BIOINFORMATICS ORIGINAL PAPER
Vol. 21 no. 15 2005, pages 3138–3145
doi:10.1093/bioinformatics/bti494

Data and text mining

Proteomic mass spectra classification using decision tree based ensemble methods

Pierre Geurts[1,*], Marianne Fillet[2], Dominique de Seny[2], Marie-Alice Meuwis[2], Michel Malaise[2], Marie-Paule Merville[2] and Louis Wehenkel[1]

[1]Department of Electrical Engineering and Computer Science and [2]Laboratory of Clinical Chemistry and Rheumatology, CBIG—Centre of Biomedical Integrative Genoproteomics, University of Liège, 4000 Liège, Belgium

Received on April 13, 2005; revised and accepted on May 6, 2005
Advance Access publication May 12, 2005

ABSTRACT

Motivation: Modern mass spectrometry allows the determination of proteomic fingerprints of body fluids like serum, saliva or urine. These measurements can be used in many medical applications in order to diagnose the current state or predict the evolution of a disease. Recent developments in machine learning allow one to exploit such datasets, characterized by small numbers of very high-dimensional samples.

Results: We propose a systematic approach based on decision tree ensemble methods, which is used to automatically determine proteomic biomarkers and predictive models. The approach is validated on two datasets of surface-enhanced laser desorption/ionization time of flight measurements, for the diagnosis of rheumatoid arthritis and inflammatory bowel diseases. The results suggest that the methodology can handle a broad class of similar problems.

Supplementary information: Additional tables, appendicies and datasets may be found at http://www.montefiore.ulg.ac.be/~geurts/Papers/Proteomic-suppl.html

Contact: p.geurts@ulg.ac.be

1 INTRODUCTION

Modern mass spectrometry (MS) generates protein profiles from body fluids like serum, saliva or urine. In medical applications, the analysis of such measurements can help to better understand the differences between various diseases, and to develop new diagnostic criteria. For example, the information contained in mass spectra could be used to improve sensitivity and/or specificity of predictive models used in medical diagnosis and prognosis. It could also help to monitor the response of patients to a given treatment or drug.

Datasets from proteomic MS are typically characterized by very high-dimensional input spaces (several thousand variables) but relatively small numbers of samples (a few hundred at best), which precludes the use of classical statistical techniques such as linear discriminants, or even neural networks, unless one is able to reduce very drastically the problem dimensionality. On the other hand, recent developments in machine learning (ML), namely tree-based ensembles and kernel-based methods, allow one to exploit such datasets without prior feature selection or extraction.

The application of ML techniques to mass-spectra classification has been recently proposed by several researchers, which use for some specific problems, among other methods, decision tree induction (Fung and Enderwick, 2002; Rai *et al*., 2002), tree based ensemble methods, e.g. boosted decision trees (Qu *et al*., 2002), random forests (Izmirlian, 2004) or specific new methods (Li *et al*., 2003), support vector machines (SVM) (Pusch *et al*., 2003; Jong *et al*., 2004) or several of these methods (Wu *et al*., 2003; Liu *et al*., 2002). In the present paper, we go one step further in this direction by proposing a systematic approach to solve a large class of clinical proteomics problems. Our approach aims, in particular, at being independent of the peak detection algorithms and other preprocessing methods which have been developed on an ad hoc basis for specific data acquisition conditions, so as to increase flexibility and broaden the range of application. The proposed software aims both at developing diagnostic tools and/or at identifying biomarkers depending on the focus of the particular study under consideration. It includes clearly defined pre- and post-processing steps as well as the invocation of a toolbox of generic decision tree based methods. We choose decision tree methods because these methods are computationally very efficient and can be easily exploited to assist physicians in identifying among a large number of candidate biomarkers, those that are best suited for a particular discrimination task. We provide results of our approach on datasets of two experimental studies based on surface-enhanced laser desorption/ionization time of flight mass spectrometry (SELDI-TOF-MS). They concern the diagnosis of patients suffering from inflammatory diseases, namely rheumatoid arthritis (RA) and inflammatory bowel diseases (IBD).

This paper is organized as follows: Section 2 describes the generic problem that we want to solve and its requirements from the ML viewpoint. Section 3 describes the main steps of the algorithmic solution that we propose and Section 4 provides a summary of our experimental results. Finally, Section 5 provides the overall conclusions.

2 GENERIC PROBLEM STATEMENT

2.1 Practical objective

The overall objective is to use experimental datasets obtained from proteomic MS for identification of one or several biomarkers specific of a given disease, or enable discrimination among a certain class of diseases, or be indicative of treatment response, and for the construction of predictive models exploiting the biomarker

*To whom correspondence should be addressed.

