Robust SVM-based biomarker selection with noisy mass spectrometric proteomic data

Elena Marchiori[1], Connie R. Jimenez[2], and Mikkel West-Nielsen, Niels H.H. Heegaard[3]

[1] Department of Computer Science, Vrije Universiteit Amsterdam, The Netherlands, elena@cs.vu.nl
[2] Department of Molecular and Cellular Neurobiology, Vrije Universiteit Amsterdam, The Netherlands, connie.jimenez@falw.vu.nl
[3] Department of Autoimmunology, Statens Serum Institut, Copenhagen, Denmark, MWN@ssi.dk, NHE@ssi.dk

Abstract. Computational analysis of mass spectrometric (MS) proteomic data from sera is of potential relevance for diagnosis, prognosis, choice of therapy, and study of disease activity. To this aim, feature selection techniques based on machine learning can be applied for detecting potential biomarkes and biomaker patterns. A key issue concerns the interpretability and robustness of the output results given by such techniques. In this paper we propose a robust method for feature selection with MS proteomic data. The method consists of the sequentail application of a filter feature selection algorithm, RELIEF, followed by multiple runs of a wrapper feature selection technique based on support vector machines (SVM), where each run is obtained by changing the class label of one support vector. Frequencies of features selected over the runs are used to identify features which are robust with respect to perturbations of the data. This method is tested on a dataset produced by a specific MS technique, called MALDI-TOF MS. Two classes have been artificially generated by spiking. Moreover, the samples have been collected at different storage durations. Leave-one-out cross validation (LOOCV) applied to the resulting dataset, indicates that the proposed feature selection method is capable of identifying highly discriminatory proteomic patterns.

1 Introduction

Feature selection (FS) for classification can be formulated as a combinatorial optimization problem: finding the feature set maximizing the predictive performance of the classifier trained from these features. FS is a major research topic in supervised learning and data mining [10, 16, 12]. For the sake of the learning performance, it is highly desirable to discard irrelevant features prior to learning, especially when the number of available features significantly outnumbers the number of samples, like in biomedical studies. Because of its computational intractability, the FS problem has been tackled by means of heuristic algorithms based on statistics and machine learning [10, 20, 22]. Biological experiments from

laboratory technologies like microarray and mass spectrometry techniques, generate data with a very high number of variables (features), in general much larger than the number of samples. Therefore FS provides a fundamental step in the analysis of such type of data [27]. Ideally one would like to detect potential biomarkers and biomarker patterns, that both highly discriminate diseased from healthy samples and are biological interpretable. However, as substantiated in recent publications like [19, 3, 21], reliability and reproducibility of results depend on the particular way samples are handled [26], on the instability of the laboratory technology, as well as on the specific techniques employed in the computational analysis.

In this paper we consider FS for classification with MS proteomic data from sera. Various machine learning and statistical techniques for feature selection have been applied to proteomic data, like [15, 17, 8, 13, 5–7], in order to detect potential tumor biomarkers for (early) cancer diagnosis (clinical proteomics). A summary of actual challenges and critical assessment of clinical proteomics can be found, e.g., in [21].

Here we propose a new method for FS with MS proteomic data. The goal is to identify potential biomarker patterns that not only highly discriminate diseased and healthy samples, but also are robust with respect to perturbation of the data. The method consists of three main steps. First, a popular filter feature selection algorithm, RELIEF, is used as pre-processing in order to reduce the number of considered features. Next, multiple runs of linear SVM are considered, where at each run a perturbed training set is used, obtained by changing the class label of one support vector. Each run generates a large subset of selected features. The frequency (over the runs) of selection of the features is used to choose the most robust ones, namely those with highest frequency. Finally, the resulting features are transformed into feature intervals, by considering the ordering of the features, where neighbour features refer to peptides of similar masses.

The method generates a subset of feature intervals, where both the number of intervals and features are automatically selected. These intervals describe potential biomarker patterns.

We analyze experimentally the performance of the method on a real-life dataset with controlled insertion of noisy samples (long storage time samples) and "relevant" features (spiked molecules) [26]. The results indicate that the method performs robust feature selection, by selecting features corresponding to m/z measurements near to the (average of m/z values of the peak of the) spiked molecules, and by misclassifying only 1 noisy sample (with long storage time).

2 Background

This section describes in brief the Machine Learning techniques we use in the proposed feature selection method.

2.1 Linear Support Vector Machines

In linear SVM-based binary classification [25, 2], samples of two classes are linearly separated by means of a maximum margin hyperplane, that is, the hyperplane that maximizes the sum of the distances between the hyperplane and its closest points of each of the two classes (the margin). When the classes are not linearly separable, a variant of SVM, called soft-margin SVM, is used. This SVM variant penalizes misclassification errors and employs a parameter (the soft-margin constant C) to control the cost of misclassification.

