Journal of Machine Learning Research 3 (2003) 1399-1414 Submitted 5/02; Published 3/03

Ranking a Random Feature for Variable and Feature Selection

Hervé Stoppiglia HERVE_STOPPIGLIA@PECHINEY.COM
Informatique Caisse des Dépôts et Consignations
113, rue Jean Marin Naudin
F – 92220 Bagneux, France

Gérard Dreyfus GERARD.DREYFUS@ESPCI.FR
Rémi Dubois REMI.DUBOIS@ESPCI.FR
Yacine Oussar YACINE.OUSSAR@ESPCI.FR
ESPCI, Laboratoire d'Électronique
10, rue Vauquelin
F – 75005 Paris, France

Editors: Isabelle Guyon and Andrè Elisseeff

Abstract

We describe a feature selection method that can be applied directly to models that are linear with respect to their parameters, and indirectly to others. It is independent of the target machine. It is closely related to classical statistical hypothesis tests, but it is more intuitive, hence more suitable for use by engineers who are not statistics experts. Furthermore, some assumptions of classical tests are relaxed. The method has been used successfully in a number of applications that are briefly described.

Keywords: Model selection, variable selection, feature selection, kernel, classification, neural networks, leave-one-out, Gram-Schmidt orthogonalization, statistical tests, information filtering

1. Introduction

The present paper addresses (i) the problem of variable selection for polynomials, and (ii) the problem of selecting explicitly computed kernels such as radial basis functions or wavelets. It is thus essentially a filter method, although it can be used indirectly for selecting a learning machine, e.g. for selecting the inputs and the hidden neurons of neural networks.

Assume that a database is available, including measurements of a set of candidate variables, from which a set of features are computed (for linear machines, the variables are identical to the features). The latter can be ranked in order of decreasing relevance to the output; only the most relevant features, i.e., the top features of the list, should be selected; the question that we address here is that of setting the boundary between the "top" and the "bottom" features, i.e., those which should be selected and those which should be discarded, given the available experimental data.

The following intuitive method, whose close relation to statistical tests will be proved in Section 4, is discussed in the present paper: append to the set of candidate features a "probe" feature, which is a random variable; if the amount of available data were infinite, this feature should be ranked last, or should be ranked as low as other irrelevant features, if any. Since the amount of available data is finite, the probe feature will appear somewhere in the ranked feature list; all features that are ranked below the probe should be discarded. Actually, since the probe is a random variable, its rank in the list is a random variable too. Therefore, the decision of keeping or discarding a given feature is based on the probability that this feature be ranked higher or lower than the probe. In the spirit of classical hypothesis tests, the designer of the model will choose a risk of selecting a feature although it is less relevant than a random one (or a risk of discarding a feature although it is more relevant than a random one), and will base its decision on that risk.

The first part of the paper recalls the Gram-Schmidt orthogonalization procedure, whereby the candidate features are ranked in order of decreasing relevance to the measured process output, or concept. Section 3 describes the use of the probe feature, the computation of the probability distribution function of its rank in the list, and its use for feature selection. The relation between the present procedure and Fisher's test is subsequently derived. In Section 4, the extension to models that are nonlinear with respect to their parameters is described. Section 5 discusses several applications of the method, both academic and industrial. Finally, we discuss the limitations of the method and show that it is potentially useful in a larger framework.

2. Feature Ranking

A general, lucid discussion of the feature ranking and feature selection problems can be found in the paper of Guyon et al. (2002). The present section is devoted to recalling briefly the use of the Gram-Schmidt orthogonalization procedure for ranking the variables of a model that is linear with respect to its parameters; it was first described by Chen et al. (1989); it was first used in the machine learning context for RBF networks by Chen et al. (1991), for neural networks by Urbani et al. (1993), and for wavelet networks by Oussar (1998), Oussar and Dreyfus (2000); variants of the method were developed recently under the name of Matching Pursuit (see for instance Vincent and Bengio, 2001).

We consider a model with Q candidate features; a data set containing N input-output pairs (measurements of the output of the process to be modelled, and of the candidate features - or of the candidate variables for linear models) is available. We denote by $\boldsymbol{x}^i = [x_1^i, x_2^i, ..., x_N^i]^T$ the vector of values of feature i, or of input i. We denote by $\boldsymbol{y}_p$ the N-vector of the measured values of the output of the process to be modelled. We consider the (N, Q) matrix $X = [\boldsymbol{x}^1, \boldsymbol{x}^2, ..., \boldsymbol{x}^Q]$. The model can be written as $\boldsymbol{y} = X\theta$, where θ is the vector of the parameters of the model.

The first iteration of the procedure consists in finding the feature vector that best explains the concept, i.e., which has the smallest angle with the process output vector in the N-dimensional space of observations. To this end, the following quantities are computed

$$\cos^2\left(\boldsymbol{x}^k, \boldsymbol{y}_p\right) = \frac{\left(\boldsymbol{x}^k . \boldsymbol{y}_p\right)^2}{\left\|\boldsymbol{x}^k\right\|^2 \left\|\boldsymbol{y}_p\right\|^2}, \quad k = 1 \text{ to } Q \tag{1}$$

and the vector $\boldsymbol{x}^k$ for which this quantity is largest is selected. In order to discard the part of the concept that is explained by the first selected vector, all remaining candidate inputs, and the output vector, are projected onto the null subspace (of dimension N-1) of the selected feature. In that subspace, the projected input vector that best explains the projected output is selected, and the Q-2 remaining feature vectors are projected onto the null subspace of the first two ranked vectors. The procedure (termed "classical Gram-Schmidt" algorithm) terminates when all Q input vectors are ranked, or when a stopping criterion is met; the main point of the present paper is the description of a new stopping criterion for that procedure.

