditional MI is zero

$$I(X_n;Y|X_1,\cdots,X_{n-1}) = 0, \tag{1}$$

so X_n provides no extra information about Y when $X_1, \cdots, X_{n-1}$ are known. In particular, when X_n is a function of $X_1, \cdots, X_{n-1}$, the equality (1) holds. This is the reason why the joint MI can be used to eliminate redundant inputs.

In practice, one can only estimate the conditional MI from data. We need a significance test to check whether the true conditional MI is zero.

The conditional MI is useful when the input variables cannot be distinguished by the mutual information $I(X_i; Y)$. For example, assume $I(X_1; Y) = I(X_2; Y) = I(X_3; Y)$, and the problem is to select $(x_1, x_2), (x_1, x_3)$ or (x_2, x_3).

Since

$$\begin{aligned} & I(X_1, X_2; Y) - I(X_1, X_3; Y) \\ & = I(X_2; Y|X_1) - I(X_3; Y|X_1), \end{aligned}$$

we should choose (x_1, x_2) if

$$I(X_2; Y|X_1) > I(X_3; Y|X_1).$$

Otherwise, we should choose (x_1, x_3). All possible comparisons are represented by a binary tree in Figure 1.

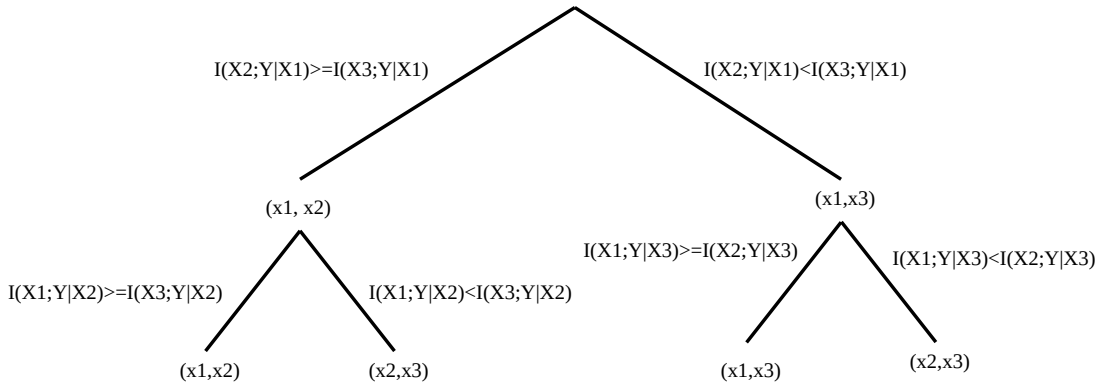


Figure 1: Input selection based on the conditional MI.

Selecting two or three input variables is very important for data visualization. The most natural way to visualize high-dimensional input patterns is to display them using two of the existing coordinates, where each coordinate corresponds to one input variable. Those inputs which are most relevant to the target variable corresponds the best coordinates for data visualization. Let

$$(i^*, j^*) = \arg \max_{(i,j)} I(X_i, X_j; Y).$$

Then, the coordinates (x_{i^*}, x_{j^*}) should be used for visualizing the input patterns since the corresponding inputs achieve the maximum joint MI.

To find the maximum $I(X_{i^*}, X_{j^*}|Y)$, we need to evaluate every joint MI, $I(X_i, X_j; Y)$ for $i < j$. The number of evaluations is $O(n^2)$.

