Feature Selection Methods for Multiphase Reactors Data Classification

Feature Selection Methods for Multiphase Reactors Data Classification

Laurentiu A. Tarca, Bernard P. A. Grandjean, and Faïçal Larachi*

Department of Chemical Engineering, Laval University, Québec G1K 7P4, Canada

The design of reliable data-driven classifiers able to predict flow regimes in trickle beds or bed initial behavior (contraction/expansion) in three-phase fluidized beds requires as a first step the identification of a restrained number of salient variables among all the numerous available features. Reduction of dimensionality of the feature space is urged by the fact that lesser training samples may be required and/or more reliable estimates for the classifier parameters may be achieved and/or improvement in accuracy can be achieved. This work investigates several methodologies to identify the relevant features in two classification problems belonging to a multiphase reactor context. Relevance of the subsets was assessed using mutual information between the subsets and the class variable (filter approach) and by the accuracy rate of a one-nearest neighbor classifier (wrapper approach). Algorithms for generating feasible sets to maximize these relevance criteria that were investigated were the sequential forward selection and the plus-*l*-take away *r*. Another conceptually different method to feature ranking that was tested was based on the Garson's saliency indices derived from the weights of classification neural networks. Reliability of the feature selection methodologies was first evaluated on two benchmark problems (a synthetic problem and the Anderson's iris data). They were henceforth applied to the two multiphase reactors classification problems with the goal of identifying the most appropriate features subsets to be used into classifiers. Finally, a new feature selection algorithm which combines filter and wrapper techniques proved to yield the same solutions as the wrapper technique while being less computationally expensive.

1. Introduction

Chemical reactor engineering, as many other fields of science and engineering, faces challenging tasks of knowledge extraction from data which increasingly become available and more accurate. The final goal for researchers and engineers is to anticipate and predict the behavior of such complex systems as multiphase reactors that still challenge the current physically based first-principles approaches.[1] One such problem often encountered is the ability to accurately identify the state a particular system belongs to, given some prior information that takes the form of measurable observations. When the state to be predicted takes the form of a *categorical* variable, the problem is known as a *classification* problem.

Despite several decades of research in the area of pattern recognition dealing with the general classification issue, yet the design of a *general purpose machine pattern recognizer* remains an actively pursued goal. As a matter of fact, provided enough data samples are available, the process of reaching this goal splits into two steps: *feature selection* followed by *classifier design*. The first refers to the task of selecting, among all the candidates, a limited number of features (or variables) which are the most relevant to the classification task. The second refers to the choice and the design of a particular inference engine (classifier) able to learn from data and make reliable prediction in new situations. Feature selection is important as it was shown to reduce classifier complexity and cost and improve model accuracy, and visualization and comprehensibility of induced concepts.

Suppose that n examples (or instances) ω_k, $k = 1...n$, represented by the input vector $\mathbf{x}_p(\omega_k) = (x_{k,1},x_{k,2},...,x_{k,p})$ and the label of class $y(\omega_k)$, are available. (Here $y = 1, 2,...,N_c$ are particular values of the generic class variable denoted Y, N_c being the number of classes). Using this data set, it is desirable to design a classifier able to assign the correct class label y for a new instance ω'. Prior to design of the classifier itself that does this task, a feature selection step is required as one does not know a priori which among the p available features are important in the classification. Selecting only a reduced number d of features among all p, $d < p$, is attractive because the classifier performance depends on the interrelationship between the sample size n used for learning, the number of features d, and the classifier complexity.[2] As discussed by Bishop,[3] the number of learning samples needs to be an exponential function of the feature dimension so that there is an obvious interest in keeping d the lowest possible.

Selection of a good feature *subset* may be of little interest when the number of training samples is large enough and representative (or equivalently when the class-conditional probability density functions are fully known). In this case, the probability of misclassification does not increase as the number of features increases.[4] In practice, however, it has been observed that added features might degrade the classifier performance when n is relatively small with respect to the number of features. This behavior, known as *peaking*, occurs when for a given sample size (n) supplementary features

* To whom correspondence should be addressed. Tel.: 418-656-3566. Fax: 418-656-5993. E-mail: faical.larachi@gch.ulaval.ca.

10.1021/ie049667a CCC: $30.25 © 2005 American Chemical Society
Published on Web 01/25/2005

increase the number of classifier parameters, thereby reducing classifier reliability.[2] In those instances, low-dimensional pattern representations are more advantageous in terms of cost and accuracy. Notwithstanding, excessive reduction in the number of features may alter classifier discrimination power and inflate inaccuracy.

The main issue in dimensionality reduction is dependent on the choice of a criterion function J. A commonly used criterion is the accuracy rate $AR = 1 - P_e$, where P_e is the prediction error.

Feature extraction and *feature selection* are the two main methodologies used in dimensionality reduction. The former refers to algorithms generating a reduced number of new features based on transformations of original features, e.g., principal component analysis. The variables that result usually lack physical sense and are thus not easily interpretable. Feature selection algorithms, on the contrary, target the best d-subset among the available p features without features alteration. For this reason, it will be this category of algorithms that will be explored in the multiphase reactor classification problems studied here.

There exist two generic approaches for feature selection, termed as *filter* and *wrapper* techniques.[5] Filter model, through statistical techniques, are indicative of the accuracy of potentially induced classifiers. They "filter out" irrelevant features before the induction process and are usually fast (absence of training). A common quality criterion in filter models is the Shannon's *mutual information*, $I(Y|\mathbf{X}_s)$, which measures the information contributed by $\mathbf{X}_s$ on the class variable Y.[6,7] In wrapper models, good subsets $\mathbf{X}_s$ are searched using the induction algorithm itself where the accuracy rate (AR), estimated by holdout, cross-validation, or bootstrap, is to be maximized. Here more CPU time is needed and the solution depends on the particular classifier.

Once an appropriate relevance criterion J is defined, the problem of feature selection can be formulated as follows: given a set of p features, select a subset of size d that maximizes J.

The simplest approach would be to examine all C_p^d possible combinations and choose that with the largest J value. But even for moderate p and d values, such an exhaustive search may become impractical. Most currently used methods evaluate only a fraction of them, providing speed, but with the cost of not guaranteeing optimality. A second simple method would be to select the best d individual features as an approximate solution to the feature selection problem. Using the mutual information criterion, Batitti[8] was able to select the inputs for a neural network classifier. Inter-feature mutual information was considered for selecting not only features informative about the class Y but also which are as independent as possible among each other.

Most current feature selection algorithms are in the class of sequential methods. The *sequential forward selection* (SFS) is a first example. With SFS the best single feature is first selected. Then an extra feature is added which, combined with the already selected features, maximizes criterion J. SFS in conjunction with the mutual information criterion was implemented by Sridhar et al.[9] to select inputs for neural networks in functions approximation. The main distinction with respect to previous works[8] was the possibility to identify jointly important features, though at the expense of a supplementary computation overhead.

The heuristic basis of most sequential feature selection algorithms is the assumption that the criterion J is monotonic; i.e., any change in feature set size (and therefore feature set information content) is positively correlated with the change in J. If this is true when J is the mutual information, some class reparability measures or others, it is not always the case with other criteria such as the accuracy rate AR (or its complement, the prediction error).[10] In this case, sequential selection methods with backtracking such as the "*plus-l-take-away-r*" (or (l,r) search) method[11] or its improved version, the *sequential floating forward selection* (SFFS),[10] are more suitable. These methods first enlarge the feature subset by l features using SFS and then delete r of them one by one. The one withdrawn at each step is that which causes the smallest decrease in J. Though computationally more demanding than SFS (because more combinations are being evaluated), such methods are more efficient in conjunction with nonmonotonic criteria.[10]

The single "optimal" technique (optimal only if J is monotonic) is based on the Branch and Bound algorithm,[12] which avoids exhaustive search by using intermediate results for obtaining bounds on the final criterion value. Requiring to evaluate a number of possibilities that is still an exponential function of the number of variables (p) and because the monotonicity criterion does not hold here, the Branch and Bound was not considered in this study.

Other features search methods, such as those based on genetic algorithms (GA), were proposed for classification[13] or for the identification of neural net inputs in functions approximation.[14]

An alternative method for determining the most important features for a classification task is the method of Garson[15] of interpreting the weights of neural networks. In this method, a feed-forward neural network is trained to learn the mapping $Y(\mathbf{X}_s)$. Then a saliency index for each input of the network is computed. This index is calculated by assessing the share of weights associated with each one of them. This method was experimentally evaluated by Nath et al.,[16] concluding that the method has potential merit.

