Behaviormetrika, Vol. 26, No. 1, 145-166
1999

FEATURE SELECTION WITH NEURAL NETWORKS

Philippe Leray* and Patrick Gallinari*

The observed features of a given phenomenon are not all equally informative : some may be noisy, others correlated or irrelevant. The purpose of feature selection is to select a set of features pertinent to a given task. This is a complex process, but it is an important issue in many fields. In neural networks, feature selection has been studied for the last ten years, using conventional and original methods. This paper is a review of neural network approaches to feature selection. We first briefly introduce baseline statistical methods used in regression and classification. We then describe families of methods which have been developed specifically for neural networks. Representative methods are then compared on different test problems.

1. Introduction

The primary source of information in learning systems is data. For numerical systems like Neural Networks (NN), data is usually represented as vectors in a subspace of R^k whose components—or features—may correspond to measurements performed on a physical system, for example, or to information collected from observation of a phenomenon. Usually, not all the features observed are equally informative : some may be noisy, meaningless, correlated or irrelevant to the task. The purpose of the feature selection process is to select a subset of features relevant to a given problem. This is often an important phase of work because, among other things, it can reduce the amount of data to be collected or processed, make training easier, improve estimates by using relevant features on small data sets, allow the use of more sophisticated processing methods on dimensional spaces that are smaller than the original measurement space, or improve performance by avoiding the interference of non-relevant information.

Feature selection has been the subject of intense research in statistics and in applied fields such as pattern recognition, process identification, time series modeling, and econometrics. It has recently began to be investigated in the machine learning community, which has developed its own methods. Whatever the domain, feature selection is always a difficult problem. Most of the time the solution is non-monotone, *i.e.* the best subset of p variables does not always contain the best subset of q variables ($q<p$). Also, the best subset of variables depends on the model that will later be used to process the data. Usually, these two steps are treated sequentially. Most variable selection methods rely on heuristics, which perform a limited exploration on the whole set of variable combinations.

Key Words and Phrases : Feature Selection, Subset selection, Variable Sensitivity, Sequential Search.
* LIP6-Pôle IA-Université Paris 6-boite 169, 4, Place Jussieu-75252 Paris cedex 05-France. {Philippe.Leray, Patrick.Gallinari}@lip6.fr

In the field of NNs, feature selection has been approached by conventional and original methods for ten years now. The present paper discusses the problem of feature selection specifically for NNs, and reviews original methods that have been developed in this field. The review is certainly not exhaustive, considering the extensive literature in the field, but the main ideas proposed are described.

Sections 2 and 3 describe the basic ingredients of feature selection methods and the notation. Then, in section 4, we briefly present the statistical methods used in regression and classification. These will be used as baseline techniques. Section 5 describes families of methods that have been developed specifically for neural networks and are easy to use for regression or classification tasks. Representative methods are then compared on different test problems in section 6.

2. Basic ingredients of feature selection methods

A feature selection technique typically requires the following ingredients:

- a feature evaluation criterion to compare variable subsets for selection;
- a search procedure, to explore a (sub)space of possible variable combinations,
- a stop criterion or model selection strategy.

2.1 Feature evaluation

Depending on the task (e.g. prediction or classification) and on the model (linear, logistic, neural networks or other), several statitical and heuristic-based evaluation criteria have been proposed for measuring the importance of a variable subset. For classification, the usual criteria use probabilistic distances or entropy measures, often replaced in practice by simple interclass distance measures. For regression, the usual candidates are prediction error measures. A survey of classical statistical methods may be found in Thompson (1978) for regression and McLachlan (1992) for classification.

Certain methods rely only on the data for computing relevant variables without considering the model used for processing the data after the selection. Some of these (the parametric methods) may rely on assumptions about the data distribution while others do not (non-parametric methods). Still other methods do take the model and data into account simultaneously—this is usually the case for NN variable selection.

2.2 Search

In general, since evaluation criteria are non-monotonic, comparing feature subsets is equivalent to a combinatorial problem (there are 2^k-1 possible subsets for k variables), which rapidly becomes computationally unfeasible, even for moderate input size. *Branch and Bound* exploration (Narendra & Fukunaga, 1977) reduces the search for monotonic criteria; however, the complexity of these proce-

dures is still prohibitive in most cases. Due to these limitations, most algorithms are based on heuristic performance measures for the evaluation and sub-optimal search. Most sub-optimal search methods follow one of the following sequential search techniques (see e.g. Kittler, 1986):

- Start with an empty set of variables and add variables to the variable set already selected (*forward* methods).
- Start with the full set of variables and proceed by elimination of variables from the selected variable set (*backward* methods).
- start with an empty set and alternate between the above forward and backward steps (*stepwise* methods). The *Plus* l—*Take away* r algorithm is a generalization of the basic stepwise method which alternates l forward selections and r backward deletions.

2.3 Subset selection—Stopping criterion

Given a feature subset evaluation criterion and a search procedure, there exist several methods for examining all the subsets provided by the search (e.g. 2^k-1 for an exhaustive search or k for a simple backward search) and selecting the most relevant according to the evaluation criterion.

When the empirical distribution of the evaluation measure or of related statistics is known, tests exist for determining the relevance (or irrelevance) of an input variable. The usual sequential selection procedures use a stop criterion, by which they examine the variables sequentially and stop as soon as a variable is found to be irrelevant according to some statistical test. For ordinary parametric methods, the distribution characteristics (e.g. estimates of the evaluation measure variance) are easily derived (see sections 4.1 and 4.2). For non parametric or flexible methods like NNs, these distributions are more difficult to obtain. Confidence intervals for performing significance testing could be computed by Monte Carlo simulations or bootstrapping, but this is extremely complex and of no practical use except for very special cases (e.g. Baxt & White 1996). Hypothesis testing is thus seldom used with these models. Many authors use heuristic stop criteria instead.

Another methodology of reasonable complexity in most applications is to compute an estimate of the generalization error (or prediction risk) for each of a series of variable subsets provided by the search algorithm. The variables selected are the ones offering the best performance. The generalization error estimate may be computed using a validation set or cross-validation or algebraic methods although the latter are not easy to obtain with nonlinear models. Note that this strategy involves retraining an NN for each subset.

3. Notation

We will use $(\boldsymbol{x}, \boldsymbol{y}) \in \Re^k \times \Re^g$ to denote the realization of a random variable pair

(X, Y) with probability distribution P. x_i will be the i^{th} component of $\boldsymbol{x}$, and $\boldsymbol{x}^l$ the l^{th} pattern in a given data set D of cardinality N. In the following, we will restrict ourselves to one hidden layer NNs, and the number of input and output units will be denoted respectively by k and g. The transfer function of the network will be denoted f. Training will be performed here according to a Mean Squared Error criterion (MSE) although this is not restrictive.