Noticing that

$$I(X_i, X_j; Y) = I(X_i; Y) + I(X_j; Y|X_i),$$

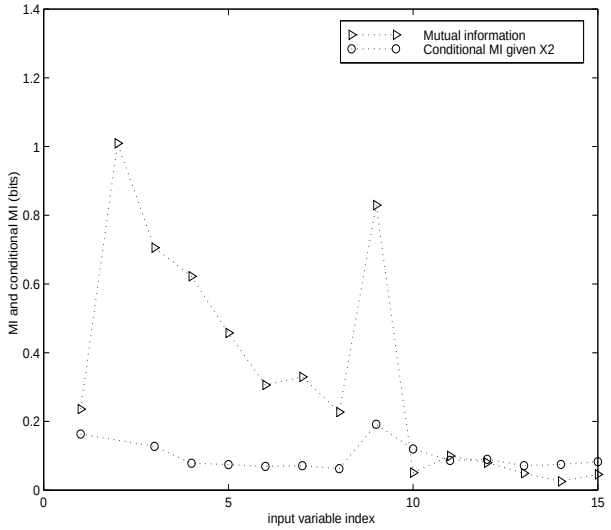
we can first maximize the mutual information $I(X_i; Y)$, then maximize the conditional MI. This algorithm is suboptimal, but only requires the evaluation of the joint mutual information $n - 1$ times. Sometimes, this is equivalent to exhaustive search. One such example is given in next section.

Another method to visualize high-dimensional patterns is to use a dimension reduction method such as PCA to find the new coordinates and use the two principal components to display the data. We shall compare these two visualization methods later.

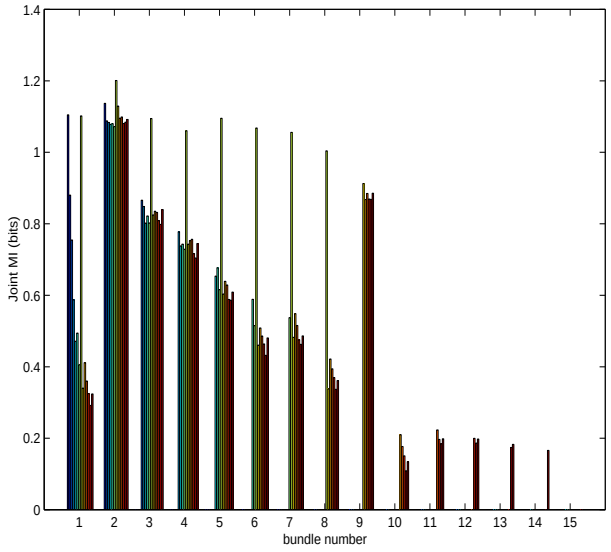
To estimate the high dimensional MI $I(X_1, \cdots, X_k; Y)$, we need to estimate the joint probability $p(x_1, \cdots, x_k, y)$. This suffers from the curse of dimensionality when k is large. Assume that all random variables involved are discrete, and each of them may take one of M different values. Then, the random vector $(X_1, \cdots, X_k, Y)$ has M^{k+1} possible states. In order to estimate $p(x_1, \cdots, x_k, y)$ by a histogram, the sample size should be in the order of $O(M^{k+1})$. Sometimes, we may not be able to estimate high dimensional MI due to the sample shortage. Further work is needed

to estimate high dimensional joint MI based on parametric and non-parametric density estimations, when the sample size is not large enough. Such approaches could involve techniques related to ICA (see [9] for example).

In some real world problems such as mining large data bases and radar pulse classification, the sample size is large, but the parametric densities for the underlying distributions are unknown. It is better to use non-parametric methods such as histograms to estimate the joint probability and the joint MI.



(a)



(b)

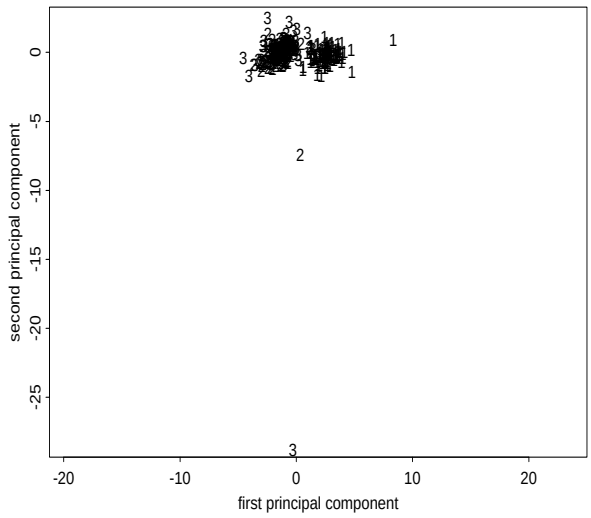
Figure 2: (a) MI vs conditional MI for the radar pulse data; maximizing the MI then the conditional MI with $O(n)$ evaluations gives $I(X_{i_1}, X_{j_1}; Y) = 1.201$ bits. (b) the joint MI for the radar pulse data; maximizing the joint MI gives $I(X_{i^*}, X_{j^*}; Y) = 1.201$ bits with $O(n^2)$ evaluations of the conditional MI. $(i_1, j_1) = (i^*, j^*)$ in this case.