2. Study's Objective

The objective of this present work was to examine the extension of feature selection algorithms in two classification problems relevant to the field of multiphase reactor engineering, namely, flow regime assignment in trickle-bed reactors (LIR, TR, and HIR), and identification of bed initial expansion/contraction (IBE, IBC) in three-phase fluidized-bed reactors. The ability of these methods to provide good quality solutions (Elite subsets $\mathbf{X}_s$) and being in agreement with each other is investigated prior to two benchmark problems: a synthetic problem and the Anderson's iris data classification problem. For these two cases a priori knowledge on the relevance of the features is available. Furthermore, a new feature selection algorithm is devised in this work which mixes filter and wrapper algorithms.

Table 1 summarizes the four methods (M-I through M-IV) to be tested in this work, to identify the subset $\mathbf{X}_s$ (of size d) to be used in classification instead of the whole feature subset $\mathbf{X}_p$. The method of Garson,[15] referred to as the M-V method, is not shown there because it shares nothing in common with this clas-

Table 1. Feature Selection Strategies

	criterion to maximize (J)		
selection method	mutual information (information theory) $J = I(Y\|\mathbf{X}_s)$	accuracy rate of a 1-NN classifier *AR(1-NN)*	$I(Y\|\mathbf{X}_s)$ and *AR(1-NN)*
sequential forward selection (SFS)	yes (M-I) (filter method)	yes (M-II) (wrapper method)	yes (M-IV) SFS with $I(Y\|\mathbf{X}_s)$ continued by (l,r) search with *AR(1-NN)* (filter-wrapper method)
plus-l-take-away-r	no (not necessarily justified as J is monotonic)	yes (M-III) (wrapper method)	

sification of methods M-I to M-IV, except perhaps the fact that it needs training of the classifier (common to filters). A "Yes" (respectively, "No") entry in Table 1 means that the selection algorithm specified by the row header in conjunction with the criterion specified by the column header will (respectively, not) be tested.

The classifier used to assess the importance of sets in methods M-II to M-IV is the one-nearest neighbor (1-NN) classifier. Its performance is evaluated by 5-fold cross validation. Inherent details will be unfolded in the following sections.

A further extension of this work would be to build simple reliable and interpretable classifiers using as input variables the solutions $\mathbf{X}_s$ found within this study. A new brand of design tools may be therefore provided[17] as an alternative to the existing first-principle based models that are still unsatisfactory.

3. Relevance Assessment

What determines whether features are to be retained or dropped by a feature selection algorithm depends on their *relevance*. There are many definitions of relevance, each addressing from a particular point of view the question "relevant to what?".[5] Here, the relevance of a feature subset $\mathbf{X}_s$ in predicting the class variable Y will be judged using three measures:

(a) Mutual information $I(Y|\mathbf{X}_s)$ between the feature vector $\mathbf{X}_s$ and the class variable to be predicted Y;

(b) Accuracy rate of a 1-NN classifier *AR(1-NN)* that uses $\mathbf{X}_s$ as discriminatory features;

(c) The saliency index of Garson.[15]

Note that the (a) and (b) relevance measures, in conjunction with a selection algorithm such as SFS or (l,r) search, identify which d-subset $\mathbf{X}_s$ to use for predicting class membership without too much a sacrifice with respect to a discrimination involving all p features. The Garson's saliency index,[15] on the other hand, judges only on how relevant each feature is if the whole $\mathbf{X}_p$ set is used as a neural net input. This mostly provides a relevance order for all features in $\mathbf{X}_p$, i.e., features ranking rather than indications on which $\mathbf{X}_s$ subset is the most pertinent for classification.

3.1. Mutual Information. Recall first that the records ω_k have an input vector from which are selected a features subset $\mathbf{x}_s(\omega_k) = (x_{k,1},x_{k,2},...,x_{k,d})$ of cardinality d, and a (class membership) label $y(\omega_k)$. A classifier that uses $\mathbf{X}_s$ to predict the class Y decreases the initial *uncertainty* about the class Y by using the *information* in the features of $\mathbf{X}_s$. Because insufficient input information or suboptimal operation of the classifier, the uncertainty about the class cannot be decreased to zero. Shannon's information theory[6,7] gives a suitable formalism to quantify these concepts.

If the probability that the class variable Y takes a particular value y is denoted with $P(y)$, $y = 1,2,...,N_c$, the initial uncertainty in the output class variable is given by the *entropy*:

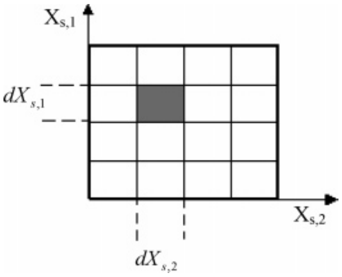

Figure 1. Bins construction: Suppose a 2-feature vector $\mathbf{X}_s$. Here a bin is the volume $dV = dX_{s,1} \times dX_{s,2}$.

$$H(Y) = -\sum_{y=1}^{N_c} P(y)\cdot\log P(y) \qquad (1)$$

Practically, the probability $P(y)$ can be estimated using the occurrence frequency of y

$$P(y) = \frac{n_y}{n} \qquad (2)$$

where n_y is the number of occurrences of y and n the total number of samples.

The entropy of the features vector $\mathbf{X}_s$ can also be similarly estimated. Since features are continuous variables, the sum is replaced by an integral:

$$H(\mathbf{X}_s) = -\int P(\mathbf{X}_s)\cdot\log P(\mathbf{X}_s)\, d\mathbf{X}_s \qquad (3)$$

The simplest way to estimate $P(\mathbf{X}_s)$ is by the use of histograms. First, each feature X_s, among all d constituting $\mathbf{X}_s$, is discretized into a large number of intervals, *nbx*. For simplicity, the same number of intervals is used for each feature. The hypercubes with the volume $dV = dX_{s,1} \times dX_{s,2}... \times dX_{s,d}$ are called bins. Bins construction is exemplified in Figure 1 for a 2-D set of features.

Consider now each of the nbx^d bins, and count how many samples, among all n, fall into each bin. For all bins, $b = 1...nbx^d$ probabilities $P(\mathbf{X}_{s\subset b}) = (n_b/n)$ (n_b = number of samples falling in bin b) are evaluated for $\mathbf{X}_s$ to occur in a particular bin b. The entropy $H(\mathbf{X}_s)$ is then computed using a discretized form of eq 3:

$$H(\mathbf{X}_s) = -\sum_{b=1}^{nbx^d} P(\mathbf{X}_{s\subset b})\cdot\log P(\mathbf{X}_{s\subset b}) \qquad (4)$$

The average uncertainty on Y after knowing the feature vector $\mathbf{X}_s$ with d components is the *conditional* entropy $H(Y|\mathbf{X}_s)$

$$H(Y|\mathbf{X}_s) = H(\mathbf{X}_s,Y) - H(\mathbf{X}_s) \qquad (5)$$

where $H(\mathbf{X}_s,Y)$ is the *joint entropy* estimated using a similar box counting procedure:

$$H(\mathbf{X}_s,Y) = -\sum_{b=1}^{nbx^d}\sum_{y=1}^{N_c} P(\mathbf{X}_{s\subset b},y)\cdot\log P(\mathbf{x}_{s\subset b},y) \quad (6)$$

in which $P(\mathbf{X}_{s\subset b},y)$ is the joint probability that $\mathbf{X}_s$ belongs to bin b and Y takes the value y.

By definition, the amount by which the uncertainty is decreased is the mutual information $I(Y|\mathbf{X}_s)$ between variables Y and $\mathbf{X}_s$:[8]

$$I(Y|\mathbf{X}_s) = H(Y) - H(Y|\mathbf{X}_s) \quad (7)$$

This function, symmetric with respect to Y and $\mathbf{X}_s$, can be reduced to

$$I(Y|\mathbf{X}_s) = I(\mathbf{X}_s|Y) = \sum_{y=1}^{N_c}\int P(Y,\mathbf{X}_s)\log\frac{P(Y,\mathbf{X}_s)}{P(Y)\cdot P(\mathbf{X}_s)}d\mathbf{X}_s \quad (8)$$

The uncertainty $H(Y,\mathbf{X}_s)$ in the combined events $(Y,\mathbf{X}_s)$ is usually less than the sum of the individual uncertainties $H(Y)$ and $H(\mathbf{X}_s)$. Using eqs 7 and 5, one obtains a symmetric function:

$$I(Y|\mathbf{X}_s) = H(Y) + H(\mathbf{X}_s) - H(Y,\mathbf{X}_s) \quad (9)$$

In function approximations, it was suggested[9] to derive an *asymmetric dependency coefficient* by dividing eq 9 rhs by $H(Y)$. This normalization has been adopted here. In these circumstances $I(Y|\mathbf{X}_s) = 0$ means that X_s contains no useful information about the class Y, whereas $I(Y|\mathbf{X}_s) = 1$ means that Y is completely predictable if X_s is known. In practice, however, the value of $I(Y|\mathbf{X}_s)$ is likely to depend also on grid coarseness (or number of bins). Coarser grids are suspected to inflate inaccuracy of $I(Y|\mathbf{X}_s)$ as important functional variations might be overlooked, while finer grids tend to overestimate $I(Y|\mathbf{X}_s)$ by counting noise as meaningful functional variation.