In the following, we will consider selection methods for classification and regression tasks.

4. Model independent feature selection

Here we will introduce a number of methods that perform the selection and classification or regression steps sequentially, i.e. without considering the classification or regression model during the selection process. These methods are not NN-oriented and are used here for the purposes of experimental comparison with NN-specific selection techniques (section 6). The first two are basic statistical techniques aimed at regression and classification, respectively. These are not well suited for NNs, since the hypothesis they rely on does not correspond to situations where NNs might be of use. However, since most NN-specific methods are heuristic, they should be used for a baseline comparison. The third method has been developed more recently. It is a general selection technique including no assumptions concerning the data. It can be used for any system, either for regression or classification. It is based on a probabilistic dependence measure between two sets of variables.

4.1 Feature selection for linear regression

We will consider only linear regression, but the approach described below may be trivially extended for multiple regression. Let $x_1, x_2, \ldots x_k$ and y be real variables which are assumed to be centered. Let us use:

$$f_{(p)}(\boldsymbol{x}) = \sum_{i=1}^{p} b_i x_i \tag{4.1.1}$$

to denote the current approximation of y with p selected variables (the x_i are renumbered so that the p first selected variables correspond to numbers 1 to p). The residuals $e_{(p)} = f_{(p)}(\boldsymbol{x}) - y$ are assumed to be identically and independently distributed.

Let us denote:

$$SST = \sum_{l=1}^{N} (y^l)^2 \qquad SSR_p = \sum_{l=1}^{N} f_{(p)}^2(\boldsymbol{x}^l) \tag{4.1.2}$$

For forward selection, the choice of the p^{th} variable is usually based on R_p^2, the partial correlation coefficient (table 1) between y and regressor $f_{(p)}$, or on an

Table 1
Choice and Stop criteria used with statistical forward and backward methods

	Choice	Stop
Forward	$R_p^2=\frac{SSR_p}{SST}=\frac{\sum_1^N f_{(p)}^2(x^l)}{\sum_1^N (y^l)^2}$	$F_s(p)_{forward}=\frac{SSR_p-SSR_{p-1}}{(SST-SSR_p)/(N-p)}$
Backward	$SSR_{p-1}=\sum_{l=1}^{N} f_{(p-1)}(x^l)^2$	$F_s(p)_{backward}=\frac{SSR_p-SSR_{p-1}}{(SST-SSR_p)/(N-p)}$

adjusted coefficient (the adjusted coefficient $\overline{R}_p^2=\frac{N*R_p^2-p}{N-p}$ is often used instead of R_p^2). This coefficient represents the proportion of the total variance of y due to the regressor $f(p)$. The p^{th} variable to select is the one for which $f(p)$ maximizes this coefficient. The importance of a new variable is usually measured by a Fisher test (Thompson, 1978), which compares the models with $p-1$ and p variables ($F_s(p)_{forward}$ in table 1). Selection is stopped if $Fs(p)_{forward}<F(1,\ N-p,\ \alpha)$, the Fisher statistics with $(1,\ N-p)$ degrees of freedom for a confidence level of α.

Note that F_s could also be used as a selection criterion in place of R_p^2 criterion:

$$Fs(p)_{forward}=\frac{R_p^2-R_{p-1}^2}{R_p^2/(N-p)} \tag{4.1.3}$$

When $p-1$ variables have already been selected, R_{p-1}^2 has a constant value in $[0, 1]$ and maximizing F_s is similar to maximizing R_p^2. Equation (4.1.3) selects variables in the same order as R_p^2 does.

For the backward process, the variable eliminated from the remaining p is the least significant in terms of the Fisher test, i.e. the one with the smallest value of SSR_{p-1} or, equivalently, $Fs(p)_{backward}$ (table 1). Selection is stopped if $Fs(p)_{backward}>F(1,\ N-p,\ \alpha)$.

4.2 Feature selection for classification

For classification purposes, we select the variable subset that offers the best separation of the data. Variable selection ordinarily uses a class separation as selection criterion and an F-test as stopping criterion. As for regression tasks, forward, backward or stepwise methods may be used.

Data separation is usually computed by means of some inter-class distance measure (Kittler, 1986). The most frequent discriminating measure is the Wilks lambda (Wilks, 1963) Λ_{SV_p} defined as:

$$\Lambda_{SV_p}=\frac{|\boldsymbol{W}|}{|\boldsymbol{W}+\boldsymbol{B}|} \tag{4.2.1}$$

where $\boldsymbol{W}$ is the intra-class matrix dispersion corresponding to the selected variable set SV_p, $\boldsymbol{B}$ the corresponding inter-class matrix, and $|\boldsymbol{M}|$ the determinant of matrix $\boldsymbol{M}$.

$$\boldsymbol{W}=\sum_{j=1}^{g}\sum_{x^l\in class\ j}(\boldsymbol{x}^l-\mu_j)^t(\boldsymbol{x}^l-\mu_j) \tag{4.2.2}$$

$$\boldsymbol{B}=\sum_{j=1}^{g} N_j(\mu-\mu_j)^t(\mu-\mu_j) \tag{4.2.3}$$

with g the number of classes, N_j the number of samples in class j, μ_j the mean of class j and μ the global mean.

As the determinant of a covariance matrix is a measure of the volume occupied by the data, $|\boldsymbol{W}|$ measures the mean volume of the different classes and $|\boldsymbol{W}+\boldsymbol{B}|$ the volume of the whole data set. These quantities are computed for the selected variables so that a good discriminating power corresponds to a small value of Λ_{SV_p}: the different classes are represented by compact clusters and are well separated. This criterion is well suited to the case of multiple normal distributions with equal covariance for each class, but it is meaningless for multimode distributions, for example. This is clearly a very restrictive hypothesis.

With this measurement, the statistic F_s defined below has a $F(g-1, N-g-p+1, \alpha)$ distribution (McLachlan, 1992):

$$F_s=\frac{(N-g-p+1)}{(g-1)}\frac{(1-\Lambda_{SV_p})}{\Lambda_{SV_p}} \tag{4.2.4}$$

We can then use the Wilks lambda both for estimating the discriminating power of a variable and for stopping the selection in a forward, backward (Habbema & Hermans, 1977), or stepwise method.

For the comparisons in section 6, we used Stepdisc, a stepwise method based on (4.2.4) with a 95% confidence level.

4.3 Mutual information

When data is considered as a realization of a random process, probabilistic information measures may be used in order to compute the relevance of a set of variables with respect to other variables. Mutual information is such a measure which is defined as:

$$MI(a, b)=\sum_{a,b} P(a, b)\times\log\left(\frac{P(a, b)}{P(a)P(b)}\right) \tag{4.3.1}$$

where a and b are two variables with probability density $P(a)$ and $P(b)$.