In addition, the procedure computes the parameters of the model that are optimum in the least squares sense, so that a model is built while the feature selection procedure is performed. It provides valuable information:

- if the resulting linear-in-its-parameters model does not perform well, the validity of the result of the selection procedure should be questioned; this point will be developed below;
- if the resulting linear-in-its-parameters model performs well, the selected features, or variables derived from the selected features, may be fed to a nonlinear-in-its-parameters model, which may be more parsimonious, hence provide better generalization.

The performance of the linear-in-its-parameters model can be assessed efficiently by making use of the analytic expression of the leave-one-out error computed by using the Sherman-Morrisson-Woodbury theorem (Myers, 1990), which was extended to nonlinear-in-their-parameters models by Monari (1999), Monari and Dreyfus (2000, 2002).

For improved numerical stability, it is recommended to use a slightly different procedure, termed "modified Gram-Schmidt" (Bjoerck, 1967). Full algorithmic descriptions of both the classical Gram-Schmidt and the modified Gram-Schmidt algorithms are available in the paper of Chen et al. (1989).

Given a set of Q candidate features, there are 2^Q possible models. The above procedure allows us to consider only Q models for selection: the model with the feature ranked first, the model with the first two features, etc. The price paid for that complexity reduction is the fact that there is no guarantee that the best model is among the Q models generated by the procedure. It can be shown that the procedure is almost optimal (de Lagarde, 1983).

3. Feature Selection

The main point of the present paper is the presentation of a stopping criterion, which exempts the model designer from ranking all parameters.

Assume that a "probe" feature, which is simply a realization of a random variable, is ranked, just as all other candidate features, by the procedure described in the previous section. It would be natural to discard all features that are ranked below the realization of the probe. However, the rank of the probe feature is actually a random variable, whose cumulative distribution function can be computed exactly as shown below. Once the cumulative distribution function is available, one has to choose an acceptable value of the risk that a random variable might explain the concept more efficiently than one of the selected features, i.e., the risk that a feature might be kept although, given the available data, it might be less relevant than the probe.

Therefore, at each step of the Gram-Schmidt orthogonalization, the procedure is the following:

- after orthogonalization (by classical of modified Gram-Schmidt), pick the projected candidate feature (not selected at previous steps) that has the smallest angle with the projected output,
- compute the value of the cumulative distribution function as described in the next section,
- if that value is smaller than the risk, keep the feature and perform the next step of Gram-Schmidt orthogonalization

- if that value is larger than the risk, discard the feature under consideration and terminate the procedure.

The choice of the risk is problem-dependent: if data is sparse, the model should be as parsimonious as possible, hence a low value of the risk should be chosen, in order to make sure that only relevant inputs are present (but some features with low relevance might be missed); conversely, if data is abundant, a higher risk may be acceptable (but some irrelevant features might be kept).

3.1 Computation of the Cumulative Distribution Function of the Rank of the Probe

We proceed to prove that the cumulative distribution function of the squared cosine of the angle between a given vector and a random vector can be computed exactly, and that the cumulative distribution function of the rank can be derived from that result.

The first step is the computation of the probability distribution function of the squared cosine of the angle φ between a fixed vector and a vector whose components are normally distributed, in a space of dimension ν. It can be expressed as:

$$f_\nu(x) = \frac{\Gamma\left(\frac{\nu}{2}\right)}{\Gamma\left(\frac{1}{2}\right)\Gamma\left(\frac{\nu-1}{2}\right)} \frac{(1-x)^{\frac{\nu-3}{2}}}{\sqrt{x}} \tag{2}$$

where $\Gamma(.)$ is the gamma function, with $x = \cos^2\varphi$, $\nu \geq 2$ and $0 \leq x \leq 1$. $f_\nu(x)$ is a beta-function with a = 1/2 and b = (ν-1)/2 (see for instance Mood et al., 1974).

The cumulative distribution function $F_\nu(\cos^2\varphi)$ is obtained by integration of relation (2). It can be computed exactly as indicated in Appendix A. From the cumulative distribution function, the probability that the angle between a probe and a fixed vector be smaller than a given angle φ is easily derived as

$$P_\nu(cos^2\varphi) = 1 - F_\nu(cos^2\varphi) \tag{3}$$

for $\nu \geq 2$.

Finally, the cumulative distribution function of the rank of a probe can be derived as follows. At iteration n, n candidate features have been ranked, and a new feature is chosen among the $Q-n$ remaining ones. We denote by φ_n the angle (in a space of dimension $\nu = N - n$) between the selected projected feature and the projected output, and by Π_n the probability that the angle between a realization of the probe and the projected output be smaller than φ_n: $\Pi_n = P_{N-n}(cos^2\varphi_n)$. We denote by G_{n-1} the probability that a realization of the probe be more relevant than one of the n-1 candidate features selected at the n-1 previous steps of the Gram-Schmidt procedure. The probability that a realization of the probe be less relevant than one of the n-1 previous features is equal to 1 - G_{n-1}. Therefore, the probability that a realization of the probe be more relevant than the n-1 previous features but less relevant than the n-th feature is equal to

$$P_{N-n}(cos^2\varphi_n)(1 - G_{n-1})$$

Hence, the probability that a realization of the probe be more significant than one of the n features selected after iteration n is given by

$$G_n = G_{n-1} + P_{N-n}(\cos^2\varphi_n)(1 - G_{n-1}) \tag{4}$$

with $G_0 = 0$.