3.2. 1-NN Classifier Accuracy Rate. The nearest neighbor classifier is one of the simplest methods to perform nonparametric general-purpose classification. Proven to give good results on many pattern recognition problems,[2] it can be represented by the following decision rule: *assign a new pattern to the class of its nearest example in the training set as measured by a metric* (usually Euclidian) *distance*. The Euclidian distance between two points **a** and **b** is simply the following:

$$d_E(\mathbf{a},\mathbf{b}) = ||\mathbf{a} - \mathbf{b}|| = \sqrt{\sum_{i=1}^{d}(a_i - b_i)^2} \quad (10)$$

As it requires no training, the nearest neighbor classifier was used in this study to assess the capability of a subset $\mathbf{X}_s$ drawn from the larger $\mathbf{X}_p$ set to predict the class Y.

One method to estimate the accuracy rate AR of a classifier is to compute its confusion matrix on several z-fold cross-validation sets (see Kohavi[18] for cross-validation issues). Consider a set A of records ω_k, $k = 1, ...n$, available for the classification task. Each record ω_k is represented by the feature vector $\mathbf{x}_{s(\omega_k)}$ and label $y(\omega_k)$. Let set A be partitioned in say $z = 5$-folds. Each fold, once each time, is set aside as a *test* set while the ensemble of remaining $z - 1$ sets are taken as *training* set. For each test set, the confusion matrix of size $(N_c \times N_c)$ synthesizes the predictions of the classifier in the form

$$\begin{array}{cc} & \text{Pr}\,edicted \\ Actual & \begin{pmatrix} 80 & 10 & 10 \\ 10 & 75 & 15 \\ 5 & 5 & 90 \end{pmatrix}. \end{array}$$

For illustration here, a three-class classification problem was considered in which there are 100 samples in each class in the test set. Each cell of the confusion matrix indicates how many of the 100 samples were assigned to class y labeled column-wise when the actual (true) class index was the one indicated row-wise. Note that the sum over each row is equal to 100. For convenience, each element in a row of the confusion matrix can be normalized by the number of samples in the class indicated by the row index.

As there are z test sets, we may compute the mean confusion matrix over the z sets, and the standard deviation as well. The elements lying along the main diagonal of the confusion matrix give the per-class accuracy rate, whereas their averaged value gives the global accuracy rate AR of the classifier.

3.3. Garson's Saliency Index. The third criterion to be implemented is the saliency index[15] S_{ind}, which determines the importance of the candidate features by interpreting the weights of feed-forward neural networks (Figure 2) trained with back-propagation algorithm.[19] The Garson method[15] determines which among the input nodes (and thus features) is responsible for most of the changes undergone by the output.

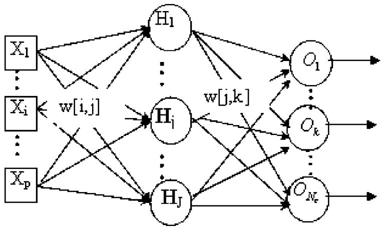

Figure 2. Feed-forward neural network used for classification involving N_c classes.

All the features in the set $\mathbf{X}_p$ are fed into the network by means of the input nodes which preprocess the data by normalizing it to the [0,1] interval. The J hidden units perform nonlinear projection of the feature space and the output layer decides into which class a particular input point is assigned. The number of output nodes equals the number of classes in which the patterns are to be classified. For a 2-class problem, a single network output suffices to do the task. The parameters $w[i,j]$ and $w[j,k]$ on the interlayer connections are the network weights. Network training is meant to optimize such weights in a way that when a particular record ω, belonging to the class k, is presented as input, the network output vector $[O_1,O_2,...,O_{Nc}]$ shows (nearly) zero elements except for the kth component which will be 1 (or near to).

Saliency $S_{ind}(i,k)$ of input i with respect to output k is estimated as follows: First each hidden-to-output weight $w[j,k]$ is incorporated into the input-to-hidden weights $w[i,j]$. Then for each input variable a summation is made over all hidden units and converted into a percentage of the total for all input nodes,

$$S_{ind}(i,k) = \frac{\sum_{j=1}^{J}\left(\frac{|w[i,j]|}{\sum_{i=1}^{p}|w[i,j]|} \times |w[j,k]|\right)}{\sum_{i=1}^{p}\sum_{j=1}^{J}\left(\frac{|w[i,j]|}{\sum_{i=1}^{p}|w[i,j]|} \times |w[j,k]|\right)} \qquad (11)$$

where $i = 1,2,...,p$ input variables; $j = 1,2,...,J$ sweeps the hidden nodes and | | means absolute value.

In Garson's work,[14] however, situations in which there are more than two classes (so more than one output node) were not treated.

We proposed therefore to generalize eq 11 to cases where there are N_c classes. We compute therefore, an overall saliency $S_{ind}(i)$ for each input node and multiple-output neural networks ($k = 1,2,...,N_c$) as

$$S_{ind}(i) = 1/N_c \cdot \sum_{k=1}^{N_c} S_{ind}(i,k) \qquad (12)$$

The logic behind eq 12 is that the importance of an input variable should be judged by its impact on the whole output layer.

4. Feature Selection Methods

The three relevance criteria being described above let us now turn to present the methods for solving the following feature selection problem:

Given a set $\mathbf{X}_p$ of p features, select subset $\mathbf{X}_s$ of size d, d ≤ p, that maximizes the relevance criterion J

Among the several techniques introduced in section 1, we shall focus only on the class of sequential methods, i.e., sequential forward selection (SFS), sequential backward selection (SBS), and *(l,r)* method that were effectively implemented in this work.

SFS and SBS are step-optimal because only the best (respectively, the worst) feature is always added (respectively, deleted). A limitation inherent to these two methods is their inability with the nested feature combinations to correct decisions in later steps, achieving therefore suboptimal subsets with lower J. Stearns[11] combined SFS and SBS to avoid the nesting effect, bringing thus a net improvement to the sequential methods. This combined method, referred to as the *(l,r)* method, repeatedly applies SFS l times followed by r SBS steps until the required number of features is reached. The *(l,r)*-search algorithm, or its particular cases (1,0)-search (or SFS) and (0,1)-search (or SBS), as described by Pudil et al.[10] is detailed in Appendix A.

The termination criterion (Appendix A) is based on a priori knowledge of d, i.e., size of the best subset. As in our problems p is not so large termination is set for $d = p$. The best $\mathbf{X}_{s,d}$ subset is retained according to either modality: (a) mutual information based criterion J: d is chosen which yields $1 - J(\mathbf{X}_{s,d}) \leq \zeta$, with ζ a very close to zero threshold value; (b) *AR (1-NN)* based criterion J: d is chosen as $d = \arg\max_i J(\mathbf{X}_{s,i})$.

The *(l,r)* method was improved further by automatically tuning l and r values in the work of Pudil et al.[10] Their so-called sequential floating forward selection (SFFS) consists of applying after each forward step a number of backward steps as long as the resulting subsets are better than the previously evaluated ones for the same size d.

Using as relevance criterion the mutual information, SFS holds promise as $I(Y|\mathbf{X}_s)$ will always increase through adding a supplementary feature. This relevance criterion combined with SFS was already used to identify neural net inputs[9] but not in classification problems as the current work deals with. Earlier work sketching the use of $I(Y|\mathbf{X}_s)$ to select inputs for neural network classifiers used the mutual information between the individual features $X_{s,i}$ and the class Y, to assess the relevance of the whole subset $\mathbf{X}_s$.[8] However, that approach does not promote jointly relevant variables as we do here.

Conversely, if *AR(1-NN)*, which is a nonmonotonic criterion, is used as a relevance measure, the *(l,r)* search is more appropriate since it is able to reconsider previous decisions.

The use of $I(Y|\mathbf{X}_s)$ enables one to find the subset $\mathbf{X}_s$ that gives about the same information about the class Y as the entire set $\mathbf{X}_p$. But $I(Y|\mathbf{X}_s)$ is a statistical measure of the *general dependency* between $\mathbf{X}_s$ and Y, and not the classifier accuracy itself. Therefore, it does not guarantee that the resulting $\mathbf{X}_s$ will also have the best accuracy rate of the 1-NN classifier. However, mutual information filter criterion is faster in execution than a classifier training session because it does not require iterative computation on the data set as classifiers generally do.