Training a linear SVM classifier amounts to solving the following constrained optimization problem:

$$min_{\mathbf{w},b,\xi_k} \frac{1}{2}||\mathbf{w}||^2 + C\sum_{i=1}^{m}\xi_i \quad \text{s.t.} \quad \mathbf{w}\cdot\mathbf{x}_i + b \geq 1 - \xi_i$$

with one constraint for each training sample $\mathbf{x}_i$. Usually the dual form of the optimization problem is solved:

$$min_{\alpha_i} \frac{1}{2}\sum_{i=1}^{m}\sum_{j=1}^{m}\alpha_i\alpha_j y_i y_j \mathbf{x}_i\cdot\mathbf{x}_j - \sum_{i=1}^{m}\alpha_i$$

such that $0 \leq \alpha_i \leq C$, $\sum_{i=1}^{m}\alpha_i y_i = 0$. SVM requires $O(m^2)$ storage and $O(m^3)$ to solve.

The resulting decision function $f(\mathbf{x}) = \mathbf{w}\cdot\mathbf{x} + b$ has weight vector $\mathbf{w} = \sum_{k=1}^{m}\alpha_k y_k \mathbf{x_k}$. Samples $\mathbf{x}_i$ for which $\alpha_i > 0$ are called *support vectors*, since they uniquely define the maximum margin hyperplane. Samples with $\alpha_i - C$ are misclassified.

Maximizing the margin allows one to minimize bounds on generalization error. Because the size of the margin does not depend on the input dimension, SVM are robust with respect to data with high number of features. However, SVM are sensitive to the presence of (potential) outliers, (cf. [11] for an illustrative example) due to the regularization term for penalizing misclassification (which depends on the choice of C).

2.2 Variable Selection Techniques

One can distinguish three main approaches for feature ranking/selection: wrapper, filter and embedded.

- In the wrapper approach features are selected/ranked by taking into account their contribution to the performance of a given type of classifier (e.g., SVM).
- In the filter approach the selection/ranking of features is not (directly) biased towards the performance of a specific type of classifier, but is based on an evaluation criterion for quantifying how well feature (subsets) discriminate the two classes.
- Finally, in the embedded approach feature selection/ranking is part of the training procedure of a classifier, like in decision trees.

The effectiveness of these approaches depends on the application domain. The wrapper approach is favoured in many works since the selection of the features is directly linked to the performance of a chosen type of classifier. On the other hand, algorithms based on the filter approach, like RELIEF, are in general more efficient and have been successfully applied to real life domains [28]. Techniques based on the embedded approach provide a global approach, where feature selection is a by-product of the training algorithm for classification.

SVM Feature Selection (SVMFS)

The weights w_i of a linear SVM classifier provide information about feature relevance, where a bigger weight value implies higher feature relevance. In this paper a feature x_i is scored by means of w_i^2 [11]. Feature weights, obtained by training a linear SVM on the training set, are used in a scoring function for ranking features as described above. The algorithm for feature selection based on SVM is illustrated below (in pseudo-code).

```
SVMFS
%input: training set X, number of features
              to be selected M
%output: subset Selected of M features
train linear classifier with SVM on X;
score features using the squared value of
      the weights of the classifier;
Selected = M features with highest score;
return Selected;
```

RELIEF

RELIEF [23, 14] is a filter-based feature ranking algorithm that assigns a score to features based on how well the features separate training samples from their nearest neighbours from the same and from the opposite class.

The algorithm constructs iteratively a weight vector, which is initially equal to zero. At each iteration, RELIEF selects one sample, adds to the weight the difference between that sample and its nearest sample from the opposite class (called nearest miss), and subtracts the difference between that sample and its nearest neighbour from the same class (called nearest hit). The iterative process terminates when all training samples have been considered. The resulting weight of a feature is divided by its range of values (computed using only the training set). Subsampling can be used to improve efficiency in case of a large training set. The pseudo-code of the RELIEF algorithm used in our experiments is given below.

```
RELIEF
%input: training set X
%output:  Ranking of features
```

```
nr_feat = total number of features;
weights = zero vector of size nr_feat;
for all samples exa in training set do
    hit(exa) = nearest neighbour of exa
             from same class;
    miss(exa) =  nearest neighbour of exa
             from opposite class;
    weights =  weights-abs(exa-hit(exa))+
             abs(exa - miss(exa));
end;
scale each weight using range of corresponding m/z
value intensity over the training set;
Ranking =  obtained by sorting weights
           in decreasing order;
return Ranking;
```