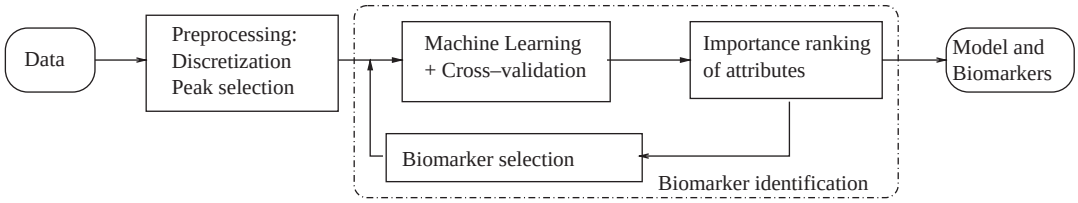

Fig. 1. The different steps of the proposed algorithmic solution.

intensities to help physicians in the context of medical diagnosis or prognosis.

2.2 Proteomic mass spectrometry datasets

The datasets are obtained from biological samples collected from different patients classified in different classes (e.g. disease versus control, disease A versus disease B, successful versus unsuccessful treatment ...), processed by a mass spectrometer after sample fractionation in different physical conditions (so-called chip surfaces).

The mass spectrometer typically provides signal intensities in a range of mass to charge (m/z) ratios between 0 and 20 000 Da, with a typical accuracy of 0.25%. This leads to a vector of 5000 to 20 000 numerical values for each mass spectrum analysis. In practice, for a given patient these analyses can be obtained from the sample pre-processed on several different chip surfaces and in several replicas, thus potentially leading to over 100 000 numerical variables per patient.

While the number of variables can be very high in these applications, the number of patients (in other words, of samples) is typically rather small (say, several tens of patients only for each class). This leads to rather untypical ML problems where the number of input variables is several orders of magnitude higher than the number of samples.

2.3 Discussion of specific ML requirements

ML offers various paradigms to extract information from datasets. In the context of the present application, the so-called supervised learning paradigm is applicable where the datasets are composed of samples described by input variables and a specific output information, and the objective is to derive from the dataset a synthetic model which predicts the output information of a sample as a function of its input variables. Actually, we will use the learning algorithm to construct classification models, since the output information is a discrete label (type of disease, success/nosuccess...).

There exists a multitude of algorithms for supervised classification, such as neural networks, genetic algorithms, discriminant analysis or decision trees. However, given the specificities of our application, the algorithm must cope with datasets where the number of variables is (much) larger than the number of samples and identify automatically and explicitly the informative variables, so as to determine biomarkers for the task at hand.

Decision tree based techniques appear to fit well to these problem characteristics. The computational complexity of these methods is (in the worst case) linear in the number of input variables, and they are also able to cope with problems where the input space dimensionality is larger than the number of samples and/or where the large majority of input variables are irrelevant. They are, furthermore, totally autonomous and computationally very fast. Moreover, they can be easily exploited in order to identify a subset of relevant input variables i.e. of candidate biomarkers for further analysis.

The combination of the basic decision tree induction method (e.g. CART (Breiman *et al.*, 1984)) with various ensemble approaches (bagging; Breiman, 1996; boosting; Freund and Schapire, 1995 ...) leads itself to a rather rich class of algorithms, which all share the computational and functional properties discussed above. However, for a given application it is not possible to predict in advance which one of these methods would lead to the best compromise between sensitivity and specificity. Thus we advocate at this level a toolbox approach, applying in parallel a sufficiently rich class of decision tree based algorithms and retaining the best results obtained.

In the rest of this paper we use the term attribute to denote an input variable used in a supervised learning problem, class to denote the (value of the) output variable and learning set to denote a dataset used by a supervised learning algorithm.

3 PROPOSED METHODOLOGY

The different steps of our algorithmic solution are represented in Figure 1. They are described in detail below:

3.1 Pre-processing

Proteomic MS provides usually rather noisy signals, both in terms of intensities and mass to charge ratios. In particular, small m/z errors may lead to large distances in the attribute space, and it is preferable to filter this noise out before applying the ML algorithms. We propose to use for this purpose a simple m/z ratio discretization algorithm, which roughness can be adapted to the problem at hand. Given the value of a roughness parameter r in $[0, 1]$, it works as follows:

1. Let m be the first m/z ratio present in a spectrum;
2. Create a new attribute equal to the cumulative intensities in the m/z interval $[m, m + r.m]$, and set $m = m + r.m$;
3. Unless m is larger than the largest m/z value in the original data, return to step 2.

This technique is compared with the peak detection and alignment software developed by the manufacturer of the SELDI-TOF-MS device that we used (ProteinChip Biomarker 3.0 by Ciphergen Biosystems, Inc.,). This latter removes m/z values that do not correspond to a 'real peak' in a sufficient number of spectra and aligns the spectra so that their peaks correspond to the same m/z ratio.

3.2 Machine learning algorithms

We first describe the model building techniques based on decision tree induction that we use and then the leave-one-out validation method that we use to evaluate their generalizability to unseen cases.