Mutual information is independent of any inversible and differentiable transformation of the variables. It measures the "uncertainty reduction" on b when a is known. It is also known as the Kullbak-Leibler distance between the joint distribution $P(a,b)$ and the marginal distribution product $P(a)*P(b)$.

The method described below does not make use of restrictive assumptions on the data and is therefore more general and attractive than the ones described in sections 4.1 and 4.2, especially when these hypotheses do not correspond to the data processing model, which is usually the case for NNs. It may be used either for regression or discrimination. On the other hand, such non-parametric methods are computationally intensive. The main practical difficulty here is the estimation of

the joint density $P(a, b)$ and of the marginal densities $P(a)$ and $P(b)$. Non-parametric density estimation methods are costly in high dimensions and require a large amount of data.

The algorithm presented below uses the Shannon entropy (equation 4.3.2) to compute the mutual information $MI(a, b)=H(a)+H(b)-H(a, b)$. It is possible to use other entropy measures like quadratic or cubic entropies (Kittler, 1986).

$$H(a)=-\int p(a)\ \log(p(a))da \tag{4.3.2}$$

Battiti (1994) proposed using mutual information with a forward selection algorithm called *MIFS* (Mutual Information-based Feature Selection). $P(a, b)$ is estimated by Fraser algorithm (Fraser & Swinney, 1986), which recursively partitions the space using χ^2 tests on the data distribution. This algorithm can only compute the mutual information between two variables. In order to compute the mutual information between x_p and the selected variable set SV_{p-1}($x_p$does not belong to SV_{p-1}) Battiti uses simplifying assumptions. Moreover, the number of variables to select is fixed before the selection. This algorithm uses forward search and variable x_p is the one that maximises the value:

$$MI(SV_{p-1}\cup\{x_p\},\ \mathbf{y}) \tag{4.3.3}$$

where SV_{p-1} is the set of $p-1$ variables already selected.

Bonnlander and Weigend (1994) use Epanechnikov kernels for density estimation (Härdle, 1990) and a Branch and Bound (B&B) algorithm for the search (Narendra & Fukunaga, 1977). B&B warrants an optimal search if the criterion used is monotonic and is computationally less intensive than exhaustive search. For the search algorithm, one can also consider the suboptimal floating search techniques proposed by Pudil et al. (1994) which offer a good compromise between the simplicity of sequential methods and the relative computational cost of the Branch and Bound algorithm.

For the comparisons in section 6, we have used Epanechnikov kernels for density estimation in (4.3.3), a forward search, and the selection is stopped when the MI increase falls below a fixed threshold (0.99).

5. Model-dependent feature selection for neural networks

Model-dependent feature selection attempts to perform the selection and the processing of the data simultaneously: the feature selection process is part of the training process, and features are sought for optimizing a model selection criterion. This "global optimization" seems to be more attractive than model-independent selection where the adequacy of the two steps is up to the user. However, since the value of the selection criterion depends on the model parameters, it might be necessary to train the NN with different sets of variables: some selection proce-

dures alternate between variable selection and retraining of the model parameters. This forbids the use of sophisticated search strategies which would be computationally prohibitive.

Some specificities of NNs should also be taken into consideration when deriving feature selection algorithms:

• NNs are usually nonlinear models. Since many parametric model-independent techniques are based on the hypothesis that input-output variable dependency is linear or that input variables redundancy is well measured by linear correlation between these variables, such methods are clearly ill-suited to NNs.

• The search space usually has many local minima, and relevance measures will depend on the minimum the NN will have converged to. These measures should be averaged over several runs. For most applications, this is prohibitive and has not been considered here.

• Except for (White, 1989), who derives results on the weight distribution, there is no work in the NN community which might be used for hypothesis testing.

The selection criteria in NN feature selection algorithms are based mainly on heuristic individual feature evaluation functions. Several have been proposed in the literature. We have attempted to classify them by family, in which we find:

• zero-order methods which use only the network parameter values;

• first-order methods using the first derivatives of network parameters;

• second-order methods using the second derivatives of network parameters.

Most feature evaluation criteria will rank variables at a given time, the value of the criterion itself is non-informative. However, we will see that most of these methods work reasonably well.

Feature selection methods with neural networks use mostly backward search, although some forward methods have also been proposed (Moody, 1994; Goutte, 1997). Several methods evaluate and rank features individually without considering their dependencies or correlations. This may be risky when selecting minimum relevant sets of variables. Using the correlation as a simple dependence measure is not enough, since NNs capture nonlinear relationships between variables. On the other hand, measuring nonlinear dependencies is not trivial. Certain authors simply ignore this problem, but others propose to select only one variable at a time and then retrain the network with the newly selected set before evaluating the relevance of the remaining variables. Some of the dependencies the network has discovered among the variables can be taken into account this way.

More critical is the difficulty for defining a sound stop criterion or model choice. Many methods use very crude techniques for stopping the selection, e.g. a threshold on the choice criterion value. Some rank the different subsets using an estimation of the generalization error. This is the expected error performed on future data and is defined as:

$$R=\int r(\boldsymbol{x}, \boldsymbol{y})\ p(\boldsymbol{x}, \boldsymbol{y})\ d\boldsymbol{x}d\boldsymbol{y} \tag{5.0.1}$$

where, in our case, $r(\boldsymbol{x}, \boldsymbol{y})$ is the Euclidean error between the desired and computed outputs. Estimates can be computed using a validation set, cross-validation, or algebraic approximations of this risk like the Final Prediction Error (Akaike, 1970). Several estimates have been proposed in the statistical (Gustafson & Hajlmarsson, 1995) and NN (Moody, 1991 ; Larsen & Hansen, 1994) literature.

For the comparison in section 6, we have used a simple threshold when the authors gave no indication for the stop criterion, and a validation set approximation of the risk otherwise.

5.1 Zero-order methods

For linear regression models, the partial correlation coefficient can be expressed as a simple function of the weights. Although this is not sound for nonlinear models, attempts have been made to use the input weight values in computing the relevance of the variable. This has been observed to be an inefficient heuristic : weights are not easily interpreted in these models.

A more sophisticated heuristic has been proposed by Yacoub and Bennani (1997), using both the weight value and the network structure of a multilayer perceptron. They derived the following criterion :

$$s_i=\sum_{j\in H}\left(\frac{|w_{ji}|}{\sum_{i'\in I}|w_{ji'}|}\sum_{k\in O}\frac{|w_{kj}|}{\sum_{j'\in H}|w_{kj'}|}\right) \tag{5.1.1}$$

where I, H, and O denote the input, hidden, and output layer, respectively.