As $I(Y|\mathbf{X}_s)$ evaluates the intrinsic properties of the data, rather than interactions with a particular classifier, the result is supposed to exhibit more generality; i.e., the solution will be "good" for several types of classifiers. On the contrary, using *AR(1-NN)* relevance criterion is more computationally intensive and is incumbent on the capability of this particular classifier to exploit those features in the classification task. However, as the accuracy is evaluated on test data, wrapper criteria possess a mechanism to avoid overfitting, and generally achieve better recognition rates than filters since they are tuned to the specific interactions between the classifier and the data set.

Hence, the M-IV method (Table 1) is devised in this work as an alternative. It first selects an initial set of predictive features $\mathbf{X}_{s,initial}$ identified by SFS in conjunction with $I(Y|\mathbf{X}_s)$, and then it grows up and prunes down this set with a *(l,r)* search in conjunction with *AR(1-NN)*. First of all, this method is faster than launching the *(l,r)* search in conjunction with *AR(1-NN)* starting with an empty set. This is because $I(Y|\mathbf{X}_s)$ is easier to compute than *AR(1-NN)* and because SFS is more rapid than the *(l,r)* search. Further adjustments of the resulting selection $\mathbf{X}_{s,initial}$ allow deletion (or addition) of some features which appear to be less (or more) important for the nearest neighbor classification rule.

5. Problems and Data Sets Description

Now that the description of the selection algorithms and relevance criteria is completed, the four problems

Table 2. Candidate Features in a Synthetic Problem

var. no.	mean for class c_1	mean for class c_2	mean for class c_3	relevance rank
1	3	0	−3	1
2	4	0	0	2
3	0	0	4	2
4	1	0	0	3
5	0	0	1	3
6	1	0	1	3
7	0	0	0	4-irrelevant
8	0	0	0	4-irrelevant
9	0	0	0	4-irrelevant
10	0	0	0	4-irrelevant

Table 3. Candidate Features for the Bed Iris Data Classification

no.	variable name	symbol	max	min
1	sepal length	SL	7.9	4.3
2	sepal width	SW	4.4	2
3	petal length	PL	6.9	1
4	petal width	PW	2.5	0.1

to be tested will be presented next to (i) compare the feature selection methods (M-I to M-IV) and judge their efficiency to identify the relevant features and (ii) identify the most predictive variables in a couple of actual multiphase reactor problems, namely, flow regime classification in trickle beds and initial bed expansion/contraction classification in three-phase fluidized beds.

5.1. Synthetic Problem. A synthetic domain problem is built in which the correct answer is known a priori. The setup is as follows: Generate three sets of one hundred 10-dimensional data points. In each set, the 10 variables are random normally distributed. The mean for each feature in each class (c_1 to c_3) was set as shown in Table 2.

From this setup, the single most relevant variable is 1, as the average interclass distance between the central points of the normal distributions is the largest for this variable. The most important two features are 1 and 3 (according to the same class separability measure), and the best three features are {1,3,2}. The variables 4, 5, and 6 are equally important to the classification task but are far less important than the previous ones. Features 7−10 are irrelevant to the classification task as they are normally distributed random variables with the same 0 central value for each class. This problem was considered only for benchmarking purposes and we are expecting that the features {1,3,2} will successfully be identified by the feature selection methods M-I to M-V. A similar setup was used by Nath et al.[16] to evaluate the efficacy of Garson's method[15] to rank features in a classification problem. However the setup chosen here is more realistic as we introduce redundant features in the set and the overlapping between classes is higher.

5.2. Anderson's Iris Data. Anderson's iris data is an open source collection of 150 4-dimensional data points, each falling into one of the three classes: Setossa (Se), Versicolor (Ve), and Virginica (Vi). In each class there are 50 data points. The independent variables considered as candidates are shown in Table 3.

The class Se is easily separable from the other two, which are overlapping classes. Features 3 and 4 can jointly play the role of features 1 and 2 as shown by Li et al.[20] Using a neuro-fuzzy classifier, they were able to classify data and reveal the important features in the same time. For this classifier only the features {3,4} solely brought the same prediction accuracy as all the variables 1 to 4. This does not imply that this is the unique best subset of features for all other types of classifiers. Batitti[8] for example showed that subset {1, 3} (or {1, 4}) is the best in terms of information content and also best for a multilayer feed-forward neural network classifier. It is expected thus that the feature selection methods to be tested will point to one of the two answers {3,4}, {1,3}, or {1,4}.

Table 4. Candidate Features for Flow Regime Class Prediction in Trickle Beds

no.	variable name	symbol	max	min
1	liquid superficial velocity (m/s)	u_L	1.74×10^{-1}	4.36×10^{-4}
2	gas superficia velocity (m/s)	u_G	3.74	4.98×10^{-4}
3	foaming property	foam.[a]	1	0
4	column diameter (m)	D_c	5.10×10^{-1}	2.30×10^{-2}
5	bed porosity	e	7.40×10^{-1}	2.63×10^{-1}
6	grain specific area (m^{-1})	a_G	5.16×10^{3}	4.67×10^{2}
7	bed specific area (m^{-1})	a_T	3.81×10^{3}	2.78×10^{2}
8	effective particle diameter (m)	d_p	1.28×10^{-2}	1.16×10^{-3}
9	sphericity	ϕ	1.00	3.36×10^{-1}
10	liquid density (kg/m^3)	ρ_L	1.18×10^{3}	6.50×10^{2}
11	liquid viscosity (Pa•s)	μ_L	6.63×10^{-2}	3.10×10^{-4}
12	surface tension (N/m)	σ_L	7.62×10^{-2}	1.90×10^{-2}
13	gas density (kg/m^3)	ρ_G	1.16×10^{2}	1.60×10^{-1}
14	gas viscosity (Pa•s)	μ_G	1.97×10^{-5}	1.45×10^{-5}

[a] Foaming property is a categorical variable. 0 = coalescing, 1 = foaming.

5.3. Three-Class Flow Regimes Classification in Trickle Beds. The first real classification problem in the realm of multiphase reactors, for which we search the most pertinent features, concerns a trickle-bed reactor in which the gas (G) and liquid (L) are flowing co-currently downward throughout a bed of catalytic solid (S) particles. The efficiency of such a device is highly dependent on the flow regime that prevails in the reactor for a given set of operational conditions.[1] Depending on the level of interaction between fluids, one may generally distinguish three flow regimes, namely: low interaction regime (LIR), transitional regime (TR), and high interaction regime (HIR). The flow regime is therefore the class variable Y which takes particular values $y = 1,2,...,N_c$. Here, $y = 1$ for LIR, $y = 2$ for TR, and $y = 3$ for HIR while $N_c = 3$. The type of flow regime is susceptible to be determined by the available p features that characterize the three phases (G−L−S), e.g., porosity, sphericity, gas density, liquid viscosity, fluids' superficial velocity, etc., which are contained in $\mathbf{X}_p = (X_1,X_2,...,X_p)$.

The Laval University comprehensive trickle bed reactor database[21] was interrogated to extract 2809 flow regime data points (945 LIR, 945 TR, and 919 HIR). The independent variables considered as candidates are summarized in Table 4 along with the span over which the measurements were available. In the working database, the variables were beforehand normalized to fall between 0 and 1. For variables covering more than 2 decades log values were used. The classes to be predicted are low interaction regime (LIR), transition regime (TR), and high interaction regime (HIR).

5.4. Two-Class Bed Expansion/Contraction in Three Phase Fluidized Beds. The second problem refers to the initial bed expansion (IBE) or contraction (IBC) in a gas−liquid−solid fluidized bed. This phenomenon occurs upon introduction of a tiny gas flow rate

Table 5. Candidate Features for the Bed Contraction−Expansion in Fluidized Beds

no.	variable name	symbol	max	min
1	liquid velocity (m/s)	u_L	2.60×10^{-1}	1.09×10^{-3}
2	liquid density (kg/m^3)	ρ_L	1.62×10^3	7.78×10^2
3	liquid viscosity (Pa·s)	μ_L	7.19×10^{-2}	7.16×10^{-4}
4	surface tension (N/m)	σ_L	7.59×10^{-2}	2.48×10^{-2}
5	solid density (kg/m^3)	ρ_s	2.90×10^3	1.07×10^3
6	effective particle diameter (m)	d_p	1.54×10^{-2}	6.50×10^{-4}
7	terminal velocity (m/s)	u_t	7.84×10^{-1}	4.32×10^{-2}
8	column diameter (m)	D_c	2.18×10^{-1}	4.00×10^{-2}
9	bed height at rest (m)	H_0	6.00	5.08×10^{-2}
10	foaming property	foam.[a]	1	0

[a] Foaming property is a categorical variable. 0 = coalescing, 1 = foaming.

in a liquid fluidized bed and is an important indicator of the bubble wake activity and bubble size. Large-sized bubbles are correspondingly associated with large wakes that suck liquid into their structures, thereby inducing liquid starvation in the emulsion phase so the bed contracts. On the contrary, small-size bubbles are correspondingly associated with small (or no) wakes that barely affect the emulsion liquid so the bed smoothly expands with further increasing gas throughputs. Similarly, the class variable Y in this case refers to IBE (y = 1) or IBC (y = 2) may depend on the p features characterizing the three phases (G−L−S) grouped in input vectors $\mathbf{X}_p$ for which we have measurements available.