3 A Real World Application

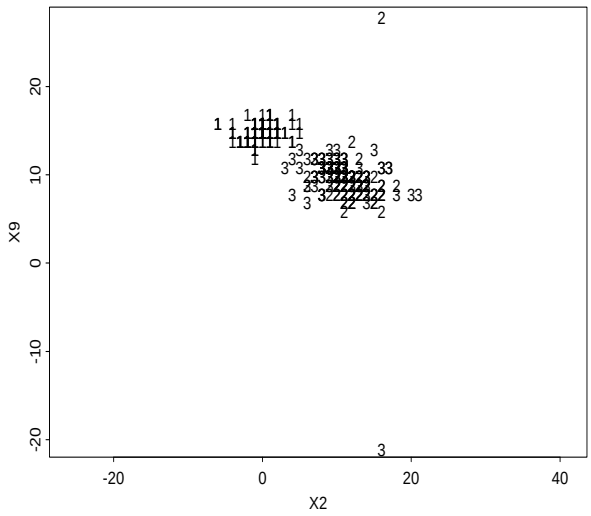
3.1 The joint mutual information of radar pulse patterns

Our goal is to design a classifier for radar pulse recognition. Each radar pulse pattern is a 15-dimensional vector. We first compute the joint MIs, then use them to select inputs for the visualization and classification of radar pulse

patterns.



(a)



(b)

Figure 3: (a) Data visualization by two principal components; the spatial relation between patterns is not clear. (b) Use the optimal view coordinates (x_{i^*}, x_{j^*}) to project the radar pulse data; the patterns are well spread to give a better view on the spatial relation between patterns and the boundary between classes.

A set of radar pulse patterns is denoted by

$$D = \{(\boldsymbol{x}^i, y^i) : i = 1, \cdots, N\}$$

which consists of patterns in three different classes. Here, each $\boldsymbol{x}^i \in R^{15}$ and each $y^i \in \{1, 2, 3\}$.

Let $i_1 = \arg\max_i I(X_i; Y)$ and
$j_1 = \arg\max_{j \neq i_1} I(X_j; Y|X_{i_1})$.

From Figure 2(a), we obtain $(i_1, j_1) = (2, 9)$ and
$I(X_{i_1}, X_{j_1}; Y) = I(X_{i_1}; Y) + I(X_{j_1}; Y|X_{i_1}) = 1.201$ bits.

If the number of total inputs is n, then the number of evaluations for computing the mutual information $I(X_i; Y)$ and the conditional mutual information $I(X_j; Y|X_{i_1})$ is $O(n)$.

To find the maximum $I(X_{i^*}, X_{j^*}; Y)$, we evaluate every $I(X_i, X_j; Y)$ for $i < j$. These MIs are shown by the bars in Figure 2:Right, where the i-th bundle displays the MIs $I(X_i, X_j; Y)$ for $j = i+1, \cdots, 15$.

In order to compute the joint MIs, the MI and the conditional MI are evaluated $O(n)$ and $O(n^2)$ times respectively. The maximum joint MI is

$$I(X_{i^*}, X_{j^*}; Y) = 1.201 \text{ bits}.$$

Generally, we only know

$$I(X_{i_1}, X_{j_1}; Y) \leq I(X_{i^*}, X_{j^*}; Y).$$

But in this particular application, the equality holds. This suggests that sometimes we can use an efficient algorithm with only linear complexity to find the optimal view coordinates (x_{i^*}, x_{j^*}). The joint MI also gives other good sets of view coordinates with high joint MI values.