For a better understanding of this measure, let us suppose that each hidden and output unit incoming weight vector has a unit norm L_1. The above equation can then be written as :

$$S_i=\sum_{o\in O}\sum_{j\in H}|w_{oj}w_{ji}| \tag{5.1.2}$$

In (5.1.2), the inner term is the product of the weights from input i to hidden unit j, and from j to output o. The importance of variable i for output o is the sum of the absolute values of these products over all the paths in the NN from unit i to unit o. The importance of variable i is then defined as the sum of these values over all the outputs. The denominators in (5.1.1) operate as normalizing factors. This is important when using squashing functions, since these functions limit the effect of weight magnitude. Note that this measure will depend on the magnitude of the input. The different variables should then be in a similar range. The two weight layers do have different roles in an MLP that are not reflected in (5.1.1). For example, if the outputs are linear, the normalization should be suppressed in the inner summation of (5.1.1).

They used a backward search and the NN is retrained after each variable deletion, the stop criterion is based on the variation of the performance on a

validation set. Elimination is stopped as soon as performance decreases.

5.2 First-order methods

Several methods evaluate the relevance of a variable by the derivative of the error or of the output with respect to this variable. Such evaluation criteria are easy to compute, and most lead to very similar results. There derivatives measure the local change in the outputs with a given input while the other inputs remain fixed. Since these derivatives are not constant, as they are in linear models, they must be averaged over the training set. For these measures to be fully meaningful, inputs should be independent; and since they average the local sensitivity, the training set should be representative of the input space.

5.2.1 Saliency Based Pruning (SBP)

The evaluation criterion in this backward method (Moody & Utans, 1992) is the variation of the learning error when a variable x_i is replaced by its empirical mean $\bar{x}_i$ (zero, here, since variables are assumed to be centered):

$$S_i = MSE - MSE(\bar{x}_i) \tag{5.2.1}$$

where

$$MSE(\bar{x}_i) = \frac{1}{N}\sum_{l=1}^{N} \| f(x_1^l, \ldots, \bar{x}_i^l, \ldots, x_k^l) - y^l \|^2$$

This is a direct measure of the usefulness of the variable for computing the output. Computing S_i is costly for large values of N, and a linear approximation may be used:

$$S_i = \frac{1}{N}\sum_{l=1}^{N} \frac{\partial \| f(\boldsymbol{x}^l) - y^l \|^2}{\partial x_i} (\bar{x}_i^l - x_i^l) \tag{5.2.2}$$

Variables are eliminated by increasing order of S_i.

For each feature set, as NN is trained and an estimate of the generalization error—a generalization of the Final Prediction Error criterion—is computed. The model with minimum generalization error is selected.

Changes in MSE are not ambiguous only when inputs are not correlated. As this method compute variable relevance just once, it does not take possible correlations between variables into account. Relevance could be computed from the successive NNs in the sequence at a computational extra-cost ($O(k^2)$ S_i computations instead of $O(k)$ in the present method).

5.2.2 Methods using output derivatives

For a linear model, the output derivative with respect to any input is a constant, which is not the case for nonlinear NNs. Several authors have proposed to measure the sensitivity of the network transfer function with respect to input x_i by computing the mean value of the output derivatives with respect to x_i over the

whole training set. In the case of multilayer perceptrons, this derivative can be computed progressively during learning (Hashem, 1992). Since these derivatives may take both positive and negative values, they may compensate and produce an average near zero. Most measures use average squared or absolute derivatives. Dozens of measures based on derivatives have been proposed, and many others could be defined. The following is only a representative sample.

The sum of the derivative absolute values has been used e.g. in Ruck et al. (1990):

$$S_i=\sum_{l=1}^{N}\sum_{j=1}^{g}\left|\frac{\partial f_j}{\partial x_i}(x^l)\right| \tag{5.2.3}$$

For classification, Priddy et al. (1993) remark that since the error for decision j $P_{err}(j/\boldsymbol{x})$ may be estimated by $1-f_j(\boldsymbol{x})$, (5.2.3) may be interpreted as the absolute value of the error probability derivative averaged over all decisions (outputs) and data.

Squared derivatives may be used instead of the absolute values. For example, Refenes et al. (1996) proposed a normalized sum for regression:

$$S_i=\frac{1}{N}\frac{VAR(x_i)}{VAR(f(\boldsymbol{x})-y)}\sum_l\left(\frac{\partial f}{\partial x_i}(x^l)\right)^2 \tag{5.2.4}$$

where VAR stands for variance. They also proposed a series of related criteria, including:

—a normalized standard deviation of the derivatives:

$$S_i=\frac{1}{N^{1/2}}\frac{\left(\sum_l\left(\frac{\partial f}{\partial x_i}(\boldsymbol{x}^l)-\frac{1}{N}\sum_j\frac{\partial f}{\partial x_i}(\boldsymbol{x}^j)\right)^2\right)^{1/2}}{\sum_l\frac{\partial f}{\partial x_i}(\mathbf{x}^l)} \tag{5.2.5}$$

—a weighted average of the derivative absolute values, where the weights reflect the relative magnitude of $\boldsymbol{x}$ and $f(\boldsymbol{x})$:

$$S_i=\frac{1}{N}\sum_l\left|\frac{\partial f}{\partial x_i}(\boldsymbol{x}^l)\cdot\frac{x_i}{f(\boldsymbol{x}^l)}\right| \tag{5.2.6}$$

All these measures are very sensitive to the input space representativeness of the sample set. So several authors have proposed to use a subset of the sample in order to increase the significance of their relevance measure.

In order to obtain robust methods, "pathological" training examples should be discarded. For regression and radial basis function networks, Dorizzi et al. (1996) propose using the 95th percentile of the absolute value of the derivative:

$$S_i=q_{95}\left(\left|\frac{\partial f}{\partial x_i}(\boldsymbol{x})\right|\right) \tag{5.2.7}$$

As this eliminates outlying points, it contributes to the robustness of the measure. Note that the same idea could be used with other relevance measures proposed in

this paper.

Following the same line, Czernichow (1996) proposed a heuristic criterion for regression, estimated on a set of non pathological examples whose cardinality is N'. The proposed selection criterion is:

$$S_i=\frac{\sum_{l=1}^{N'}\left(\frac{\partial f}{\partial x_i}(\boldsymbol{x}^l)\right)^2}{\max_j\left(\sum_{l=1}^{N'}\left(\frac{\partial f}{\partial x_j}(\boldsymbol{x}^l)\right)^2\right)} \tag{5.2.8}$$

For classification, Rossi (1996), following a proposition made by Priddy et al. (1993), considers only those patterns that are near the class boundaries. He proposes the following relevance measure:

$$S_i=\frac{1}{g}\sum_{x^l\in\text{boundary}}\sum_{j=1}^{g}\frac{\left|\frac{\partial f_j}{\partial x_i}(\boldsymbol{x}^l)\right|}{\|\nabla_x f_j(\mathbf{x}^l)\|} \tag{5.2.9}$$

The boundary is defined as the set of points for which $\|\nabla_x f_j(x^l)\|>\varepsilon$, where ε is a fixed threshold. Several authors have also considered the relative contribution of partial derivatives to the gradient as in (5.2.9).