3.2.1 Model building For single decision tree induction, we use the CART algorithm with cost–complexity pruning by 10-fold cross-validation (Breiman *et al*., 1984), together with an information theoretic score measure described in Wehenkel (1998). In addition to CART we use several decision tree based ensemble methods, so as to reduce variance and bias and hence improve accuracy and reliability. Notice that many empirical studies have been carried out comparing different tree based ensemble methods (Bauer and Kohavi, 1999; Dietterich, 2000) and no method has been found superior to all others in all situations. For this reason, we use the four following methods in parallel:

- Bagging (Breiman, 1996) builds tree ensembles (CART algorithm without pruning) from bootstrap samples (i.e. samples of the same size as the learning sample drawn with replacement from it), and aggregates tree predictions by majority vote.
- Random forests (Breiman, 2001) directly derive from bagging. At each test node, they select K attributes at random among all candidate attributes before determining the split. We use the default setting of K, equal to the square root of the total number of candidate attributes.
- Extra-trees (P. Geurts, D. Ernst and L. Wehenkel, submitted for publication) grow trees from the complete learning set by selecting at each node the best among K randomly generated splits. We use the same setting for K as in the random forests. This method is described in the appendix in the supplementary data.
- Boosting (Freund and Schapire, 1995) builds trees sequentially and deterministically with CART, but by increasing the weights of the learning set samples that are misclassified by the previous trees of the sequence, and by aggregating their predictions by a voting scheme where each tree is weighted according to its accuracy on the learning set. In our simulations, we have used the so-called Adaboost algorithm.

In our experiments, we use ensembles of 100 trees for each method. For comparison purpose, we have also run experiments with C4.5 (Quinlan, 1986) as the base learner. These results will be discussed below.

3.2.2 Model cross-validation and selection Because of the very small size of the medical datasets, we use leave-one-out cross-validation to estimate the accuracy of our models. It consists in removing each sample in turn from the learning set, building a model from the remaining $n-1$ samples, then classifying this sample with this model, and counting the overall frequency of different types of errors. For binary classification problems, the accuracy of a model is then measured by three values:

- Sensitivity: % of samples from the target class that are well classified by the model (true positives).
- Specificity: % of samples from the other class that are well classified by the model (true negatives).
- Error rate: % of samples (whatever their classes) that are misclassified by the model.

In practice, the selection of the best among several models according to these three measures depends on the importance or cost of misclassification in each class. In our experiments, we will select models on the basis of their error rate.

Notice that since the final model is eventually selected by leave-one-out cross-validation, this latter error rate might be slightly over-optimistic and, in any case, suffers from high variance because of the small sample sizes. Hence, we recommend that in practice final validation is done on another sufficiently large set of independent observations. Notice also that in cases where several observations in the learning set come from the same patient (replicas), we do a patient-based leave-one-out validation by removing in each fold all the data related to a patient.

3.3 Biomarker identification

The biomarker identification procedure works in two successive steps. First, attributes (corresponding to intensities in ranges of m/z ratios) are ranked by decreasing order of importance. Second, an optimal subset of biomarkers is identified by cross-validation.

3.3.1 Attribute importance ranking The tree-based algorithms we use allow to easily compute an attribute importance measure for a given classification problem. Among the importance measures proposed in the literature (Breiman *et al*., 1984; Breiman, 2001; Hastie *et al*., 2001), we use the information measure from (Wehenkel, 1998). Namely, at each test node, we compute the total reduction of the class entropy due to the split, which is defined by:

$$I(node) = \#S H_C(S) - \#S_t H_C(S_t) - \#S_f H_C(S_f), \qquad (1)$$

where S and $\#S$ denote, respectively, the subset of samples that reach this node and its size, S_t (S_f) denotes the subset of them for which the test is true (false) and $H_C(\cdot)$ computes the Shannon entropy of the class frequencies in a subset of samples. Thus, a split is considered as more important if it concerns many cases ($\#S$ is large) and at the same time discriminates well between their classes. The overall importance of an attribute for the classification problem is then computed by summing the I values of all tree nodes (of a single tree, or of an ensemble of trees), where this attribute is used to split. In the case of boosting, we use a weighted sum taking into account the different weights of the different trees in the ensemble. Those attributes that are not selected at all obtain zero, and those attributes that are selected close to the root nodes of the trees typically obtain high scores. For the sake of presentation, we normalize the importances obtained in this way for the different candidate attributes so that they sum up to 100%.

3.3.2 Biomarker selection It is often difficult to define *a priori* an importance threshold below which one can discard attributes. Thus, to automatically select an optimal subset of of biomarkers, we use the following overall procedure:

- A model is built with all candidate attributes (obtained after pre-processing) and with each ML algorithm; the best one is selected by cross-validation (see Section 3.2.2) and used to determine the importance ranking;
- The ML algorithms are run again by using only the top ranked attributes, while progressively increasing their number;
- The accuracy estimates (by leave-one-out) of the resulting sequence of models are computed so as to determine a learning curve which, typically, first increases then reaches a maximum and decreases again.

- The attributes corresponding to the maximum accuracy (within some tolerance) are then retained as the optimal set of biomarkers.