Each bar in Figure 2(b) is associated with a pair of inputs. Those pairs with high joint MI give good view coordinates for data visualization. Figure 3 shows that the data visualization by the two most relevant inputs is better than that by the two principal components. The reason is that the principal components are determined by the inputs only while the most relevant inputs are determined by both inputs and the target variable.

3.2 Radar pulse classification

Now we train a two layer feed-forward network to classify the radar pulse patterns. Figure 3 shows that it is very difficult to separate the patterns by using just two inputs. We shall use all inputs or four selected inputs. The data set D is divided into a training set D_1 and a test set D_2 consisting of 20 percent patterns in D. The network trained on the data set D_1 using all input variables is denoted by

$$Y = f(X_1, \cdots, X_n; \boldsymbol{W}_1, \boldsymbol{W}_2, \theta)$$

where $\boldsymbol{W}_1$ and $\boldsymbol{W}_2$ are weight matrices and θ is a vector of thresholds for the hidden layer.

From the data set D, we estimate the mutual information $I(X_i; Y)$ and select $i_1 = \arg\max_i I(X_i; Y)$. Given X_{i_1}, we estimate the conditional mutual information $I(X_j; Y | X_{i_1})$ for $j \neq i_1$.

Choose three inputs X_{i_2}, X_{i_3} and X_{i_4} with the largest conditional MI. We found a quartet $(i_1, i_2, i_3, i_4) = (1, 2, 3, 9)$. The two-layer feed-forward network trained on D_1 with four selected inputs is denoted by

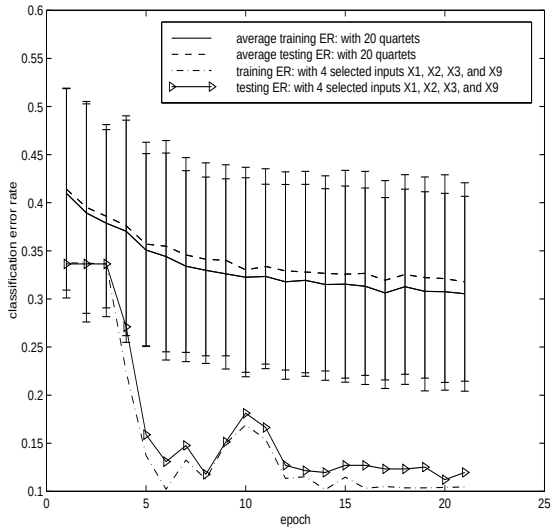
$$Y = g(X_1, X_2, X_3, X_9; \boldsymbol{W}'_1, \boldsymbol{W}'_2, \theta').$$

There are 1365 choices to select 4 input variables out of 15. To set a reference performance for network with four inputs for comparison. Choose 20 quartets from the set

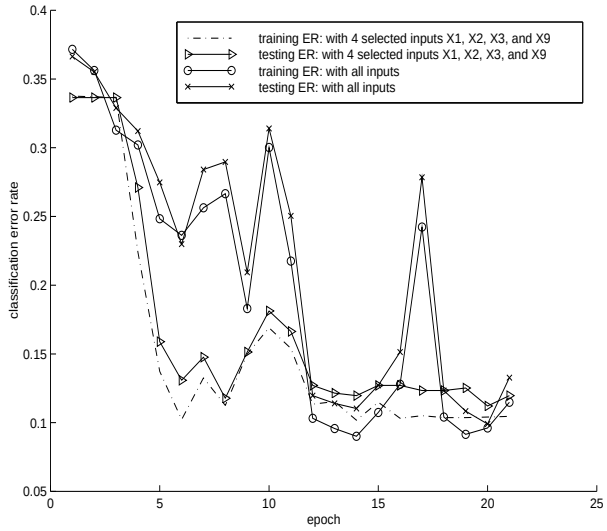
$$Q = \{(j_1, j_2, j_3, j_4) : \ 1 \leq j_1 < j_2 < j_3 < j_4 \leq 15\}.$$