All these methods use a simple backward search.

For the stopping criteria, all use heuristic rules, except for Refenes et al. (1996), who define statistical tests for their relevance measures. For nonlinear NNs, this requires an estimation of the relevance measure distribution. This is very costly and, in our opinion, usually prohibits this approach, even if it is otherwise attractive.

5.2.3 Links between methods

All these methods use simple relevance measures that depend on the gradient of network outputs with respect to input variables. There is no ranking of the different criteria. All that can be recommended are a few reasonable rules like discarding outlying points for robustness, or retraining the NN each time a variable is discarded, and computing new relevance measures for each NN in the sequence, in order to include dependencies between variables. In practice, all these methods give very similar results, as will be shown in section 6.

Table 2 summarizes the main characteristics of relevance measures for the different methods.

5.3 Second-order methods

Several methods evaluate the relevance of a variable by computing weight pruning criteria for the set of weights of each input node. We present three such methods below. The first is a Bayesian approach for computing the weight variance. The other two use the Hessian of the cost function for computing the cost function dependence on input unit weights.

Table 2
Computation of the variable relevance by different methods using the derivative of the network function.

	Derivative used	Task C/R	Data used
(Moody (5.2.1))	$\frac{\partial f}{\partial x_i}$	C/R	All
(Refenes (5.2.5))	$\frac{\partial f}{\partial x_i}$	C/R	All
(Dorizzi (5.2.7))	$\left\|\frac{\partial f}{\partial x_i}\right\|$	C/R	Non pathological data
(Refenes (5.2.6))	$\left\|\frac{\partial f}{\partial x_i}\right\|$	C/R	All
(Czernichow (5.2.8))	$\left(\frac{\partial f}{\partial x_i}\right)^2$	C/R	Non pathological data
(Refenes (5.2.4))	$\left(\frac{\partial f}{\partial x_i}\right)^2$	C/R	All
(Ruck (5.2.3))	$\sum_{j=1}^{g}\left\|\frac{\partial f_j}{\partial x_i}\right\|$	C	All
(Rossi (5.2.9))	$\sum_{j=1}^{g}\left\|\frac{\partial f_j}{\partial x_i}\right\| / \|\nabla_x f_j\|$	C	Boundary between classes

C/R denote Classification and Regression tasks, respectively.

5.3.1 Automatic Relevance Determination (ARD)

This method was proposed by MacKay (1994) in the framework of Bayesian learning. In this approach, weights are considered as random variables and regularization terms are included for each input in the cost function. Assuming that the prior probability distribution of the group of weights for the i^{th} input is a gaussian, the input posterior variance σ_i^2 is estimated (with the help of the Hessian matrix).

ARD has been successful for time series predictions, learning with regularization terms improved the prediction performance. However, ARD has not really been used as a feature selection method, since variables are not pruned during training.

5.3.2 Optimal Cell Damage

Weight pruning techniques have spawned a number of neural selection methods. For the latter, the decision to prune a weight is made according to a relevance criterion often named the weight saliency: the weight is pruned if its saliency is low. Similarly, the saliency for an input cell is usually defined as the sum of its weight saliencies.

$$Saliency\ (x_i) = \sum_{fan-out(i)} Saliency\ (w_j) \tag{5.3.1}$$

where $fan\text{-}out(i)$ is the set of weights of input i.

Optimal Cell Damage (OCD) has been proposed by Cibas et al. (1994a, 1996) (A similar method was also proposed by Mao et al., 1994). This feature selection method is inspired from the Optimal Brain Damage (OBD) weight pruning technique developed by LeCun (1990). In OBD, the connection saliency is defined by:

$$Saliency\ (w_j)=\frac{1}{2}H_{jj}w_j^2=\frac{1}{2}\frac{\partial^2 MSE}{\partial w_j^2}w_j^2, \tag{5.3.2}$$

which is an order two Taylor expansion of MSE variation around a local minimum. The Hessian matrix H can easily be computed using gradient descent, but this may be computationally intensive for large networks. For OBD, the authors use a diagonal approximation for the hessian which can then be computed in $O(N)$. The saliency of an input variable is defined accordingly as:

$$S_i=Saliency\ (x_j)=\frac{1}{2}\sum_{j\in fan-out(i)}\frac{\partial^2 MSE}{\partial w_j^2}w_j^2 \tag{5.3.3}$$

Cibas et al. (1994) proposed (5.3.3) as a selection criterion for eliminating variables. The NN is trained to reach a local minimum. Variables whose saliency is below a given threshold are eliminated. The threshold value is fixed by cross validation. This process is then repeated until no variable is found below the threshold.

This method has been tested on several problems, with satisfactory results. Once again, the difficulty lies in selecting an adequate threshold. Furtermore, since several variables can be eliminated simultaneously whereas only individual variable pertinence measures are used, significant sets of dependent variables may be eliminated.

For stopping, the generalization performance of the NN sequence are estimated via a validation set and the variable set corresponding to the NN with the best performance is chosen.

The hessian diagonal approximation has been questioned by several authors, Hassibi and Stork (1993), for example, proposed a weight pruning algorithm, Optimal Brain Surgeon (OBS), which is similar to OBD, but uses the whole hessian for computing weight saliencies. Stahlberger and Riedmiller (1997) proposed a feature selection method similar to OCD except that it takes into account non-diagonal terms in the hessian.

For all these methods, saliency is computed using the error variation on the training set as a performance measure. Weight estimation and model selection both use the same data set, which is not optimal. Pedersen et al. (1996) propose two weight pruning methods, γOBD and γOBS, that compute weight saliency according to an estimate of the generalization error: the Final Prediction Error (Akaike, 1970). Similarly to OBD and OBS, these methods could also be transformed into feature selection methods.

5.3.3. Early Cell Damage (ECD)

Using a second-order Taylor expansion, as in the OBD family of methods, is justified only when a local minimum is reached and the cost is locally quadratic in this minimum. Both conditions are rarely met in practice. Tresp et al. (1997) propose two weight pruning techniques from the same family, dubbed EBD (Early Brain Damage) and EBS (Early Brain Surgeon). They use a heuristic justification for early stopping by adding a new term in the saliency computation. These methods can be extended for feature ranking. We will use ECD (Early Cell Damage) to denote the EBD extension. For ECD, the saliency of input i is defined as:

$$S_i = \sum_{j \in fan-out(i)} \frac{1}{2} \frac{\partial^2 MSE}{\partial w_j^2} w_j^2 - \frac{\partial MSE}{\partial w_j} w_j + \frac{1}{2} \frac{\left(\frac{\partial MSE}{\partial w_j}\right)^2}{\frac{\partial^2 MSE}{\partial w_j^2}} \tag{5.3.4}$$

The algorithm we propose is slightly different from OCD: only one variable is eliminated at a time, and the NN is retrained after each deletion.