Notice that since the error estimates are here based on a very small sample, the location of the exact minimum in the learning curve can be unstable. Thus, we use the so-called one standard error rule defined in Breiman *et al*. (1984), which selects the smallest set of attributes that gives an error not greater than $\mathrm{Err}^* + \sigma^*$, where Err^* is the minimal error and σ^* is an estimate of the standard deviation of this error. In our experiments, the standard deviation σ^* is estimated by $\sqrt{\mathrm{Err}^*(1-\mathrm{Err}^*)/n}$ where n is the size of the learning set. Notice that being more conservative, this rule yields a smaller set of biomarkers and also less biased accuracy estimates.

3.4 Remarks

3.4.1 Selection bias We stress the already mentioned fact that since the cross-validation error rates are used to select models and/or biomarkers, they should not be considered as an unbiased way of estimating the error rates of the finally obtained models. For this purpose, either a second level cross-validation technique needs to be superimposed upon the described procedure (see Ambroise and McLachlan, 2002, and the related discussion below in Section 4.3.2) or preferably, an independent test set of carefully selected patients should be used for final validation.

3.4.2 Other ranking schemes The biomarker selection method can be combined with any other attribute-ranking scheme that is deemed of interest, such as for example the so-called P-values ranking compared in Section 4.3.2.

3.4.3 Models built with a reduced number of biomarkers After final biomarker selection one can use the resulting subset of attributes to build a new model. This may reduce variance and hence further increase accuracy, especially with very small datasets (see Section 4.3.2).

4 RESULTS AND DISCUSSION

4.1 Datasets

Our experiments use two datasets of SELDI-TOF-MS obtained from serum samples of several patients. The main goal is to detect patients suffering from one particular inflammatory disease: RA for the first dataset and IBD for the second. In both datasets, the control group is composed of samples of healthy patients and patients affected by other similar inflammatory diseases. Samples were collected at the University Hospital of Liège from 2002 until present. In the first problem, two serum samples have been collected and analyzed per patient, and in the second problem four. The composition of each dataset in terms of the number of samples in the target and the non-target classes is given in Table 1. Details about these two datasets are given in the Appendix in the Supplementary data.

In both cases, several chip arrays were actually tested but we report only on the best results obtained among these latter. Mass spectra were obtained from chip arrays by a PBS II Protein Chip reader (Ciphergen Biosystems Inc.). Several standard pre-processing steps (baseline subtraction, normalization...) were applied to the resulting spectra before applying our methodology. A detailed description of the experiments is given in De Seny *et al*. (2005) (D. De Seny, M. Fillet, M.A. Meuwis, P. Geurts, L. Lutteri, C. Ribbens, V. Bours, L. Wehenkel, J. Piette, M. Malaise and M.P. Merville submitted for publication).

Table 1. Summary of the datasets: class composition and number of attributes for different discretization steps

Dataset	No. of target	No. of others	0.0%	0.3%	0.5%	1.0%	Peaks
RA	68	138	15445	1026	626	319	136
IBD	240	240	13799	1086	664	338	152

On both datasets, we have tried four different values of the parameter r: 0.0, 0.3, 0.5 and 1% and we have also used peak alignment and detection as carried out by the ProteinChip Biomarker Software version 3.0 (Ciphergen Biosystems, Inc.) with default parameters (Fung and Enderwick, 2002). Table 1 shows the resulting number of attributes.

4.2 Model building and validation

Table 2 compares the ML algorithms on the two problems with discretized spectra, reporting for each method the best results obtained by selecting the optimal roughness parameter r^*. Sensitivities, specificities, and error rates (in %) are leave-one-out estimates as indicated in Section 3.2.2. As our learning sets contain two or four replicas for each patient, we have removed all data corresponding to a single patient in each leave-one-out round, so as not to bias the estimates. Table 3 reports the results obtained with pre-processing by peak alignment and detection. All ensemble methods are quite close but single trees are clearly inferior. Using the data pre-processed by peak selection gives less good results on both problems. Overall, results in terms of sensitivity and specificity are quite good on both problems. For RA, the best result is obtained with boosting. On IBD, the best method is extra-trees. Results obtained in identical conditions with C4.5 as the base learner are given on the Supplementary data on web site. In general, they are slightly less good.

In the case of the pre-processing by discretization, we find that a value of $r = 1\%$ works better in almost all cases. Table 4 gives more details about the evolution of the error with this parameter (on RA with boosting, and on IBD with extra-trees).

For comparison purpose, we also report in Tables 2 and 3 results obtained with the k-nearest neighbors method (k-NN) and SVM. Implementation details related to these two algorithms are given in the Appendix in Supplementary data. We observe that on both problems and pre-processings, the k-NN method is clearly suboptimal in terms of accuracy. The results of the SVM method are better, but on the average less good than those of the best tree based method. Only on IBD with pre-filtered peaks, SVM provides the best results, but they are nevertheless slightly less good than those obtained on this problem by extra-trees without pre-filtered peaks.