As a first illustration, we consider (de Lagarde, 1983) a data set of 15 observations generated by the following simulated process:

$$\boldsymbol{y}_p = X\theta + \omega \tag{5}$$

where $\boldsymbol{y}_p$ and ω are 15-dimensional vectors, θ is a 10-dimensional vector, X is a (15, 10) matrix. The data generating process has actually 5 relevant features ($\{x^1$ to $x^5\}$) only, chosen from a normal distribution: $\theta_i \neq 0$ for i = 1 to 5, $\theta_i = 0$ for i = 6 to 10. The components of vector ω are Gaussian distributed with zero mean and variance 2 10^{-2}. The input vectors $\boldsymbol{x}^j$ (j = 1 to 10) are also chosen from normal distributions.

Figure 1 shows the computed cumulative distribution function of the rank of a realization of the probe. If a model with 5 features is selected, the probability that a random feature might explain the output better than one of the 5 features chosen is lower than 10%. As expected, the five selected features are the features with non-zero parameters of the data generating process.

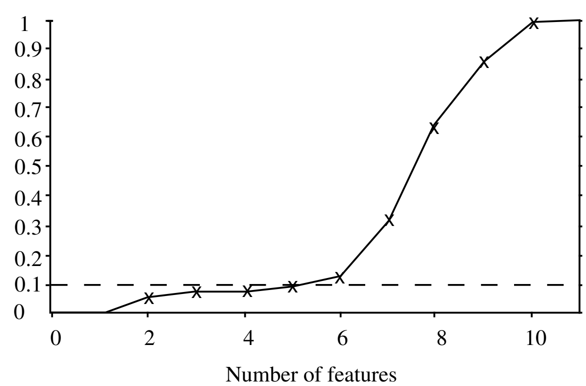

Figure 1: Computed cumulative distribution function of the rank of the probe feature, as a function of the number of selected features. The 5 relevant inputs of the generating procedure are selected if a risk of 10% is chosen.

3.2 Summary

In the present section, we summarize the feature selection procedure for a linear-in-its-parameters model.

First, one should choose a risk r of selecting a feature that is less relevant than a random feature.

At step n of the orthogonalization algorithm ($n < Q$):

- choose the n-th candidate feature in the ranked list,
- compute $\cos^2\varphi_n$ from relation (1), $F_\nu(\cos^2\varphi_n)$ from Appendix A, $P_{N-n}(\cos^2\varphi_n)$ from relation (3), G_n from relation (4),
- if $G_n > r$, select the n-th feature and proceed to step n+1; otherwise, terminate the procedure.

4. Relation to Fisher's Test

Fisher's test is a classical statistical (frequentist) approach to the selection of models that are linear with respect to their parameters. It relies on the assumption that the model is *complete*, i.e., that the regression function belongs to the family of functions within which the model is searched for. If one (or more) input is irrelevant, the corresponding parameter(s) of the model should be equal to zero. Therefore, the hypothesis that is tested is the fact that one or more parameters are equal to zero.

Fisher's test compares a sub-model to the complete model. Other tests, such as the Likelihood Ratio Test (Goodwin and Payne, 1977) and the Logarithm Determinant Ratio Test (Leontaritis and Billings, 1987) compare models that are not thus related. It is proved that these tests are asymptotically equivalent to Fisher's test (Soederstroem, 1977).

In principle, the complete model (with Q parameters) should be compared, using Fisher's test, to all 2^Q sub-models. Using feature ranking with the Gram-Schmidt method as explained above, the number of comparisons can be reduced to Q.

It is shown in Appendix B that the random variable that is used by Fisher's test to discriminate between the null hypothesis and the alternative one can be derived from the probe feature method. The latter thus appears as an alternative to Fisher's test, which (i) gives the model designer a clear explanation as to why features should be discarded (given the available data) and (ii) does not rely on the assumption that the complete model actually contains the regression.

5. Application to the Selection of Models that are Nonlinear with Respect to their Parameters

Since this procedure applies only to models that are linear with respect to their parameters, it is not directly applicable to the selection of the inputs of nonlinear-in-their-parameters models: multilayer perceptrons, radial basis function networks, wavelet networks, etc. This drawback can be circumvented by noting that a variable which is irrelevant is irrelevant irrespective of the model, provided that the latter can learn the task; therefore, the variables can be first selected with a model linear with respect to its parameters (a polynomial model for instance), and subsequently used as inputs to a neural net, thereby taking advantage of the parsimony of the latter. In the next section, we describe an example where the relevant variables in a XOR classification problem are discovered among many irrelevant variables, by selecting the inputs of a polynomial model of degree 2 that solves the problem.

Therefore, the procedure is as follows:

- perform feature selection on a model that is linear with respect to its parameters, e.g. a polynomial; check that the model gives reasonable results on the training set (if there is no point in checking its generalization ability), or assess the generalization performance by computing the leave-one-out error as mentioned above;
- if the linear-in-its-parameters model can learn the task, use the variables that appear in the selected monomials as inputs to a nonlinear-in-its-parameters model;
- if the polynomial model cannot learn the task, increase the degree of the polynomial.

The main limitation of the procedure is the increase of the complexity of the polynomial with the number of features to be ranked. It might become intractable if thousands of features were to be ranked.