In this example it is hoped that the features that allow discrimination of whether *initial* expansion or contraction of the bed will occur when a tiny gas stream is introduced into the liquid fluidized bed reactor will be identified. The porosity data set is extracted from the Laval University three-phase fluidization database[22] after analyzing the behavior of several porosity series at constant liquid velocities u_L and increasing gas velocities u_G. From each such series, the observation corresponding to the smallest u_G is retained. The class is considered expansion (IBE) if bed porosity increases with respect to that in the initial liquid fluidized state, or conversely contraction (IBC) if bed porosity decreases with respect to that in the initial liquid fluidized state. As expansion data points were about three times the contraction ones, two replicates of the contraction points were made to keep about the same number of samples for each class. The data set counted 402 contraction points and 414 expansion points and the candidate features are summarized in Table 5.

6. Results

In this section, we discuss results obtained by applying the feature selection methods M-I to M-IV and feature ranking method M-V to the four above-described problems. For each separate problem, we provide the solutions found by each method. While methods M-I to M-IV identify which subset $\mathbf{X}_s$ produces about the same discrimination power as the whole set $\mathbf{X}_p$, method M-V provides only a ranking of variables in the set $\mathbf{X}_p$.

6.1. Synthetic Problem. As explained in section 5.1, the features subset {1,3,2} is expected to be identified. First, SFS with mutual information as a relevance criterion (method M-I) is used. Figure 3 shows the result obtained by applying this filter method to the synthetic problem. The number of divisions in the domain for each feature was nbx = 10. The "+"sign before the feature label indicates that the feature was added to the combination at the corresponding epoch. Similarly, the corresponding J value was the one contributed only by the persistent features involved.

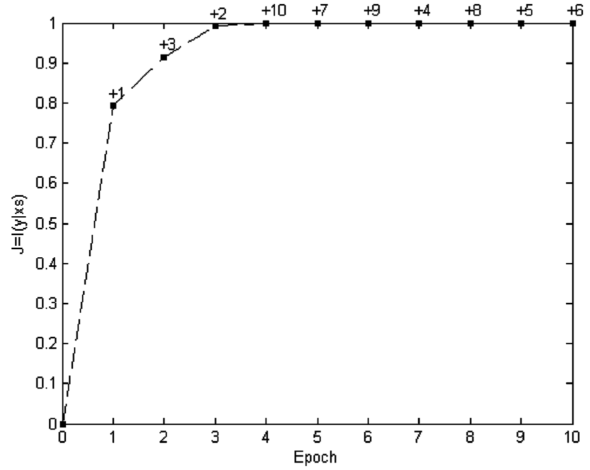

Figure 3. Sequential forward selection with mutual information as relevance criterion (M-I). "+" means that the corresponding feature was added to the current subset.

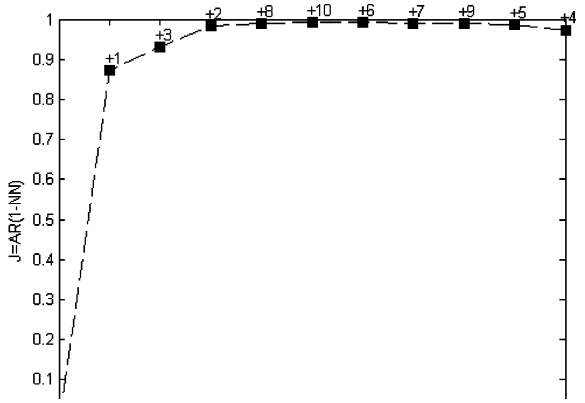

Figure 4. Sequential forward selection with accuracy rate as relevance criterion (M-II). "+" means that the corresponding feature was added to current subset.

As expected, the first selected features are {1,3,2}, which contain about all of the information available in the set of features $\mathbf{X}_p$={1,2,...,10}. Once feature 2 is added to the combination, further enlargement of the feature set brings no more significant increase in the relevance criterion J.

Applying SFS with the alternate relevance criterion *AR(1-NN)* (method M-II) induces the same order of preference for the first three variables, features set {1,3,2}, see Figure 4. After the 3rd variable is added, the accuracy rate does not increase significantly; rather, it starts decreasing after the 6th epoch.

Method M-III that consists of the (l,r) search with *AR(1-NN)* as the relevance criterion is also tested. In this work l = 2 and r = 1 were chosen as they require minimum computational effort. At each step, two features are added and one is removed. The suggested selection subset $\mathbf{X}_s$ is also {1,3,2}. This example is therefore too simple to show any difference in the searching power between SFS and (l,r) search. The filter/wrapper approach (method M-IV) gives the same solution as M-III but of course with less computational effort.

The saliency index values calculated with eq 12 (method M-V) are shown in Figure 5. Here the mean

Table 6. Summary of Methods M-I to M-V for the Synthetic Problem

selection strategy	no. of var.	order[a]	AR(1-NN) (%)
none (all available features considered)	10	NA	97.33
M-I: forward selection with $I(Y\|\mathbf{X}_s)$. ($nbx = 10$)	3	**1 3 2** (4 5 6 7 8 9 10)	98.33
M-II: forward selection with AR(1-NN).	3	**1 3 2** (4 5 6 7 8 9 10)	98.33
M-III: (l,r) search with AR(1-NN).	3	**1 3 2** (4 5 6 7 8 9 10)	98.33
M-IV: forward with $I(Y\|\mathbf{X}_s)$ continued with (l,r) search with AR(1-NN).	3	**1 3 2** (4 5 6 7 8 9 10)	98.33
M-V: Garson's saliency values through ANN	NA	(1 2 3) (4 5 6 7 8 9 10)	NA

[a] Parentheses here denote that the features inside cannot be confidently ranked. NA means not available.

Table 7. Methods M-I to M-V Compared on the Iris Data Classification Problem

method	no. of var.	order[a]	AR(1-NN) (%)
none (all available features considered)	4	NA	93.52
M-I: forward selection with $I(Y\|\mathbf{X}_s)$. ($nbx = 20$)	2	**4 3** (1 2)	94.12
M-II: forward selection with AR(1-NN).	2	**4 3** (1 2)	94.12
M-III: (l,r) search with AR(1-NN).	2	**4 3** (1 2)	94.12
M-IV: forward with $I(Y\|\mathbf{X}_s)$ continued with (l,r) search with AR(1-NN).	2	**4 3** (1 2)	94.12
M-V: Garson's saliency values through ANN	NA	3 (4 2) 1	NA

[a] Parentheses here denote that the features inside cannot be confidently ranked with the respective feature selection method. NA means not available.

Table 8. Methods M-I to M-V Compared on Flow Regime Classification

method	no. of var.	order[a]	AR(1-NN) (%)
none (all available features considered)	14	NA	91.86
M-I: forward selection with $I(Y\|\mathbf{X}_s)$. ($nbx = 50$)	8	**5 1 2 11 14 4 12 13** (3 6 7 8 9 10)	92.79
M-II: forward selection with AR(1-NN)	11	**1 7 14 2 12 11 9 4 13 10 3** (5 6 8)	93.04
M-III: (l,r) search with AR(1-NN)	11	**1 14 2 12 4 9 11 13 5 10 3** (6 7 8)	93.18
M-IV: forward with $I(Y\|\mathbf{X}_s)$ continued with (l,r) search with AR(1-NN)	11	**1 14 2 12 4 9 11 13 5 10 3** (6 7 8)	93.18
M-V: Garson's saliency values through ANN	NA	1 13 2 (4 5 6 7 8 9 10 11 12 3)	NA

[a] Parentheses here denote that the features inside cannot be confidently ranked. NA means not available. Meaning of the variables: 1, u_L; 2, u_G; 3, foam; 4, D_c; 5, ϵ; 6, a_G; 7, a_T; 8, dp; 9, ϕ; 10, ρ_L; 11, μ_L; 12, σ_L; 13, ρ_G; 14, μ_G.