For choosing the "best" set of variables, we have used a variant of the "selection according to an estimate of the generalization error" method. This estimate is computed using a validation set. Since the performance may oscillate without changing significantly, several subsets may have the same performance (e.g. see figure 1). Using a Fisher test, we compare the performance of a given model with that of the best model and then select the set of networks whose performance is similar to the best. From these, we choose the one with the smallest number of input variables.

6. Experimental comparison

We now present comparative performance of different feature selection methods. These methods are not easy to compare, as there is no single measure that characterizes the importance of each input. The selection accuracy also depends on the search technique and on the criterion for choosing variable subset. In the case of NNs, these different steps rely on heuristics that could be exchanged from one method to another. The NNs used are multilayer perceptrons with one hidden layer of 10 neurons.

The comparison we provide here is not intended to be a final ranking of the different methods, but to illustrate the general behavior of some of them, which have been described before. We have used two synthetic classification problems that illustrate different difficulties of variable selection. In the first, the boundaries are "nearly" linear and there are dependent variables as well as pure noise variables. The second problem has nonlinear boundaries, and independent or correlated variables can be chosen.

Table 3
Performance comparison of different variable selection methods on the noisy wave problem

Method	p*	Selected Variables	Perf.
None	40	11111111111111111111 11111111111111111111	82.51% [81.35-83.62]
Stepdisc (4.2.4)	14	00011011111111011100 00000000000000000000	85.35% [84.26-86.38]
(Bonnlander (4.3.3))	12	00001110111111110000 00000000000000000000	85.12% [84.02-86.15]
(Yacoub (5.1.1))	16	00011111111111111100 00000000000000000000	85.16% [84.07-86.19]
(Moody (5.2.1))	16	00011111111111111100 00000000000000000000	85.19% [84.10-86.22]
(Ruck (5.2.3)) (Dorizzi (5.2.7))	18	01111111111111111100 00000000000000000000	85.51% [84.43-86.53]
(Czernichow (5.2.8))	17	01011111111111111100 00000000000000000000	85.67% [84.59-86.69]
(Cibas (5.3.3))	9	00000111110111000000 00000000000000000000	82.26% [81.09-83.37]
(Leray (5.3.4))	11	00000111111111100000 00000000000000000000	84.56% [83.45-85.61]

The first problem was originally proposed by Breiman et al. (1984). It is a three-class waveform classification problem with 19 noisy dependent features. We have also used a variation of this problem where 21 pure noise variables are added to the 19 initial variables (there are 40 inputs for this variant). The training set has 300 patterns and the test set 4300. A description of this problem is provided in the appendix. The performance of the optimal Bayes classifier estimated on this test set is 86% correct classification. A performance comparison appears in tables 3 and 4 for these two instances.

For the noisy problem, all methods do eliminate pure noise variables. Except for the two methods at the bottom of table 3 which give slightly lower performance and select fewer variables, all give similar values around 85% correct. Stepdisc also gives good performance since in this problem data have a unimodal distribution and the boundaries are nearly linear. For the non-noisy problem, the performance and methods ordering change. The two techniques at the bottom of table 4 offer now slightly better performance.

Figure 1 shows performance curves for two methods, OCD and ECD, estimated on a validation set. Since we have used a single validation set, there are small fluctuations in the performance. Some form of cross validation should be used in order to get better estimates, the test strategy proposed for ECD also looks attractive in this case. It can be seen that, for this problem, performance is more or less similar during the backward elimination (it rises slightly) and drops off quickly when relevant variables are removed.

Table 4
Performance comparison of different variable selection methods on the original wave problem

Method	p*	Selected Variables	Perf.
None	21	111111111111111111111	85.28% [84.19-86.31]
Stepdisc (4.2.4)	14	00111010111111011100	84.19% [83.07-85.25]
(Bonnlander (4.3.3))	8	00000110011101010000	83.05% [81.90-84.14]
(Yacoub (5.1.1))	18	01111111111111111100	85.46% [84.38-86.48]
(Moody (5.2.1))	16	00011111111111111100	85.63% [84.65-86.65]
(Ruck (5.2.3)) (Dorizzi (5.2.7))	12	00011110111111010000	84.65% [83.54-85.70]
(Czernichow (5.2.8))	10	00011010101111010000	82.58% [81.42-83.68]
(Cibas (5.3.3))	15	00101111111111110100	85.23% [84.14-86.26]
(Leray (5.3.4))	13	00001111111111110000	85.67% [84.59-86.69]

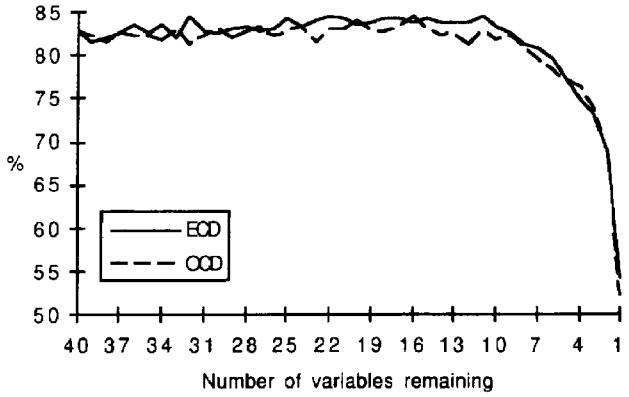

Fig. 1 Performance comparison of two variable selection methods (OCD and ECD) according to the number of variables remaining for the noisy wave problem.

Figure 2 gives the performance distribution of different variable selection methods for the original wave problem (y axis) and the percentage of selected variables (x axis). The best methods are those with the best performance, and the lower number of variables. In this problem, "Leray" is satisfactory (see figure 2). "Yacoub" does not delete enough variables, while "Bonnlander" deletes too many.

The second problem is a two class problem in a 20-dimensional space. The classes are distributed according two gaussians with, respectively, $\mu_1=(0,\ldots,0)$, $\Sigma_1=4*I$, $\mu_2=(0, 1, 2,\ldots, 19)/\alpha$($\alpha$ is chosen so that $\|\mu_1\ \mu_2\|=2$) and $\Sigma_2=I$. In this prob-

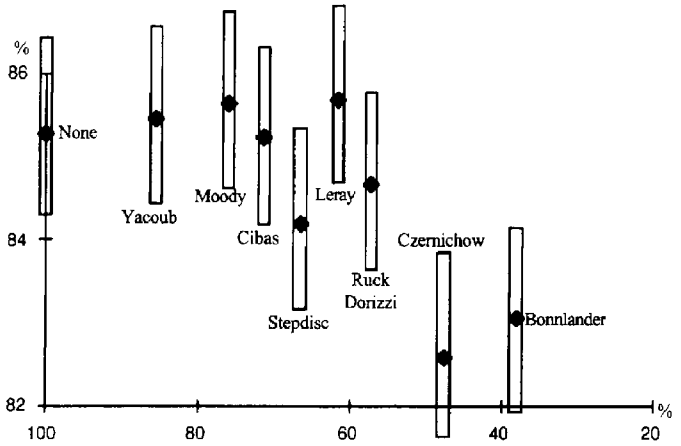

Fig. 2 Performance comparison of different variable selection methods vs. percentage of selected variables on the original wave problem. x axis : percentage of variables selected, y axis : percentage of correct classification.