Furthermore, the procedure can be applied to the estimation of the number of hidden neurons. Consider a feedforward neural network, with a single layer of hidden neurons and a linear output neuron, whose inputs are known. We address the problem of estimating the minimum number of hidden neurons required to perform a nonlinear regression on the available data. The application of the method described above is straightforward since the output of the hidden layer may be considered as the input of a linear model. The method can be applied either destructively, starting with a large model and using the selection method to discard useless neurons, or constructively, adding neurons until a hidden neuron is considered to be less relevant than a "probe" hidden neuron. In the first case, one trains a large neural network, performs the above procedure by using as candidate features the outputs of the hidden neurons, and discards the neurons that are less relevant than the probe, with the chosen risk. The new architecture is retrained fully, and the procedure is iterated until no irrelevant neuron is detected. In the second case, one starts from a minimal network and increments the number of hidden neurons until an irrelevant neuron is detected.

In the same spirit, the procedure can be used for selecting RBF or wavelet networks: a library of RBF's (resp. wavelets), with fixed centers and widths (resp. translations and dilations) is created, and the procedure is applied to select the most relevant kernels. The surviving kernels are subsequently trained (i.e., their centers or translations, and widths or dilations, are adjusted). If necessary, the process can be iterated for further selection (Chen et al., 1989, Oussar and Dreyfus, 2000).

6. Numerical Illustrations and Applications

This section is devoted to the presentation of academic problems and of several applications in which the method proved successful.

6.1 Academic Problems

Before proceeding to industrial and financial applications, we illustrate the method's use on three problems of academic interest.

6.1.1 Variable Selection in a Synthetic Classification Problem

In http://www.clopinet.com/isabelle/Projects/NIPS2001/#dataset, a database is generated as follows: a linear discriminant function is chosen with random parameters in 2-dimensional space, random examples are generated from that separator, and a given percentage of the outputs are flipped. Additional features are generated randomly, with given percentages of independent features, dependent features, and repeated features. In our experiments, 100 such databases were generated. For each database, 800 examples were generated for training and 400 for testing; 238 additional features were generated. 10 % of the outputs were flipped randomly.

One "true" feature was among the top two ranked features 100 times, both "true" features were selected 74 times; the selected features were used as inputs of linear separators with sigmoid nonlinearity, which were trained by minimizing the usual least squares cost function with the Levenberg-Marquardt algorithm; for comparison, similar linear separators were trained with the "true" features, whenever the latter were not selected. The mean misclassification rate on the training sets was 10.4% (standard deviation 1.1%) with the selected features, whereas it was 10.1% (standard deviation 0.7%) with the "true" features. A t-test performed with 0.5% significance accepts the hypothesis that the difference between the means is smaller than or equal to 0.125 %, which is the smallest misclassification rate that can be detected (1 error out of 800). Thus, the performances of the true features and of the selected ones are not distinguishable with that level of significance, thereby proving that the probe feature method selects either the "true" features, or features that are essentially as good as the "true" ones for the problem under consideration, with the machine that was implemented. The classification rates on the test sets are not significantly different from those on the training sets (a t-test with 5% significance supports the null hypothesis that the mean misclassification rates are equal). For comparison, the misclassification rate when two features are chosen randomly is about 45%; it is about 30% when one "true" feature and one randomly chosen feature are used.

If a 1% risk is used,[1] the first three features of the list are selected. The resulting misclassification rates are not significantly different from the above results.

If 100 examples only are present in the training set, both "true" features are found in only 37 cases out of 100; however, the misclassification rates of classifiers trained with both true features did not differ significantly by more than 1% from the misclassification rates of classifiers trained with the two selected features.

6.1.2 VARIABLE SELECTION FOR THE XOR PROBLEM

In the same spirit, we build a database for classification with two classes of 50 examples each, drawn from identical Gaussian distributions whose centers are in XOR positions in 2-dimensional space. 50 additional candidate variables are generated from a uniform distribution in [-2, +2]. If feature selection is attempted with a linear model, the above procedure fails to give a satisfactory model, as expected, so that the result of the selection is not valid; the relevant inputs are ranked quite low. If variable selection is attempted with a quadratic model (leading to 1,326 different features, with 52 independent features including 2 relevant features), the random probe procedure selects the relevant variables, and no other, with 1% risk. If the regression is performed with the selected variables, the valid discriminant function $f = x_1 x_2$ is found, where x_1 and x_2 are the relevant variables.

In the present example, there is no point in trying to find a better solution by feeding the selected variables x_1 and x_2 to a nonlinear-in-its-parameters model: since the problem is 2-dimensional, a neural network would not provide a more parsimonious solution.