For each quartet (j_1, j_2, j_3, j_4), a two-layer feed-forward network is trained using inputs $(X_{j_1}, X_{j_2}, X_{j_3}, X_{j_4})$. These networks are denoted by

$$Y = h_i(X_{j_1}, X_{j_2}, X_{j_3}, X_{j_4}; \boldsymbol{W}''_1, \boldsymbol{W}''_2, \theta''), \quad i = 1, 2, \cdots, 20.$$



(a)



(b)

Figure 4: (a) The error rates of the network with four inputs (X_1, X_2, X_3, X_9) selected by the joint MI are well below the average error rates (with error bars attached) of the 20 networks with different input quartets randomly selected; this shows that the input quartet (X_1, X_2, X_3, X_9) is rare but informative. (b) The network with the inputs (X_1, X_2, X_3, X_9) converges faster than the network with all inputs. The former uses 65% fewer parameters (weights and thresholds) and 73% fewer inputs than the latter. The classifier with the four best inputs is less expensive to construct and use, in terms of data acquisition costs, training time, and computing costs for real-time application.

The mean and the variance of the error rates of the 20 networks are then computed.

All networks have seven hidden units. The training and testing error rates of the networks at each epoch are shown in Figure 4.

It is shown by Figure 4 that the network with four inputs selected by the joint MI performs better than the networks with randomly selected input quartets and converges faster than the network with all inputs. The network with fewer inputs is not only faster in computing but also less expensive in data collection.

4 CONCLUSIONS

Input selection methods based on mutual information (MI) have been useful in many applications, but they have two disadvantages. First, they cannot distinguish inputs when all of them have the same MI. Second, they cannot eliminate the redundancy in the inputs when one input is a function of other inputs. In contrast, our new input selection method based on the *joint* MI offers significant advantages in these two aspects. The new method finds the optimal viewing coordinates with the highest joint MI. We have successfully applied this method to visualize radar patterns and to select inputs for a neural network classifier to recognize radar pulses.

Acknowledgement: We thank Geoffrey Barrows and John Sciortino of the Naval Research Lab for helpful discussions and for providing the dataset. This research was supported by grant ONR N00014-96-1-0476.

References

[1] G. Barrows and J. Sciortino. A mutual information measure for feature selection with application to pulse classification. In *IEEE Intern. Symposium on Time-Frequency and Time-Scale Analysis* , pages 249–253, 1996.

[2] R. Battiti. Using mutual information for selecting features in supervised neural net learning. *IEEE Trans. on Neural Networks*, 5(4):537–550, July 1994.

[3] B. Bonnlander. Nonparametric selection of input variables for connectionist learning. Technical report, PhD Thesis. University of Colorado, 1996.

[4] N. Draper and Smith H. *Applied Regression Analysis, Second Edition.* John Wiley & Sons, Inc., 1981.

[5] E. I. George and R. E. McCulloch. Approaches for bayesian variable selection. *Statistica Sinica*, 7, 1997.

[6] D. J. C. Mackay. Bayesian non-linear modelling for prediction competition. In *Ashrae Transactions, V.100, Pt.2*, pages 1053–1062. Atlanta Georgia, 1994.

[7] J. Moody. Prediction risk and architecture selection for neural network. In V. Cherkassky, J.H. Friedman, and H. Wechsler, editors, *From Statistics to Neural Networks: Theory and Pattern Recognition Applications.* NATO ASI Series F, Springer-Verlag, 1994.

[8] H. Pi and C. Peterson. Finding the embedding dimension and variable dependencies in time series. *Neural Computation*, 6:509–520, 1994.

[9] H. H. Yang and S. Amari. Adaptive on-line learning algorithms for blind separation: Maximum entropy and minimum mutual information. *Neural Computation*, 9(7):1457–1482, 1997.