4.3 Biomarker identification

4.3.1 Attribute importance ranking Table 5 gives for each problem the m/z interval and percentage of information (i.e. the importance) of the first ten attributes, respectively, ranked by a single CART tree and by boosting, with discretization ($r = 1\%$) and also with peak detection. The table also provides the ranking (R_p) of each attribute according to the P-values obtained by a statistical non-parametric Mann–Whitney test (Fung and Enderwick, 2002).

Table 2. Results with the best discretization on each problem

Method	RA dataset r^* (%)	Sensitivity	Specificity	Err.	IBD dataset r^* (%)	Sensitivity	Specificity	Err.
Single tree	1	66.18 (45/68)	86.23 (119/138)	20.39	1	81.67 (196/240)	81.17 (194/239)	18.58
Bagging	0.3	83.82 (57/68)	89.13 (123/138)	12.62	1	85.83 (206/240)	87.44 (209/239)	13.36
RF	1	89.71 (61/68)	85.51 (118/138)	13.11	1	85.42 (205/240)	92.05 (220/239)	11.27
ET	1	92.65 (63/68)	86.96 (120/138)	11.17	1	88.33 (212/240)	91.63 (219/239)	10.02
Boosting	1	83.82 (57/68)	94.93 (131/138)	8.74	1	87.08 (209/240)	92.05 (220/239)	10.44
k-NN	1	82.35 (56/68)	82.61 (114/138)	17.48	1	77.08 (185/240)	81.59 (195/239)	20.67
SVM	1	88.24 (60/68)	89.86 (122/138)	10.68	1	87.92 (211/240)	87.87 (210/239)	12.11

See Table 8 on the supplementary website for results obtained with C4.5 in the same conditions.

Table 3. Results with peak alignment and detection

Method	RA dataset Sensitivity	Specificity	Err.	IBD dataset Sensitivity	Specificity	Err.
Single tree	80.88 (55/68)	81.88 (113/138)	18.45	72.50 (174/240)	74.17 (178/240)	26.67
Bagging	75.00 (51/68)	87.68 (121/138)	16.50	84.58 (203/240)	82.92 (199/240)	16.25
RF	83.82 (57/68)	88.41 (122/138)	13.11	86.25 (207/240)	87.92 (211/240)	12.92
ET	89.71 (61/68)	86.23 (119/138)	12.62	84.17 (202/240)	87.50 (210/240)	14.17
Boosting	80.88 (55/68)	92.75 (128/138)	11.17	87.50 (210/240)	89.58 (215/240)	11.46
k-NN	77.94 (53/68)	80.43 (111/138)	20.39	87.08 (209/240)	72.50 (173/239)	20.21
SVM	83.82 (57/68)	88.41 (122/138)	13.11	90.42 (217/240)	89.17 (213/239)	10.21

See Table 9 on the supplementary website for results obtained with C4.5 in the same conditions.

Table 4. Effect of the discretization parameter, left on RA with boosting, right on IBD with extra-trees

r (%)	RA dataset Sensitivity	Specificity	Err.	IBD dataset Sensitivity	Specificity	Err.
0.0	88.23 (60/68)	88.40 (122/138)	11.65	85.00 (204/240)	87.45 (209/239)	13.78
0.3	85.29 (58/68)	90.58 (125/138)	11.17	85.42 (205/240)	91.21 (218/239)	11.69
0.5	85.29 (58/68)	90.58 (125/138)	11.17	85.83 (206/240)	91.21 (218/239)	11.48
1.0	83.82 (57/68)	94.93 (131/138)	8.74	88.33 (212/240)	91.63 (219/239)	10.02

We notice that rankings of the single tree and boosting models are quite different, but since the latter is much more accurate than the former, we deem its attribute ranking also more reliable. We also observe that 'peak detection' and our discretization method give different rankings. We believe that peak detection removes some important attributes that the ML methods have to replace with other (less informative) attributes, and so changes the order of rankings. Actually, on both problems the most important attributes (with boosting) correspond to two m/z ranges (1054–1064 on RA and 5177–5230 on IBD) where no peak was found by the peak detection method. Thus, the first m/z value detected by the boosting model with the peaks is only the second on RA and the third on IBD attribute in the ranking obtained with the discretization $r = 1\%$. This may also explain the difference in error rates between Tables 2 and 3.

While in Table 5 the first attribute given by the importance measure derived from ML models is (in every case) also ranked high (among the top three and most often first) according to the P-values, some important attributes are ranked rather low by these latter. This is explainable by the fact that the P-values detect only single-variable effects, while the importance ranking is a multivariate approach also detecting interacting effects of several attributes.