6.1.3 SELECTION OF INPUTS AND HIDDEN NEURONS IN A NEURAL NETWORK

A training set of 2,000 examples, and a test set of 2,000 examples, are generated by a neural network with 10 inputs, 5 hidden neurons with sigmoid activation function, and a linear output neuron. Its weights are drawn from a gaussian distribution (0, 0.1). The inputs are drawn from a Gaussian distribution with zero mean, whose standard deviation is computed so as to convey a given variance to the potential of the hidden neurons: the larger the variance of the potential, the more severe the nonlinearity. A zero-mean Gaussian noise is added to the output. Inputs are first selected as described in Section 3, and hidden neurons are selected as described in Section 5. Training is performed with the BFGS optimization algorithm, using gradient values computed by backpropagation. Table 1 shows the results obtained for 2 different standard deviations of the potential of the hidden neurons, and 5 different noise variances ranging from 10^{-10} to 1. In all cases, the selection method, starting from a candidate architecture with 20 candidate inputs and 10 hidden neurons, retrieves the correct

1. Experiments performed with the NeuroOne software package by Netral S.A.

Standard deviation of the potential	Standard deviation of the noise	Number of inputs of the final model	Number of hidden neurons	Root mean square training error	Root mean square test error
3	$1.\ 10^{-10}$	10	5	$9.6\ 10^{-11}$	$1.1\ 10^{-10}$
3	$1.\ 10^{-1}$	10	5	$9.8\ 10^{-2}$	$1.1\ 10^{-1}$
3	1	10	4	1.04	1.1
5	$1.\ 10^{-10}$	10	5	$9.7\ 10^{-11}$	$1.1\ 10^{-10}$
5	$1.\ 10^{-1}$	10	5	$1.0\ 10^{-1}$	$1.1\ 10^{-1}$
5	1	10	4	1.02	1.1

Table 1: Feature selection for the neural network problem, varying noise and number of hidden units.

architecture, except for high noise levels where a lower complexity is appropriate for explaining the measured output.

6.2 Industrial and Financial Applications

In this section, we describe briefly a number of real applications in which this method proved powerful, and was readily understood by field experts who were not familiar with statistical methods such as hypothesis testing.

The prediction of chemical properties of molecules (or QSAR – Quantitative Structure-Activity Relations), viewed as an aid to drug discovery, is a notoriously difficult problem (see for instance Hansch and Leo, 1995), because data is sparse, and because the candidate features are numerous. Both neural networks (see for instance Bodor et al., 1994) and support vector machines (Breneman et al., 2002) have been used extensively. The variable selection method presented here (together with an efficient machine selection method) allowed the prediction of the partition coefficient of a large number of molecules with previously unequalled accuracy on the same data sets (Duprat et al., 1998).

Spot welding is the most widely used welding process in the car industry. Two steel sheets are welded together by passing a current of a few kiloamperes between two electrodes pressed against the metal surfaces, typically for a hundred milliseconds. The heating thus produced melts a roughly cylindrical region of the metal sheets. After cooling, the diameter of the melted zone characterizes the effectiveness of the process; therefore, the spot diameter is a crucial element in the safety of a vehicle. At present, no fast, non-destructive method exists for measuring the spot diameter, so that there is no way of assessing the quality of the weld immediately after welding. Modelling the dynamics of the welding process from first principles is a difficult task, which cannot be performed in real time. These considerations led to considering black-box modelling for designing a "virtual sensor" of the spot diameter from electrical and mechanical measurements performed during welding. The main concerns for the modelling task were the choice of the model inputs, and the limited amount of examples available initially in the database. Variable selection (Monari, 1999) was performed both as described in the present paper, and by more classical methods (stepwise backward regression and statistical tests based on performance comparisons), with identical results. Our method is computationally less expensive than methods based of performance comparisons since performance comparisons between models with different inputs require (i) training several models with different initial values of the parameters (for nonlinear-in-the-parameters models), (ii) selecting the model with the smallest leave-one-out or cross-validation score for each set of candidate inputs, (iii) performing the test. The variables selected by the random probe method with a polynomial model of degree 3, were subsequently used as inputs to neural networks. The feature set was validated by the process experts. The selection of the prediction machine itself was performed on the basis on the computed leverages of the example, as described by Monari and Dreyfus (2002).

Still in the area of nondestructive testing, but in a completely different application, the feature selection method described here was implemented for the classification of electromagnetic signatures provided by eddy current sensors mounted a few millimeters above the rails, under carriages of the Paris subway. The purpose of the application is the automatic detection of rail defects. Fourier analysis yields 100 candidate features, while the number of examples was limited to 140, for a 4-class problem. The 4-class problem was split into 2-class subproblems, and feature selection was performed independently for each problem; the number of variables was thus reduced to less than 10 for each classifier (Oukhellou et al., 1998).

The present selection method was originally developed for two target applications in finance: the financial analysis of companies for investment purposes, and the financial analysis of town budgets. In the first case, experts suggested 45 financial ratios that were deemed relevant. The probe feature method reduced the number of features to 7, leading to a model that was more efficient and more clearly understandable than the previous ones; it has been in constant use for the last five years. In the second case, the modelling was a 5-class classification problem, which was split into 10 pairwise classification problems; variable selection was performed separately for each classifier. Using a 5% risk, the largest pairwise classifier had 10 variables. The classifier was applied to all 36,000 French towns for financial assessment. Both applications are described in detail by Stoppiglia (1997)

Finally, the present method proved particularly successful for information filtering. The purpose of information filtering is to find information that is relevant to a given topic (defined in a short sentence) in a wide corpus of texts. This can be formalized as a simple 2-class classification problem (a document is either relevant or irrelevant), but the selection of the variables of the classifier (related to the frequency of occurrence of words in the text to be classified) is difficult, since the vocabulary is virtually infinite. Furthermore, since isolated words tend to be ambiguous, the context must be considered, thereby making the structure of the classifier even more complex (see for instance Jing and Tzoukermann, 1999). Therefore, feature selection is crucial. Detailed comparisons between the present method, mutual information, statistical tests, and a selection method that is specific to automatic language processing, can be found in the study of Stricker (2000). The method presented here was used both to find the specific vocabulary of the topics and to find the relevant context of each word of the specific vocabulary. Experiments performed on very large corpuses (Reuters and Agence France-Presse corpuses, and other corpuses mentioned below) and large numbers of topics, showed that the specific vocabulary of a topic can be reduced to 25 words on the average, with an average of 3 context words per word of the specific vocabulary. Linear classifiers trained with regularization were found to be suitable after variable selection. Detailed descriptions of applications of the present selection method to information filtering can be found in the papers of Stricker et al. (1999), Wolinski et al. (2000).