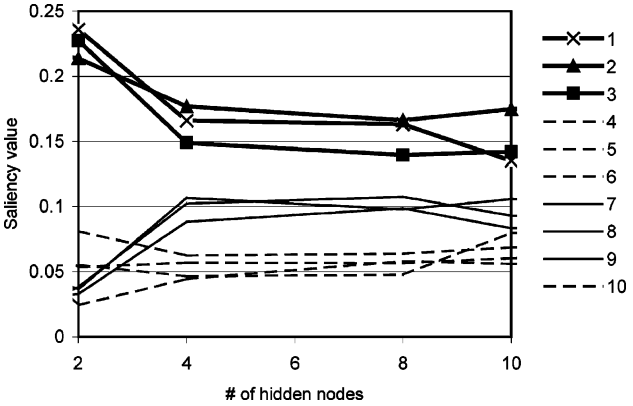

Figure 5. Saliency values for the synthetic problem (M-V)

saliency values over 10 distinct ANN training sessions for all 1 to 10 features and different numbers of hidden nodes were computed. At each training session, 4000 iterations of back-propagation with adaptive learning rate and momentum were used for a feed-forward neural network with sigmoid transfer functions in hidden and output neurons using 75% of the 300 available points. For a number of hidden nodes between 2 and 8, the accuracy rate *AR(ANN)* of the neural network classifier evaluated on the generalization set (remaining data 25%) approached 98%. It may therefore be concluded that the network was not over-fitting the training samples when the number of hidden nodes was less than 8 or under-fitting when the number of hidden nodes was equal to or more than 2.

As seen in Figure 5, the saliency values for the first 3 features clearly outperform the others, denoting their importance in the classification process. However, saliency index calculation does not help ranking or making distinction among them. Furthermore, the other somewhat significant features 4 to 6 appear to be even less relevant than the group of completely irrelevant features 7 to 10. We may conclude that Garson's method[13] is able to indicate the rank only if differences in the importance of variables is large.

The synthesis results of this problem are sketched in Table 6. To conclude, all the methods produced the same result. The method of Garson[15] identified the most important set but was unable to make confident distinctions between its constitutive features.

6.2. Anderson's Iris Data. The filter method M-I was first applied for this problem. The number of divisions in the domain for each feature was $nbx = 20$. The selected features were {4,3}, in agreement with previous findings.[20] Methods M-II through MIV produced exactly the same results with even better performance than if all features were used, Table 7. Garson's method[15] identified feature 3 as the most important and feature 1 as the least relevant, while the saliency values of features 2 and 4 could not help in concluding which of them is more important. For 4 to 10 hidden neurons, the generalization accuracy rate of the network was almost constant and approached 95%, denoting well-trained networks.

6.3. Three-Class Flow Regimes Classification in Trickle Beds. For this problem, it was desirable to find which variables among those listed in Table 4 are most likely to be predictive for the flow regimes: LIR, TR, and HIR. The summary of the analysis of the different methods is given in Table 8. For methods M-I to M-IV, the solution subset $\mathbf{X}_s$ constituted all variables added (and not removed) until the epoch when the relevance criterion reached a maximum value.

Method M-III gives better results than method M-II (Figure 6) since (l,r) search allows backtracking and removes variable 7 at the end of step 4 (epoch 12). This

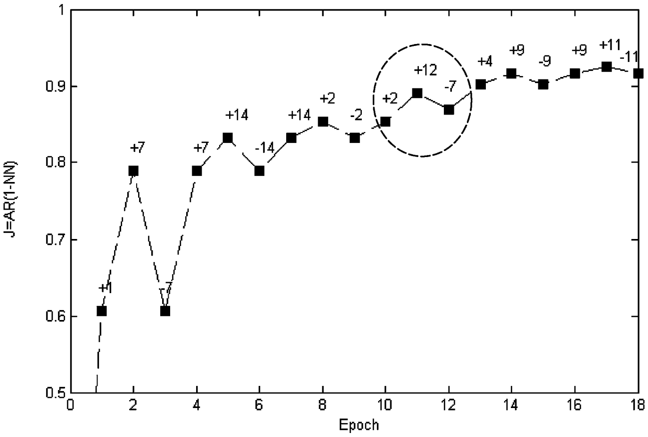

Figure 6. (l,r) search with accuracy rate as relevance criterion (M-III). Only the first 6 steps are shown for clarity. Dotted circle shows the 4th step. "+"/"−" means that the corresponding feature was added to/deleted from the current subset.

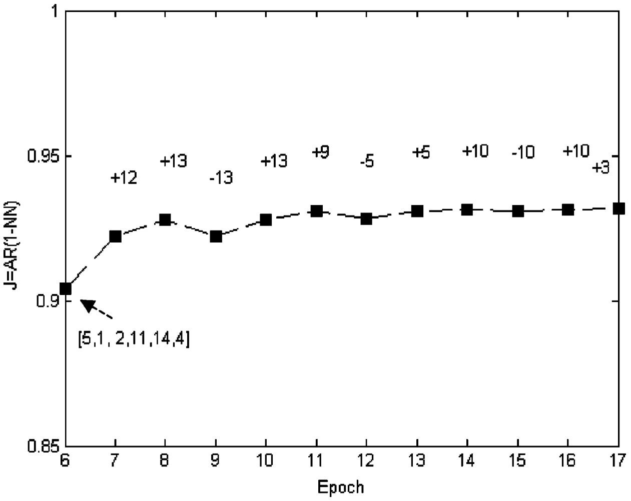

Figure 7. Method M-IV applied to flow regime problem. The method starts with the first six variables found with M-I and continues until no more increase in the accuracy rate of a 1-NN classifier is observed. "+"/"−" means that the corresponding feature was added to/deleted from the current subset.

variable, i.e., bed specific area, a_T, was removed even if it was shown to be the best at epoch 2 in conjunction with variable 1 (liquid velocity, u_L). As this problem involves the largest number of available features, p, it is opportune to illustrate method M-IV that we advocate in these circumstances. It starts with the first 6 most relevant features found with M-I, and then it continues to grow and prune this preselection (Figure 7). As can be seen from Figure 7, the (l,r) search starting with initial pre-selection {5,1,2,11,14,4} continued to improve the J value.

The Garson's method[15] was unsuccessful in providing meaningful ranking except for the first 3 variables. It can underline only that the liquid velocity (feature 1), the gas density (feature 13), and the gas velocity (feature 2) are important whereas the other features could not be confidently ranked.

The subset of variables (identified by methods M-III and M-IV): $\mathbf{X}_s = \{u_L, \mu_G, u_G, \sigma_L, D_c, \phi, \mu_L, \rho_G, \epsilon, \rho_L, \mathit{Foam.}\}$ is most likely sufficient for predicting the flow regime classes.

There are plenty of tools in the literature that allow identification of flow regimes in the form of flowcharts and empirical or fully conceptual correlations for the liquid velocity ($u_{L,tr}$) that demarcates the transition between the LIR and HIR.[1] Most of these methods involve only a few variables (features) to indicate the transition between LIR and HIR while generally lacking robustness when tested thoroughly.[21] Flowcharts like those of Turpin et al.[23] or Sato et al.[24] use only the gas and liquid mass flow rates (involving only the variables u_L, u_G, ρ_L, ρ_G), being thus restrictive and they apply mainly to water-like and air-like fluids. More recent correlations[25,26] attempt to predict $u_{L,tr}$ by taking into account also σ_L, μ_L, and eventually ϵ. A comprehensive correlation for the liquid transition velocity was also recently proposed by Larachi et al.[21] by taking into consideration the variables ρ_L, μ_L, σ_L, u_G, ρ_G, u_L, μ_G, ϵ, and d_p.

Hence, these variables inventoried from the literature to be important in flow regime identification are included directly or indirectly within the subset identified using the present selection feature algorithms. Note that the particle diameter is indirectly involved through embedding in the sphericity and the bed porosity defined in $d_P = 6(1 - \epsilon)/a_T$ and $\phi = \pi((6(1 - \epsilon)/\pi N_P))^{2/3} \times (N_P/a_T)$ (N_p: number of particles per unit bed volume).