4.3.2 Biomarker selection We used on both problems the procedure of Section 3.3, while using the boosting models at the first step to determine attribute importances (with $r = 1\%$). Table 6 shows the best methods for increasing numbers of attributes ($N = 1, \ldots, 100$), together with the corresponding accuracies determined by leave-one-out cross-validation. The last line of the table corresponds to the results (from Table 2) when all candidate attributes are used. Those error rates falling within the range $[\mathrm{Err}^*, \mathrm{Err}^* + \sigma^*]$ are underlined. In both datasets, the error goes through a minimum when N increases and the minimum corresponds to a quite large number

Table 5. Attribute importances on both problems

DT, $r = 1\%$			DT, peaks			Boosting, $r = 1\%$			Boosting, peaks		
m/z	Imp. %	R_p	m/z	Imp. %	R_p	m/z	Imp. %	R_p	m/z	Imp. %	R_p
RA dataset											
1054–1064	22.96	3	2924	22.78	1	1054–1064	5.65	3	2924	11.38	1
4275–4318	15.22	168	2778	9.36	21	2913–2942	3.99	1	4538	7.37	44
5336–5390	13.08	8	9371	9.08	113	4587–4633	3.96	52	10441	5.22	31
15324–15478	8.72	4	10441	8.01	31	6144–6206	3.78	44	4825	4.42	92
1922–1941	8.64	141	15485	7.98	67	4318–4362	3.47	2	2778	4.20	21
2942–2972	5.12	36	9509	6.51	88	4825–4873	3.36	113	5686	3.11	5
2330–2354	4.98	298	8145	6.36	8	7588–7665	3.34	10	6669	2.29	43
1737–1754	4.81	299	4538	6.16	44	15324–15478	2.05	4	5914	2.22	54
3865–3903	4.35	23	14667	5.58	130	4275–4318	2.00	168	1034	1.95	24
2403–2427	3.64	269	3748	4.65	58	5901–5960	1.96	18	4134	1.94	42
IBD dataset											
5177–5230	32.23	1	4213	27.84	1	5177–5230	11.93	1	4213	13.59	1
9951–10052	9.14	217	13886	7.18	124	5612–5668	6.49	4	3068	6.42	2
4189–4232	8.86	8	3218	6.79	59	4189–4232	5.50	8	4238	5.84	8
5612–5668	7.17	4	4289	6.70	66	3063–3095	3.38	2	4289	4.80	66
13722–13860	5.16	188	5255	6.24	34	4275–4318	3.29	162	24097	3.76	3
6332–6396	5.14	9	3320	5.71	151	24048–24290	3.20	3	3163	2.93	86
4275–4318	4.72	162	5073	4.70	115	3983–4023	2.44	17	23197	2.85	10
2022–2043	3.53	265	7773	4.34	4	23807–24048	2.19	5	5753	2.55	7
16111–16274	3.39	72	24332	4.20	15	4728–4776	1.84	13	1945	1.83	27
11571–11687	3.09	336	3965	4.13	137	8734–8822	1.64	67	1741	1.65	11

Table 6. Best results with attribute selection by boosting ($r = 1\%$)

N	RA dataset				IBD dataset			
	Method	Sensitivity	Specificity	Err.	Method	Sensitivity	Specificity	Err.
1	ET	58.82 (40/68)	74.64 (103/138)	30.58	ET	77.08 (185/240)	84.10 (201/239)	19.42
2	RF	63.24 (43/68)	84.06 (116/138)	22.82	ET	82.08 (197/240)	81.59 (195/239)	18.16
3	BA	80.88 (55/68)	86.96 (120/138)	15.05	ET	80.00 (192/240)	89.12 (213/239)	15.45
4	DT	85.29 (58/68)	85.51 (118/138)	14.56	ET	83.75 (201/240)	84.94 (203/239)	15.66
5	ET	85.29 (58/68)	87.68 (121/138)	13.11	ET	82.92 (199/240)	83.68 (200/239)	16.70
10	BO	88.24 (60/68)	92.75 (128/138)	8.74	RF	82.92 (199/240)	87.87 (210/239)	14.61
15	BO	86.76 (59/68)	94.93 (131/138)	7.77	RF	84.58 (203/240)	90.38 (216/239)	12.53
20	BO	91.18 (62/68)	94.93 (131/138)	6.31	RF	84.58 (203/240)	92.05 (220/239)	11.69
25	BO	89.71 (61/68)	94.93 (131/138)	6.80	RF	89.58 (215/240)	92.05 (220/239)	9.19
50	ET	92.65 (63/68)	92.75 (128/138)	7.28	ET	90.00 (216/240)	94.98 (227/239)	7.52
75	BO	83.82 (57/68)	95.65 (132/138)	8.25	ET	90.83 (218/240)	95.82 (229/239)	6.68
100	BO	86.76 (59/68)	95.65 (132/138)	7.28	RF	89.58 (215/240)	95.40 (228/239)	7.52
ALL	BO	83.82 (57/68)	94.93 (131/138)	8.74	ET	88.33 (212/240)	91.63 (219/239)	10.02

of attributes (respectively, 20 and 75 attributes), while the one-standard error rule retains respectively 15 and 50 attributes only. Also, the most important improvement occurs already with a much smaller number of attributes (on RA, from 5 to 10 attributes; on IBD, ∼15–20 attributes).