Task	Number of examples	Number of candidate features	Number of selected features
QSAR (regression)	321	74	8
Spot welding (regression)	310	15	4
Eddy current signals (classification, 4 classes)	100	140	< 10
Financial analysis (classification, 5 classes)	250	45	7
Information filtering (classification, 2 classes)	1000 (typical)	400 (typical)	25 (typical)

Table 2: Summary of industrial and financial application results.

Table 2 summarizes results obtained in the above applications. For the last one, typical values are given, because thousands of different classifiers were designed in order to deal with the databases that were investigated.

7. Discussion and Conclusion

The probe feature method, as described in the present paper, contains two distinct ingredients: a method for ranking features (classical or modified Gram-Schmidt orthogonalization) and a method for selecting ranked features (the introduction of a probe feature among candidate features). Although they are presented together here, they deserve separate discussions.

The ranking of features through orthogonalization for linear-in-their-parameters models is by no means new. It has many interesting features. First, it is fast. Second, it takes into account the mutual information between features: if two features are almost collinear in observation space, the fact that one of them is selected will tend to drive the other to a much lower rank in the list. It has the additional advantage of allowing an incremental construction of the model, so that training can be terminated, without using all features, as soon as a satisfaction criterion is met; if the linear-in-the-parameters machine thus trained is expected to be satisfactory, the generalization ability of the machine, as estimated by a cross-validation or leave-one-out score, can be used as a satisfaction criterion. Conversely, if the features are intended for subsequent use as inputs of a different machine, it is only necessary to make sure that the linear-in-its-parameters machine can learn the task; in the affirmative, the selected variables or features thus selected can be used as inputs to a different machine that is not necessarily linear in its parameters. On the negative side, the method is based on the minimization of a squared error loss, which is not always the most appropriate for classification, even though it gives very good results, as shown above; its extension to other loss functions (such as cross-entropy for classification) is an open problem.

The idea of appending a random probe feature to the set of candidate features and ranking it among the others is central in the present paper. It is a powerful stopping criterion for Gram-Schmidt orthogonalization or any of its variants, because the cumulative distribution function of the rank of the probe can be computed analytically as proved above, so that one does not have to actually rank realizations of the probe. However, as shown by Stoppiglia (1997), it can be used in a different way: instead of *computing* the cumulative distribution function of the probe feature analytically, one can *estimate* it by generating a number of realizations of the probe feature, and ranking them among the others by whatever ranking method is preferred, thereby generating a corresponding number of realizations of the random rank of the probe, and allowing an estimation of its cumulative distribution function. This makes the probe method potentially of more general use, e.g. for selection methods that are based on weight elimination in the spirit of OBD (see for instance Reed, 1993); since weights are related to individual examples in SVM's, the method might also be useful for example selection (Guyon et al., 2002). In addition, the assumption of normality of the probe can be relaxed since it is necessary only for the analytical computation of the cumulative distribution function.

The selection method that is described in the present paper is intuitive and easily understandable, even by engineers who are not familiar with hypothesis testing; this is an attractive feature for researchers who endeavor to make machine learning techniques popular in industry. However, the method is not yet another heuristics for model selection, since it is firmly based on statistics. Furthermore, in contrast to Fisher's test - to which the probe technique is closely related - the assumption that the complete model actually contains the regression is not required. In contrast to the approach described by Weston et al. (2001) the probe feature method does not aim directly at improving the learning machine itself. It can only be conjectured that the withdrawal of irrelevant variables or features will help the machine perform better. The method proved powerful, in several contexts involving a large number of candidate features, and compared favorably, in terms of computation times, with classical tests.

Appendix A. Computation of the Cumulative Distribution Function

The cumulative distribution function is given by:

$$P_\nu(x) = \int_0^x \frac{\Gamma\left(\frac{\nu}{2}\right)}{\Gamma\left(\frac{1}{2}\right)\Gamma\left(\frac{\nu-1}{2}\right)} \frac{(1-u)^{(\nu-3)/2}}{\sqrt{u}} du$$

with $\nu \geq 2$ and $x = \cos^2\theta$.

If ν is even, then

$$P_\nu(x) = \frac{2}{\pi}\left[\sin^{-1}\sqrt{x} + \sqrt{x(1-x)}\Phi_{\nu/2-2}(x)\right]$$

where $\Phi_{\nu/2-2}$ is a polynomial of degree $\nu/2-2$,

$$\Phi_{\nu/2-2}(x) = 1 + \sum_{k=1}^{\nu/2-2} 2^k \frac{k!}{(2k+1)!!}(1-x)^k \quad \text{for} \quad \nu \geq 6$$

$\Phi_0(x) = 1$ for $\nu = 4$,
$\Phi_{-1}(x) = 0$ for $\nu = 2$.