6.4. Two-Class Bed Expansion/Contraction in Three Phase Fluidized Beds. In this problem, there are 10 features susceptible to indicate whether there will be bed contraction/expansion upon introduction of a tiny gas flow rate in the initially liquid fluidized bed. Using all the features gave *AR(1-NN)* = 96.6%. The solution provided by methods M-I to M-IV is $\mathbf{X}_s$ = {5, 4, 1, 7, 9}. When only these 5 variables were used, the accuracy rate decreased slightly to *AR(1-NN)* = 96.3%. The first method, M-I, based on mutual information criterion, induced the following order of preference:

5	4	1	7	9
ρ_s	σ_L	u_L	u_t	H_0

while the other methods M-II to M-IV suggested rather the following order:

7	4	1	9	5
u_t	σ_L	u_L	H_0	ρ_s

Figures 8 and 9 show the difference between the importance of the same selected variables by the two different relevance criteria: respectively, mutual information (M-I) and accuracy of a nearest neighbor classifier (method M-II). As the search technique is the same in the two cases (sequential forward selection), the difference resides only in the relevance criterion. As can be seen, mutual information (M-I) gives comparable importance to the solid density, liquid surface tension, particles terminal velocity and liquid superficial velocity, and a lower importance to the bed height at rest. Method M-II shows that the terminal velocity is much more important than all the others in the selection and the rest of the features have somehow comparable contributions to the class predictability. Method M-V cannot provide confident ranking of the features, as the saliency values are not significantly different for the 10 variables.

The Elite subset of variables (identified by methods MI to M-IV): $\mathbf{X}_s = \{u_t, \sigma_L, u_L, H_0, \rho_s\}$ appears to be most pertinent for predicting the initial behavior of the fluidized bed. Of course this conclusion (as in the case

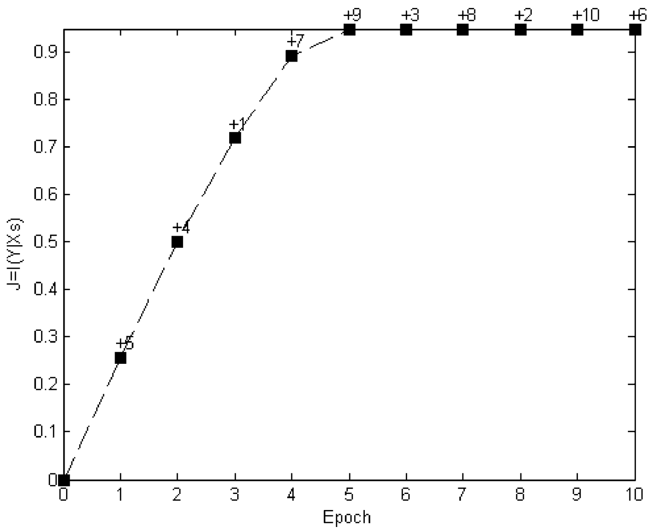

Figure 8. Sequential forward selection with mutual information as relevance criterion (M-I) on the bed expansion/contraction problem. "+" means that the corresponding feature was added to the current subset.

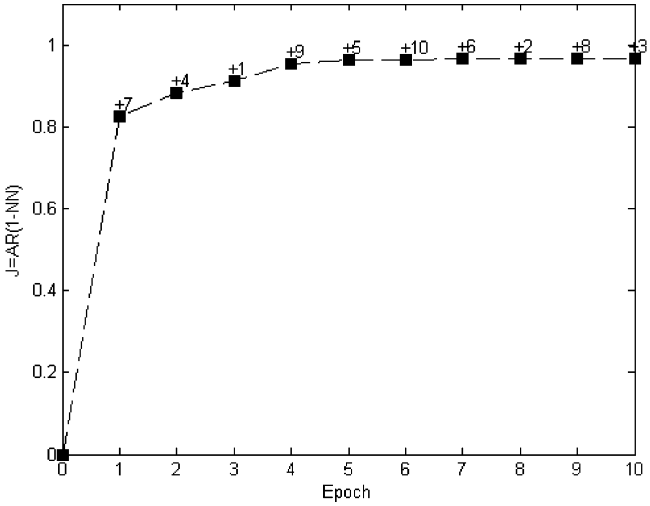

Figure 9. Sequential forward selection with accuracy rate as relevance criterion (M-II) on the bed expansion/contraction problem. "+" means that the corresponding feature was added to the current subset.

of trickle bed flow regimes) is based on the available data and backed by some physical grounds highlighted next.

For a liquid fluidized bed containing small solid particles, one expects to observe an increase in the bed porosity $\epsilon = 1 - \epsilon_s$ at superimposing a gas stream to a liquid fluidized bed (ϵ_s = solid hold-up). In some instances, however, the bed may initially contract until it reaches a certain point beyond which the bed height resumes its increase with an increase in gas flow rate. The accepted interpretation for this phenomenon is that bubbles entrain the liquid and particles into their wakes, thereby reducing the effective amount of liquid in the bed used to fluidize the remaining portion of particles.[27]

Bed contraction/expansion phenomena are conceptualized through the generalized wake model,[28,29] which suggests that the liquid velocity, the particle terminal velocity (both present in $\mathbf{X}_s$), and the k and x bubble wake parameters are controlling the initial bed state. The bubble wake parameters are influenced[28] among other parameters, by σ_L and ρ_s, both belonging to $\mathbf{X}_s$ too. Jiang et al.[27] acknowledged also that bed contraction is closely linked with the presence of large bubbles whose number is affected somehow by the liquid superficial tension, σ_L, whereas the Soung[30] empirical correlation for initial bed state indicates that u_t plays a role which increases for higher u_L/u_G ratios.

Regarding bed height at rest (H_0), the feature selection algorithms detect that this variable has an influence but which is relatively marginal with respect to the other ones in set $\mathbf{X}_s$ (see Figures 8 and 9). This is coherent with the literature studies that have also shown that higher or lower initial bed heights could affect bed expansion/contraction.

7. Conclusion

We studied in this work some feature selection algorithms, which allow identifying the most relevant variables in two multiphase reactors classification problems. Relevance here was assessed in terms of the following: (i) mutual information which measures the dependence among the features and the class variable; (ii) accuracy rate of a one-nearest neighbor classifier known to work well on most problems. The first criterion belongs to the class of filters and is a statistical measure of "goodness" of variables to filter out those that are irrelevant. The second belongs to the wrappers category and uses a particular classifier (here 1-NN) to test relevance. A third relevance criterion based on the saliency index[15] interprets the size of the weights of trained neural network classifiers that we extended to work with multiple outputs neural networks.

The selection algorithms that targeted maximization of the relevance criteria (mutual information and accuracy rate) in charge of the combinatorial search were the sequential forward selection and the (l,r) search method.

We devised here a hybrid filter-wrapper approach, which in the first step uses the mutual information criterion and SFS to get a set of features informative about the class to be predicted. This set is further updated using the (l,r) search to maximize the performance rate of a 1-NN classifier. This last method is faster that a mere (l,r) search starting with a empty set and gives the same results on the test problems.

The different selection schemes were applied to four problems. The first two problems were benchmarks (synthetic and real) to test whether the schemes capture the proper solutions. The last two problems were borrowed from the area of multiphase reactors: (i) flow regime identification in trickle beds and (ii) bed initial contraction/expansion in three-phase fluidized bed reactors.

For both latter problems, the ULaval multiphase reactors databases[21,22] were used to identify among the numerous variables (features) the most relevant ones for classification. The feature reduction induced in all cases an increase in classification performance, except for the bed expansion/contraction, where reducing the number of variables from 10 to 5 decreased very slightly the accuracy. In the case of flow regime classification, the following variables were found to be important in trickle beds: u_L, μ_G, u_G, σ_L, D_c, ϕ, μ_L, ρ_G, ϵ, ρ_L, Foam. For bed expansion/contraction in three-phase fluidized beds, the most relevant features appeared to be ρ_s, σ_L, u_L, u_t, and H_0. Further work[14] will consist of using these features to design reliable and portable classifiers with a minimum number of adjustable parameters.

Appendix

The (l,r)-search algorithm, or its particular cases (1,0)-search (or SFS) and (0,1)-search (or SBS) as described by Pudil et al.[10] adapted for the current paper notations.

```
Input:           $\mathbf{X}_p = \{X_{p,i} \mid i = 1,...,p\}$
                 //the set of all features//
Output:          $\mathbf{X}_{s,d} = \{X_{s,i} \mid i = 1,...,d, X_{s,i} \in \mathbf{X}_p\}$, $d = 0,1,...,p$;
                 //the selected subset of size $d$//
Initialization:  if $l > r$, then $d := 0$; $\mathbf{X}_{s,d} = \phi$; go to Step 1
                 else $d := p$; $\mathbf{X}_{s,d} = \mathbf{X}_p$; go to Step 2
Termination:     Stop when $d$ equals the number of
                    features required

    Step 1 (Inclusion)
    Repeat $l$ times
        $X^+ := \arg \max_{X \in \mathbf{X}_p - \mathbf{X}_{s,d}} J(\mathbf{X}_{s,d} + X)$
            //the most significant feature with respect to $\mathbf{X}_{s,d}$ //
        $\mathbf{X}_{s,d+1} := \mathbf{X}_{s,d} + X^+$; $k := k + 1$
    Step 2 (Exclusion)
    repeat $r$ times
        $X^- := \arg \max_{X \in \mathbf{X}_{s,d}} J(\mathbf{X}_{s,d} - X)$
            //the least significant feature in $\mathbf{X}_{s,d}$ //
        $\mathbf{X}_{s,d-1} := \mathbf{X}_{s,d} - X^-$; $k := k - 1$
    go to Step 1
```