Figure 2 shows further information. The curves called 'global selection' draw the evolution of the error with boosting while using importance ranking (the horizontal lines correspond to $\mathrm{Err}^* + \sigma^*$ bars used for biomarker selection). Two further pairs of curves obtained when the attributes are introduced in random order averaged over the leave-one-out folds ('Random') and when they are introduced according to the P-values are also shown. We see that for smaller numbers of attributes, P-values are competitive with boosting, but boosting scores better when N increases. The random order gives significantly worse results.

As already stated, since the attribute ranking uses internal cross-validation on the whole dataset, the optimal error rates given above might be optimistic (Ambroise and McLachlan, 2002). In the absence of a sufficiently large independent test set, we thus have used the external cross-validation procedure proposed in (Ambroise and McLachlan, 2002), to yield unbiased estimates. For each run of the external leave-one-out method, this consists of using boosting on all data but the removed patient, ranking the attributes with this model, and building a new model using the first N of these latter

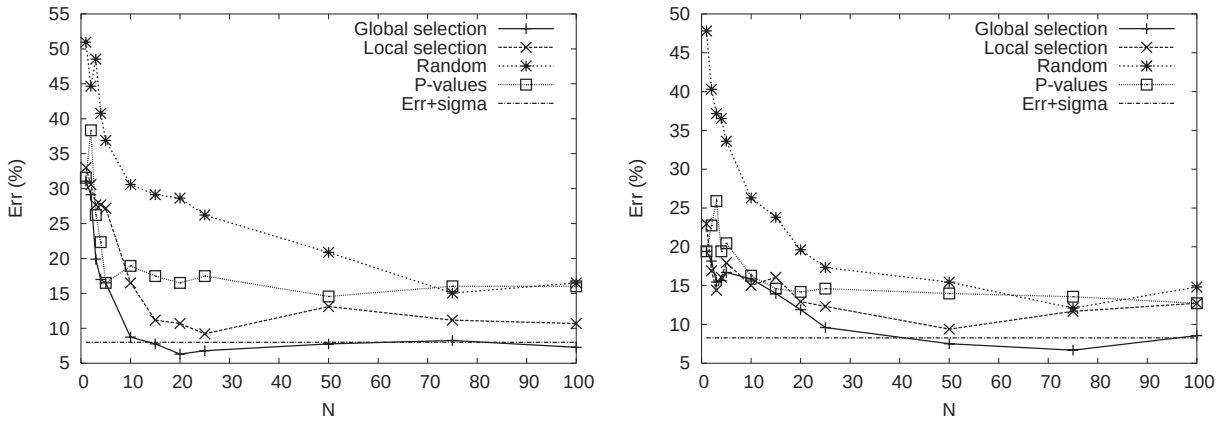

Fig. 2. Learning curves with boosting, left on the RA dataset, right on the IBD dataset.

and testing it on the removed patient. This procedure applied for increasing values of N gives the curve labeled 'local selection' in Figure 2. We observe that these errors are indeed higher than the errors obtained with internal leave-one-out procedure used to generate the 'global selection'. However, their general trend is similar and on both problems the optimal numbers of attributes is also close to the one obtained by the global selection procedure.

Hence, we deem that the procedure of Section 3.3 indeed selects a relevant set of biomarkers. We also believe that ML models using these reduced sets could possibly provide more reliable models than those using the whole set of candidate attributes, although this is not demonstrable from our experiments. We also notice that our observations are consistent with the conclusions drawn in Ambroise and McLachlan (2002) with another medical instrumentation (microarray) and another ML algorithm (SVMs).

4.4 Aggregation of replica classifications

In regard to the IBD problem, in this section we investigate the use of a certain number of replica MS measurements (say R repeated MS analyses of the serum collected from one patient). To this end, a model for classifying individual spectra is built using the same approach as in the previous experiment. Then, to predict the status of a given patient, the R spectra corresponding to this patient are first classified by this model, and the patient is diagnosed as suffering from IBD if at least M of his measurements were classified as IBD. In principle, by adjusting the value of M ($1 \leq M \leq R$) it is possible to either favor classifications in the 'IBD' class or in the 'non-IBD' class, and hence to provide different tradeoffs between sensitivity and specificity. Using smaller values of M maximizes sensitivity and thus it may be interesting, for example, to screen a population for a rapidly evolving and lethal disease. On the other hand, larger values of M allow the decrease of the number of false positives and might be preferred, for example, to decide the application of a intensive or expensive treatment.

Table 7 shows the results obtained in this way for increasing values of M ($R = 4$). We observe that this very simple approach indeed improves with respect to the results of Table 3: for $M = 2$, both sensitivity and specificity increase (and error rate decreases from 10.44 to 7.5%), while for $M = 1$ sensitivity further increases (at the price of a decrease in specificity), and for $M = 3$ or 4, specificity increases (at the price of a decrease in sensitivity).