If ν is odd, then

$$P_\nu(x) = \sqrt{x}\Psi_{(\nu-3)/2}(x)$$

where $\Psi_{(\nu-3)/2}$ is a polynomial of degree $(\nu-3)/2$,

$$\Psi_{(\nu-3)/2}(x) = 1 + \sum_{k=1}^{(\nu-3)/2} \frac{1}{2^k}\frac{(2k-1)!!}{k!}(1-x)^k \quad \text{for} \quad \nu \geq 5$$

$\Psi_0(x) = 1$ for $\nu = 3$

Appendix B. Relation of the Probe Feature Method to Fisher's Test

Fisher's test is a classical statistical (frequentist) approach to the selection of models that are linear with respect to their parameters. It is assumed that the process can be described by equation:

$$\boldsymbol{y}_p = X\theta + \omega$$

where ω is Gaussian distributed $(0, \sigma^2)$. Since $E(\omega) = 0$, it is assumed that the regression function belongs to the family of linear equations

$$\boldsymbol{y} = X\theta \tag{6}$$

within which the model is searched for (the model is said to be *complete*).

If one (or more) input is irrelevant, the corresponding parameter of the model should be equal to zero. Therefore, the hypothesis that is tested is the fact that one or more parameters are equal to zero. Assume that it is desired to test the validity of the complete model against that of a sub-model with q parameters equal to zero. The following quantities are defined

$\boldsymbol{y}_Q = X\theta_{LS}$ where θ_{LS} is the parameter vector obtained by least-squares fitting of the complete model (Q parameters) to the available data,

$\boldsymbol{y}_{Q-q} = X\theta_{LS}^q$ where θ_{LS}^q is the parameter vector obtained by least squares fitting of the complete model, under the constraint that q parameters out of $Q$are equal to zero

The considered hypotheses are

H_0: the q parameters are equal to zero,

H_1: the q parameters are not equal to zero.

If H_0 (the null hypothesis) is true, the random variable

$$R = \frac{N-Q-1}{q}\,\frac{\left\|\boldsymbol{y}_p - \boldsymbol{y}_{Q-q}\right\|^2 - \left\|\boldsymbol{y}_p - \boldsymbol{y}_Q\right\|^2}{\left\|\boldsymbol{y}_p - \boldsymbol{y}_Q\right\|^2} = \frac{N-Q-1}{q}\,\frac{\left\|\boldsymbol{y}_Q - \boldsymbol{y}_{Q-q}\right\|^2}{\left\|\boldsymbol{y}_p - \boldsymbol{y}_Q\right\|^2} \tag{7}$$

has a Fisher-Snedecor distribution with q and $(N-Q-1)$ degrees of freedom. If, with a given risk, the test leads to rejecting the null hypothesis, the sub-model with q parameters equal to zero is rejected.

Fisher's test compares a sub-model to the complete model. Other tests, such as the Likelihood Ratio Test (Goodwin and Payne, 1977) and the Logarithm Determinant Ratio Test (Leontaritis and Billings, 1987) compare models that are not thus related. It is proved in (Soederstroem, 1977) that these tests are asymptotically equivalent to Fisher's test.

In principle, the complete model (with Q parameters) should be compared, using Fisher's test, to all 2^Q sub-models. Using feature ranking with the Gram-Schmidt method as explained above, the number of comparisons can be reduced to Q.

Relation between the probe feature method and Fisher's test

In the previous section, it was proved that, at iteration n of the procedure, $\cos^2\varphi_n$ obeys a Beta distribution with a = 1/2 and $b = (N-n\text{-}1)/2$ (relation 2 with $\nu = N-n$). If a random variable X is distributed with a Beta law, then $\frac{b}{a}\frac{X}{1-X}$ obeys a Fisher law with $2a$ and $2b$ degrees of freedom. Therefore, the random variable

$$(N-n-1)\,\frac{\cos^2\varphi_n}{1-\cos^2\varphi_n} = \frac{N-n-1}{\tan^2\varphi_n}$$

obeys a Fisher law with 1 and $N - n$-1 degrees of freedom.

At iteration n of the procedure, a model with n-1 parameters is available. Assume that we want to perform Fisher's test to compare the n-parameter model obtained by adding the next parameter in the ranked list to the model with n-1 parameters (assuming that the complete model contains the regression). From relation (7), the random variable

$$R = \frac{N-n-1}{1} \frac{\left\|\boldsymbol{y}_p^{n-1} - \boldsymbol{y}_{n-1}\right\|^2 - \left\|\boldsymbol{y}_p^{n-1} - \boldsymbol{y}_n\right\|^2}{\left\|\boldsymbol{y}_p^{n-1} - \boldsymbol{y}_{n-1}\right\|^2} \tag{8}$$

should be a Fisher variable with 1 and $N - n$-1 degrees of freedom, where $\boldsymbol{y}_p^{n-1}$ is the projection of the output considered at iteration n, $\boldsymbol{y}_n$ and $\boldsymbol{y}_{n-1}$ are the outputs of the models with n and n-1 variables respectively. At iteration n, all vectors of interest are in a space of dimension $N - n$+1, and the least-squares solution of the model with n-1 parameters lies in the null space of that space, so that $\boldsymbol{y}_{n-1} = 0$. Moreover, $\boldsymbol{y}_n$ is the projection of $\boldsymbol{y}_p^{n-1}$ onto the direction of the selected feature, so that the angle between those vectors is φ_n.