Table 5
Performance comparison of different variable selection methods on the "two gaussian" problem with uncorrelated variables.

Method	p*	Selected Variables	Perf.
None	20	11111111111111111111	94.80% [94.15-95.35]
Stepdisc (4.2.4)	17	10001111111111111111	94.88% [94.23-95.43]
(Bonnlander (4.3.3))	5	00001000000000011011	90.60% [89.76-91.38]
(Yacoub (5.1.1))	18	01011111111111111111	94.86% [94.21-95.44]
(Moody (5.2.1))	9	01000100011000110111	92.94% [92.20-93.62]
(Ruck (5.2.3))	10	00000000101101111111	94.86% [94.21-95.44]
(Dorizzi (5.2.7))	11	00000000101111111111	94.66% [94.00-95.25]
(Czernichow (5.2.8))	9	00000000011011111111	94.02% [93.33-94.02]
(Cibas (5.3.3))	14	01001110010111111111	94.62% [93.96-95.21]
(Leray (5.3.4))	15	01011011101110111111	94.08% [93.39-94.70]

lem, variable relevance is ordered by index : x_1 is useless, x_{i+1} is more relevant than x_i.

Table 5 shows that Stepdisc is not suitable for this nonlinear boundary : it is the only method that selects x_1, which is useless for this problem. We can see in

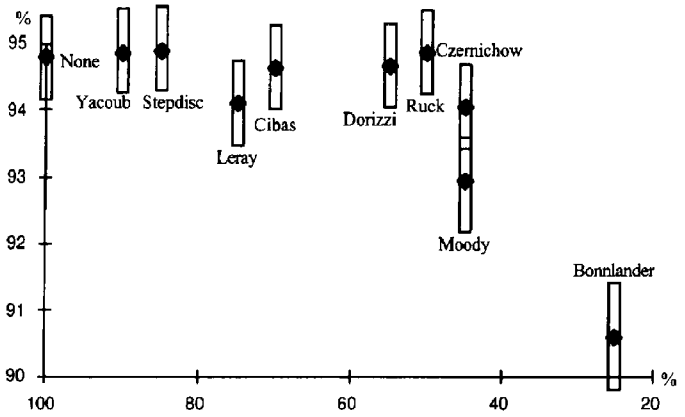

Fig. 3 Performance comparison of different variable selection methods vs. percentage of selected variables on the two gaussian problems with uncorrelated variables. x axis: percentage of variables selected, y axis: percentage of correct classification.

Table 6
Performance comparison of different variable selection methods on the "two gaussian" problem with correlated variables.

Method	p*	Selected Variables	Perf.
None	20	11111111111111111111	90.58% [89.74-91.36]
Stepdisc (4.2.4)	11	00001101011010110111	91.96% [91.17-92.68]
(Bonnlander (4.3.3))	5	00001001010000100001	88.48% [75.57-89.34]
(Ruck (5.2.3))	10	00011001011110100011	91.06% [90.24-91.82]
(Leray (5.3.4))	7	00000010101010100011	90.72% [89.88-91.49]

figure 3 that Bonnlander's method does not select too many variables, whereas Yacoub's stop criterion is too rough and does not delete enough variables.

In another experiment, we replaced the I matrix in $\sum_1$ and $\sum_2$ by a block diagonal matrix. Each block is 5×5 so that there are four groups of five successive correlated variables in the new problem.

Table 6 gives the results of some representative methods for this problem:

- Stepdisc's model still performs well but selects many correlated variables,
- Bonnlander's method selects only five variables and gives significantly lower results,
- Ruck's method obtains good performance but selects certain correlated variables,
- Leray's method, with its retraining after each variable deletion, finds models with good performance and few variables (seven, compared to 10 and 11 for

Ruck and Stepdisc).

7. Conclusion

We have reviewed variable selection methods developed in the field of Neural Networks. The main difficulty here is the NNs are nonlinear systems that do not use explicit parametric hypothesis. As a consequence, selection methods rely heavily on heuristics for the three variable selection steps: relevance criterion, search procedure (NN variable selection uses mainly backward search), and choice of the final model. We first discussed the main difficulties involved in developing each of these steps. We then introduced different families of methods and discussed their strengths and weaknesses. We believe that a variable selection method must remain computationally feasible in order to be useful, and have therefore not considered techniques that rely on computer-intensive procedures such as bootstrap at each step of the selection. Instead, we have proposed a series of rules which could be used to enhance some of the methods described at a reasonable extra computational cost, for example: retraining each NN in the sequence and computing the relevance for each of these NN, in order to bring out correlations between variables; simple estimates of the generalization error, which may be used to evaluate a variable subset; and simple tests on these estimates, to choose minimum variable sets (section 5.3.3). Finally we compared representative NN selection techniques on synthetic problems.

Appendix: Waveform problem

This problem was proposed by Breiman et al. (1984). Three vectors or *waveforms* are given in 21 dimensions, H^i, $i=1, ..., 3$. Patterns in each class are defined in $\Re^{21}$ as random convex combinations of two of these vectors (waves (1, 2), (1, 3), (2, 3) respectively for class 1, 2 and 3).

The problem is then to classify these patterns into one of the three classes. More precisely, patterns are generated according to:

$$x_i=\frac{uH_i^m+(1-u)H_i^n}{5}+\varepsilon_i \qquad 0\leq i\leq 20$$

where x_i denotes the i^{th} component of a pattern $\boldsymbol{x}$, u is a uniform random variable in $[0, 1]$, ε_i is generated according to a normal distribution $N(0, 1)$, m and n identify the two waves used in this combination, i.e. the class of pattern $\boldsymbol{x}$.

For the noisy problem, 19 additional components are added to the 21 components of the above vectors:

$$x_i=\varepsilon_i \qquad 21\leq i\leq 40$$

The training, validation, and test sets have 300, 1000, 4300 elements, respectively.

REFERENCES

Akaike, H. (1970). Statistical Predictor Identification, *Ann. Inst. Statist. Math.*, **22**, 203-217.

Battiti, R. (1994). Using Mutual Information for Selecting Features in Supervised Neural Net Learning, *IEEE Transactions on Neural Networks*, **5**(4), 537-550.