Table 7. Accuracy/patient (IBD problem), $r = 1\%$ and boosting

M	Sensitivity	Specificity	Err.
1	93.33 (56/60)	83.33 (50/60)	11.66
2	91.66 (55/60)	93.33 (56/60)	7.50
3	86.66 (52/60)	93.33 (56/60)	10.00
4	76.66 (46/60)	98.33 (59/60)	12.50

5 CONCLUSIONS

In this paper, we have proposed a systematic and flexible methodology to support the analysis and knowledge extraction from proteomic MS datasets. This framework relies on a toolbox of generic supervised ML algorithms comprising decision tree induction and several decision tree based ensemble methods, which are combined with pre- and post-processing stages. These latter complete the generic tools in order to handle effectively proteomic MS datasets obtained with various data acquisition devices and strategies and to extract from such datasets, both accurate decision criteria and interpretable information, which can directly be exploited to identify biomarkers for clinical proteomics. From an application point of view the framework remains as general as possible, while at the same time giving superior results to the standard pre- and post-processing techniques used in these applications (peak-detection and P-values based biomarker selection). We also notice from the computational point of view that the overall procedure, including biomarker selection and leave-one-out cross-validation, runs in a few minutes on a standard workstation.

The clinical research which drove the development of our work concerns two actual problems, namely the diagnosis of RA and of IBD. In both cases, the same framework was applied and gave very promising results both for the induction of predictive models and for the identification of biomarkers. This highlights clearly the flexibility of the approach.

Further research will aim at applying this methodology to other problems and other proteomic and/or genomic data acquisition schemes. While this paper focuses on the extraction of accurate predictive models and biomarkers, another interesting direction for future research is the extraction of interpretable rules from such data. In this context, the method proposed in (Li *et al*., 2003) might be an interesting complement to the methods presented here.

ACKNOWLEDGEMENTS

P.G. is a Postdoctoral Researcher and M.F. and M.-P.M. are Senior Research Assistants at the National Fund for Scientific Research (FNRS, Belgium).

REFERENCES

Ambroise,C. and McLachlan,G. (2002) Selection bias in gene extraction on the basis of microarray gene-expression data. *Proc. Natl Acad. Sci. USA*, **99**, 6562–6566.

Bauer,E. and Kohavi,R. (1999) An empirical comparison of voting classification algorithms: bagging, boosting, and variants. *Machine Learning*, **36**, 105–139.

Breiman,L. (1996) Bagging predictors. *Machine Learning*, **24**, 123–140.

Breiman,L. (2001) Random forests. *Machine learning*, **45**, 5–32.

Breiman,L., Friedman,J., Olsen,R. and Stone,C. (1984) *Classification and Regression Trees*. Wadsworth International, CA, USA.

Dietterich,T.G. (2000) An experimental comparison of three methods for constructing ensembles of decision trees: bagging, boosting, and randomization. *Machine Learning*, **40**, 139–157.

Freund,Y. and Schapire,R. (1995) A decision-theoretic generalization of on-line learning and an application to boosting. In *Proceedings of the second European Conference on Computational Learning Theory*, Springer-Verlag, London, pp. 23–27.

Fung,E.T. and Enderwick,C. (2002) Proteinchip clinical proteomics: computational challenges and solutions. *Computational Proteomics*, **3**, S34–S41.

Hastie,T., Tibshirani,R. and Friedman,J. (2001) *The Elements of Statistical Learning: Data Mining, Inference and Prediction*. Springer.

Izmirlian,G. (2004) Application of the random forest classification algorithm to a SELDI-TOF proteomics study in the setting of a cancer prevention trial. *Ann. NY Acad. Sci.*, **1020**, 154–174.

Jong,K., Marchiori,E. and van der Vaart,A. (2004) Analysis of proteomic pattern data for cancer detection. In Raidl,G.R. *et al*. (eds), *Applications of Evolutionary Computing: EvoWorkshops 2004*, Springer, pp. 41–50.

Li,J. *et al*. (2003) Discovery of significant rules for classifying cancer diagnosis data. *Bioinformatics*, **19** (Suppl. 2), ii93–ii102.

Liu,H. *et al*. (2002) A comparative study on feature selection and classification methods using gene expression profiles and proteomic patterns. *Genome inform. Ser. Workshop Genome Inform.*, **13**, 51–60.

Pusch,W. *et al*. (2003) Mass spectrometry-based clinical proteomics. *Pharmacogenomics*, **4**, 463–476.

Qu,Y. *et al*. (2002) Boosted decision tree analysis of surface-enhanced laser desorption/ionization mass spectral serum profiles discriminates prostate cancer from noncancer patients. *Clin. Chem.*, **48**, 1835–1843.

Quinlan,J. (1986) *C4.5: Programs for Machine Learning*. Morgan Kaufmann, San Mateo.

Rai,A. *et al*. (2002) Proteomic approaches to tumor marker discovery. *Arch. Pathol. Lab. Med.*, **126**, 1518–1526.

Wehenkel,L. (1998) *Automatic Learning Techniques in Power Systems*. Kluwer Academic, Boston.

Wu,B. *et al*. (2003) Comparison of statistical methods for classification of ovarian cancer using mass spectrometry data. *Bioinformatics*, **19**, 1636–1643.