Therefore,

$$\left\|\boldsymbol{y}_p^{n-1} - \boldsymbol{y}_n\right\|^2 = \left\|\boldsymbol{y}_p^{n-1}\right\|^2 \sin^2 \varphi_n$$

and

$$R = \frac{N-n-1}{\tan^2 \varphi_n}$$

Hence, the random variable that is used to discriminate between the null hypothesis and the alternative one can be derived from the probe feature method. The latter thus appears as an alternative to Fisher's test, which (i) gives the model designer a clear explanation as to why features should be discarded (given the available data) and (ii) does not rely on the assumption that the complete model actually contains the regression.

References

A. Bjoerck. Solving linear least squares problems by gram-schmidt orthogonalization. *Nordisk Tidshrift for Informationsbehadlung*, 7:1–21, 1967.

N. Bodor, M. J. Huang, and A. Harget. Neural network studies. 3. prediction of partition coefficients. *J. Mol. Struct. (Theochem.)*, 309:259–266, 1994.

C. Breneman, K. Bennett, M. Embrechts, S. Cramer, M. Song, and J. Bi. Descriptor generation, selection and model building in quantitative structure-property analysis. In J. Crawse, editor, *Experimental Design for Combinatorial and High Throughput Materials Development*. Wiley (to be publised), 2002.

S. Chen, S.A. Billings, and W. Luo. Orthogonal least squares methods and their application to non-linear system identification. *International Journal of Control*, 50:1873–1896, 1989.

S. Chen, F. Cowan, and P. Grant. Orthogonal least squares learning algorithm for radial basis function networks. *IEEE Transactions on Neural Networks*, 2:302–309, 1991.

J. de Lagarde. *Initiation à l'analyse des données*. Dunod, Paris, 1983.

A. Duprat, T. Huynh, and G. Dreyfus. Towards a principled methodology for neural network design and performance evaluation in qsar; application to the prediction of logp. *J. Chem. Inf. Comp. Sci.*, 38:586–594, 1998.

G. C. Goodwin and R.L. Payne. Dynamic system identification: Experiment design and data analysis. *Mathematics in Science and Engineering, Academic Press*, 136, 1977.

I. Guyon, J. Weston, S. Barnhill, and V. Vapnik. Gene selection for cancer classification using support vector machines. *Machine Learning*, 46:389–422, 2002.

C. Hansch and A. Leo. Exploring qsar, fundamentals and applications in chemistry and biology. *American Chemical Society*, 1995.

H. Jing and E. Tzoukermann. Information retrieval based on context distance and morphology. In *Proceedings of the 22nd Annual International Conference on Research and Development in Information Retrieval (SIGIR '99)*, pages 90–96, 1999.

I. J. Leontaritis and S. A. Billings. Model selection and validation methods for non-linear systems. *International Journal of Control*, 45:311–341, 1987.

G. Monari. *Sélection de modèles non linéaires par leave-one-out. Etude théorique et application au procédé de soudage par points*. PhD thesis, Université Pierre et Marie Curie, 1999.

G. Monari and G. Dreyfus. Withdrawing an example from the training set: an analytic estimation of its effect on a nonlinear parameterised model. *Neurocomputing*, 35:195–201, 2000.

G. Monari and G. Dreyfus. Local overfitting control via leverages. *Neural Computation*, 14(6): 1481–1506, 2002.

A. Mood, F. A. Graybill, and D. C. Boes. *Introduction to the Theory of Statistics*. MacGraw-Hill International, 1974.

R. H. Myers. *Classical and Modern Regression with Applications*. Duxbury Press, 1990.

L. Oukhellou, P. Aknin, H. Stoppiglia, and G. Dreyfus. A new decision criterion for feature selection: Application to the classification of non destructive testing signatures. In *European SIgnal Processing COnference (EUSIPCO'98)*, 1998.

Y. Oussar. *Réseaux d'ondelettes et réseaux de neurones pour la modélisation statique et dynamique de processus*. PhD thesis, Université Pierre et Marie Curie, 1998.

Y. Oussar and G. Dreyfus. Initialization by selection for wavelet network training. *Neurocomputing*, 34:131–143, 2000.

R. Reed. Pruning algorithms – a survey. *IEEE Transactions on Neural Networks*, 4:740–747, 1993.

T. Soederstroem. On model structure testing in system identification. *International Journal of Control*, 26:1–18, 1977.

H. Stoppiglia. *Méthodes Statistiques de Sélection de Modèles Neuronaux ; Applications Financières et Bancaires*. PhD thesis, Université Pierre et Marie Curie, Paris, 1997.

M. Stricker. *Réseaux de neurones pour le traitement automatique du langage : conception et réalisation de filtres d'informations*. PhD thesis, Université Pierre et Marie Curie, Paris, 2000.

M. Stricker, F. Vichot, G. Dreyfus, and F. Wolinski. Two-step feature selection and neural network classification for the trec-8 routing. In *Proceedings of the Eighth Text Retrieval Conference*, 1999.

D. Urbani, P. Roussel-Ragot, L. Personnaz, and G. Dreyfus. The selection of neural models of nonlinear dynamical systems by statistical tests. In J. Vlontzos, J.Hwang, and E. Wilson, editors, *Neural Networks for Signal Processing IV*, pages 229–237, 1993.

P. Vincent and Y. Bengio. Kernel matching pursuit. *Machine Learning*, 48:165–187, 2001.

J. Weston, S. Mukherjee, O. Chapelle, M. Pontil, T. Poggio, and V. Vapnik. Feature selection for svms. In *Neural Information Processing Systems 14*, 2001.

F. Wolinski, F. Vichot, and M. Stricker. Using learning-based filters to detect rule-based filtering obsolescence. In *Proceedings RIAO'2000*, 2000.