Baxt, W.G. & White, H. (1995). Bootstrapping confidence intervals for clinical input variable effects in a network trained to identify the presence of acute myocardial infraction, *Neural Computation*, **7**, 624-638.

Bonnlander, B.V. & Weigend, A.S. (1994). Selecting Input Variables Using Mutual Information and Nonparametric Density Evaluation, *Proceedings of ISANN'94*, 42-50.

Breiman, L., Friedman, J., Olshen, R. & Stone, C. (1984). *Classification and Regression Trees*. Wadsworth International Group.

Cibas, T., Fogelman Soulié, F., Gallinari, P. & Raudys, S. (1994a). Variable Selection with Optimal Cell Damage. *Proceedings of ICANN'94*.

Cibas, T., Fogelman Soulié, F., Gallinari, P. & Raudys, S. (1996). Variable Selection with Neural Networks. *Neurocomputing*, **12**, 223-248.

Czernichow, T. (1996). Architecture Selection through Statistical Sensitivity Analysis. *Procceings of ICANN'96*, Bochum, Germany.

Dorizzi, B., Pellieux, G., Jacquet, F., Czernichow, T. & Munoz, A. (1996). Variable Selection Using Generalized RBF Networks: Application to the Forecast of the French T-Bonds. *Proceedings of IEEE-IMACS'96*, Lille, France.

Fraser, A.M. & Swinney, H.L. (1986). Independent Coordinates for Strange Attractors from Mutual Information, *Physical Review* A, **33** (2), 1134-1140.

Goutte, C. (1997). Extracting the Relevant Decays in Time Series Modelling, *Neural Networks for Signal Processing VII*, Proceedings of the IEEE Workshop.

Gustafson & Hajlmarsson (1995). 21 maximum likelihood estimators for model selection. *Automatica*.

Habbema, J.D.F & Hermans, J. (1977). Selection of Variables in Discriminant Analysis by F -statistic and Error Rate, *Technometrics*, **19** (4), 487-493.

Härdle, W. (1990). *Applied Nonparametric Regression*. Cambridge University Press. Econometric Society Monograph n. 19.

Hashem, S. (1992). Sensitivity Adalysis for Feedforward Artificial Neural Networks with Differentiable Activation Functions. *Proceedings 1992 International Joint Conference on Neural Networks IJCNN92* I, 419-424.

Hassibi, B. & Stork, D.G. (1993). Second Order Derivatives for Network Pruning: Optimal Brain Surgeon *Neural Information Processing Systems*, **5**, 164-171.

Kittler, (1986). Feature Selection are Extraction, Chaptre 3 in *Handbook of Pattern Recogntion and Image Processing*, Eds. Tzay Y. Young, King-Sun Fu, Academic Press. 59-83.

Larsen, J. & Hansen, L.K. (1994). Generalized perfomance of regularized neural networks models. *Proceedings of the 1994 IEEE Workshop on Neural Networks for Signal Processing*. 42-51.

LeCun, Y., Denker, J.S. & Solla, S.A. (1990). Optimal Brain Damage. *Neural Information Processing Systems*, **2**, 598-605.

MacKay, D.J.C. (1994). Bayesian Non-linear Modelling for the Energy Prediction Competition. *ASHRAE Transactions*. 1053-1062.

Mao, J., Mohiuddin, K. & Jain, A.K. (1994). Parsimonious Network Design and Feature Selection Through Node Pruning. *Proceedings of the 12th International Conference on Pattern Recognition*. 622-624.

McLachlan, G.J. (1992). *Discriminant Analysis and Statistical Pattern Recognition*, Wiley-Interscience publication.

Moody, J. (1991). Note on generalization, regularization and architecture selection in nonlinear learning systems. *Proceedings of the first IEEE Workshop on Neural Networks for Signal Processing*. 1-10.

Moody, J. & Utans, J. (1992). Principled Architecture Selection for Neural Networks: Application to Corporate Bond Rating Prediction. *Neural Information Processing Systems*, **4**.

Moody, J. (1994). Prediction Risk and Architecture Selection for Neural Networks, in *From*

Statistics to Neural Networks—Theory and Pattern Recognition Applications, Eds V. Cherkassky, J.H. Friedman, H. Wechsler, Springer-Verlag.

Narendra, P.M. & Fukunaga, K. (1977). A Branch and Bound Algorithm for Feature Subset Selection. *IEEE Transactions on Computers*, **26** (9), 917-922.

Pedersen, M.W., Hansen, L.K. & Larsen, J. (1996). Pruning with generalisation based weight saliencies: γOBD, γOBS. *Neural Information Processing Systems*, **8**.

Priddy, K.L., Rogers, S.K., Ruck, D.W., Tarr, G.L. & Kabrisky, M. (1993). Bayesian Selection of Important Features for Eeedforward Neural Networks. *Neurocomputing*, **5**, 91-103. Elsevier ed.

Pudil, P., Novovicova, J. & Kittler, J. (1994). Floating search methods in feature selection. *Pattern Recognition Letters*, **15** 1119-1125.

Refenes, A.N., Zapranis, A. & Utans, J. (1996). Neural Model Identification, Variable Selection and Model Adequacy. *Neural Networks in Financial Engineering*, Proceedings of NnCM-96.

Rossi, F. (1996). Attribute Suppression with Multi-Layer Perceptron. *Proceedings of IEEE-IMACS'96*, Lille, France.

Ruck, D.W., Rogers, S.K. & Kabrispy, M. (1990). Feature Selection Using a MultiLayer Perceptron. *J. Neural Network Comput.*, **2** (2), 40-48.

Stahlberger, A. & Riedmiller, M. (1997). Fast Network Pruning and Feature Extraction Using the Unit-OBS Algorithm. *Neural Information Processing Systems*, **9**, 655-661.

Thompson, M.L. (1978). Selection of Variables in Multiple Regression. Part I: A Review and Evaluation, *International Statistical Review*, **46**, 1-19, Selection of Variables in Multiple Regression. Part II: Chosen Procedures, Computations and Examples, *International Statistical Review*, **46**, 129-146.

Tresp, V., Neuneier, R. & Zimmermann, G. (1997). Early Brain Damage. *Neural Information Processing Systems*, **9**, 669-675.

Van de Laar, P., Gielen, S. & Heskes, T. (1997). Input Selection with Partial Retraining. *Proceedings of ICANN'97*.

White, H. (1989). Learning in Artificial Neural Networks: A Statistical Perspective. *Neural Computation*, **1**, 425-464.

Wilks, S.S. (1963). *Mathematical Statistics*, Wiley, New York.

Yacoub, M. & Bennani, Y. (1997). HVS: A Heuristic for Variable Selection in Multilayer Artificial Neural Network Classifier. *Proceedings of ANNIE'97*. 527-532.

(Received March 1998, Revised September 1998)