Notation

a_G = grain specific area
AR = accuracy rate of a classifier $Ar = 1 - P_e$
a_T = bed specific area
d = dimension of a subset of the set $\mathbf{X}_p$, $d \leq p$
D_c = column diameter
d_E = Euclidian distance
d_p = particle diameter $d_P = 6(1 - \epsilon)/a_T$
Foam = foaming property: 0 = coalescing, 1 = foaming
H = the entropy function
H_0 = bed height at rest
$I(Y|\mathbf{X}_s)$ = mutual information (information content) given by $\mathbf{X}_s$ on Y
J = relevance criterion ($I(Y|\mathbf{X}_s)$ or *AR(1-NN)*); Number of hidden nodes in an ANN
n = number of training instances ω_k
nbx = number of divisions in the space of each variables when assessing probabilities using bins
N_c = number of classes
N_p = number of particles per unit bed volume
p = number of all features available for a classification problem, i.e., size of $\mathbf{X}_p$
P_e = prediction error, representing the number of misclassified samples divided by the number of all samples
$S_{ind}(i,k)$ = saliency value for the input i with respect to the output k in a ANN
u = phase velocity
u_t = particle terminal velocity in a three-phase fluidized bed
$w[i,j]$ = input to hidden layer weight in the ANN
$w[j,k]$ = hidden to output layer weight in the ANN
$\mathbf{x}_p$ = a particular realization of $\mathbf{X}_p$
$\mathbf{X}_p$ = set of all available features
$\mathbf{X}_s$ = subset of features from $\mathbf{X}_p$. $\mathbf{X}_s$ <@@Unknown-xcd@@> $\mathbf{X}_p$
y = a particular value of the generic class variable Y
ϵ = bed porosity
ϕ = particle sphericity $\phi = \pi((6(1 - \epsilon)/\pi N_P))^{2/3} \times (N_P/a_T)$
μ = phase viscosity
ρ = phase density
σ = phase superficial tension
ω_k = training instance $\omega = \{\mathbf{x}_p, y\}$

Abbreviations

ANN = artificial neural network (here this term designates a multilayer perceptron neural network)
G = gas
HIR = high interaction regime
IBC = initial bed contraction
IBE = initial bed expansion
L = liquid
LIR = low interaction regime
S = solid
SFFS = sequential floating forward selection
SFS = sequential forward selection
TR = transition flow regime

Literature Cited

(1) Dudukovic, M. P.; Larachi, F.; Mills P. L. Multiphase catalytic reactors: A perspective on current knowledge and future trends. *Catal. Rev. Sci., Eng.* **2002**, *44*, 123–246.

(2) Jain, A. K.; Duin, P. W.; Mao, J. Statistical pattern recognition: A Review. *IEEE Trans. Pattern Anal. Machine Intelligence* **2000**, *22* (1), 4–37.

(3) Bishop, C. M. *Neural Networks for Pattern Recognition*; Clarendon Press: Oxford, 1995.

(4) Sebban, M.; Nock, R. A hybrid filter/wrapper approach of feature selection using information theory. *Pattern Recognition* **2002**, *35*, 835–846.

(5) John, G. H.; Kohavi, R.; Pfleger, K. Irrelevant features and subset selection problem. *Proc. 11th Int. Conf. Machine Learning* **1994**, 121–129.

(6) Shannon, C. E.; Weaver, W. *The Mathematical Theory of Communication*; University of Illinois Press: Urbana, IL, 1949.

(7) Ash, R. B. *Information Theory*; Dover Publications: New York, 1990.

(8) Batitti, R. Using mutual information for selecting features in supervised neural net learning. *IEEE Trans. Neural Networks* **1994**, *5* (4), 537–550.

(9) Sridhar, D. V.; Bratlett, E. B.; Seagrave, R. C. Information theoretic subset selection for neural network models. *Comput. Chem. Eng.* **1998**, *22* (4/5), 613–626.

(10) Pudil, P.; Ferri, F. J.; Novovicova, J.; Kittler, J. Floating Search Methods for Feature Selection with Nonmonotonic Criterion Functions. *Proc.—Int. Conf. Pattern Recognit.* **1994**, *2*, 279–283.

(11) Stearns, S. D. On selecting features for pattern classifiers. In *Third Int. Conf. On Pattern Recognition*, Colorado, CA, 1976; pp 71–75.

(12) Narendra, P. M.; Fukunaga, K. A branch and bound algorithm for feature subset selection. *IEEE Trans. Comput.* **1977**, *C-26* (9), 917–922.

(13) Siedlecki, W.; Sklansky, J. On automatic feature selection. *Int. J. Pattern Recognit. Artif. Intelligence* **1988**, *2* (2), 197–220.

(14) Tarca, L. A.; Grandjean, B. P. A.; Larachi, F. Integrated genetic algorithm –artificial neural network strategy for modeling important multiphase flow characteristics. *Ind. Eng. Chem. Res.* **2002**, *41*, 2543.

(15) Garson, G. D. Interpreting Neural Network Connection Weights. *AI Expert* **1991**, *6* (7), 47–51.

(16) Nath, R.; Rajagopalan, B.; Ryker, R. Determining the saliency of input variables in neural network classifiers. *Comput. Oper. Res.* **1997**, *24* (8), 767–773.

(17) Tarca, L. A.; Grandjean, B. P. A.; Larachi, F. Designing supervised classifiers for multiphase flow data classification. *Chem. Eng. Sci.* **2004**, *59*, 3303–3313.

(18) Kohavi, R. A study of cross-validation and bootstrap for accuracy estimation and model selection. *Proc. 15th Int. Joint Conf. Artif. Intelligence* **1995**, 1137–1143.

(19) Rumelhart, D. E.; Hinton, G.; Williams, R. Learning internal representation by error propagation; *Parallel Distributed Processing*, 1; MIT Press: Boston, 1986; pp 318–362.

(20) Li, R.; Mukaidono, M.; Turksen, I. B. A fuzzy neural network for pattern classification and feature extraction; *Fuzzy Sets Syst.* **2002**, *130*, 101–108.

(21) Larachi F.; Iliuta I.; Chen, M.; Grandjean, B. P. A. Onset of pulsing in trickle beds: evaluation of current tools and state-of-the-art correlation. *Can. J. Chem. Eng.* **1999**, *77*, 751−758.

(22) Larachi, F.; Belfares, L.; Iliuta, I.; Grandjean, B. P. A. Three-Phase Fluidization Macroscopic Hydrodynamics Revisited. *Ind. Eng. Chem. Res.* **2001**, *40*, 993−1008.

(23) Turpin, J. L.; Huntington R. L. Prediction of pressure drop two-phase, two component concurrent flow in packed beds. *AIChE J*. **1967**, *13*, 1196−1202.

(24) Sato, Y.; Hirose, T.; Takahashi, F.; Toda, M.; Hashiguchi, Y. Flow pattern and pulsation properties of concurrent gas−liquid downflow in packed beds. *J. Chem. Eng. Jpn.* **1973**, *6*, 313−319.

(25) Dudukovic, M. P.; Mills, P. L. Contacting and hydrodinamics in trickle-bed reactors. In *Encyclopedia of Fluid Mechanics*; Chereminisoff, M. P., Ed.; Huston Gulf Publishing: 1986; pp 969−1017.

(26) Wang, R.; Mao, Z. S.; Chen, J. Y. A study of trickle-to-pulse flow transition in trickle-bed reactors (TBR). *Chem. Eng. Commun*. **1994**, *127*, 109−124.

(27) Jiang, P.; Luo, X.; Lin, T.; Fan, L. High temperature and high-pressure three-phase fluidization-bed expansion phenomena. *Powder Technol.* **1997**, *90*, 103−113.

(28) Bhatia, V. K.; Epstein, N. Three phase fluidization: a generalized wake model. In *Fluidization and Its Applications*; Angelino et al., Eds.; Cepadues-Editions: Toulouse, 1974; pp 380−392.

(29) Jean, R.; Fan, L. S. A simple correlation for solids holdup in a gas−liquid−solid fluidized bed. *Chem. Eng. Sci*. **1986**, *41* (11), 2823−2828.

(30) Soung, W. Y. Bed Expansion in Three-Phase Fluidization. *Ind. Eng. Chem. Process Des. Dev.* **1978**, *17*, 7−33.

Received for review April 24, 2004
Revised manuscript received November 25, 2004
Accepted December 6, 2004

IE049667A