3 Mass Spectrometric Proteomic Data

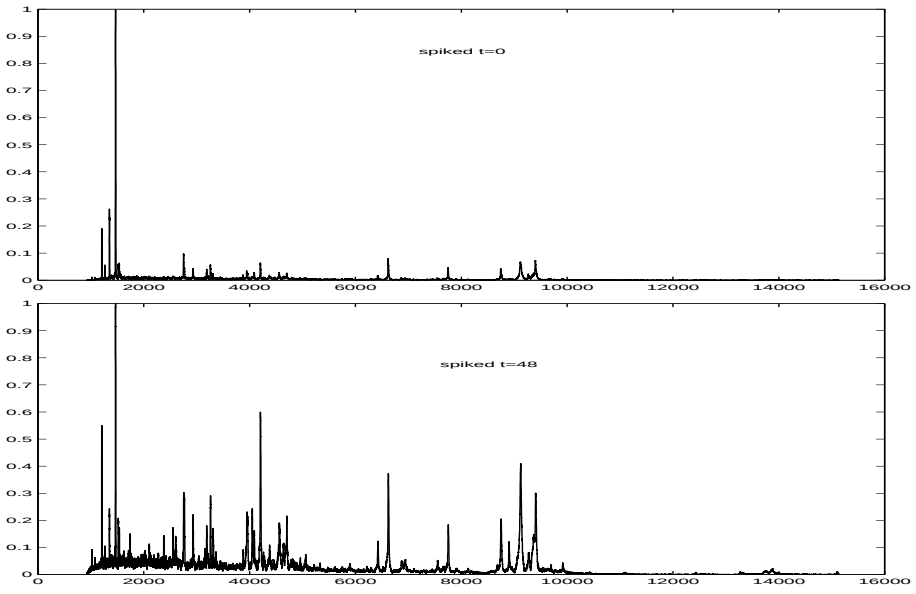

Fig. 1. A MALDI-TOF MS spiked sample for one person at storage duration time t=0 (top) and t=48 (bottom): x-axis contains (identifiers of) the m/z values of peptides and the y-axis their concentration.

The MS proteomic dataset here considered is obtained by matrix-assisted laser desorption/ionization time-of-flight mass spectrometry (MALDI-TOF MS), a recent laboratory technology which offers protein profiling at high resolution and throughput. It measures the relative abundance of ionisable low molecular weight peptides in complex mixtures, like serum (cf. e.g. [4]). Because it is relatively inexpensive and noninvasive, it is considered a promising new technology for classifying disease status and for tumor biomarker detection.

MALDI-TOF MS technology produces a graph of the relative abundance of ionized peptides (y-axis) versus their mass-to-charge (m/z) ratios (x-axis). (see Figure 1) The m/z ratios are proportional to the peptide masses, but the technique is not able to identify individual peptides, because different peptides may have the same mass and because of limitations in the m/z resolution.

Given proteomic profiles from healthy and diseased individuals, the goal is to build a classifier for tumor diagnostics and to identify those proteins that are potentially involved in the disease.

The dataset considered in this study consists of 96 samples obtained from human sera of 8 adult persons. Spiking has been used to produce two classes, and 6 different storage durations (t=0, 1, 4, 8, 24, 48 hours) have been used to produce noisy data. Each profile contains 22572 m/z measurements. Adjacent m/z measurements correspond to peptides with similar mass versus charge. Thus the ordering on the x-axis has a biological meaning. This ordering will be used in the FS method described in the next section.

The complete procedure for generating such data is described in detail in [26], where this and other datasets have been analyzed for the first time. Calibration standards containing seven peptides and four proteins were used as artificial markers (Bruker Daltonik) and consisted of the following molecules with average molecular masses given in parentheses: angiotensin II (1047.20), angiotensin I (1297.51), substance P (1348.66), bombesin (1620.88), ACTH clip 1-17 (2094.46), ACTH clip 18-39 (2466.73), somatostatin 28 (3149.61), insulin (5734.56), ubiquitin I (8565.89), and cytochrome c (6181.05) and myoglobin (8476.77). However, no signal was recovered for the following four spiked molecules, possibly due to losses during the laboratory sample processing procedure: substance P (1348.6), ACTH clip 1-17 (2094.46), cytochrome c (6181.05), and myoglobin (8476.77) [26].

In [18], we used this dataset for comparing the performance of two popular feature selection techniques, RELIEF and Recursive Feature Selection with linear SVM, and the performance of two classification techniques, SVM and K nearest neighbours. The results indicated that, in general, better predictive performance does not correspond to better biological interpretability of the selected features (m/z values).

4 The Method

We propose a FS method, consisting of three steps, called Filter, Wrapper, and Interval step (FWI). The method is illustrated below in pseudo-code.

```
FWI
%input: training set X
%number M of features to be selected by RELIEF
%number N (N<M) of features to be selected by SVMFS
%SVM parameter C
%output:  Set Int of feature intervals

%FILTER step:
F = M features selected with RELIEF;

%WRAPPER step:
SV = set of support vectors obtained by training
     SVM on X;
for x in SV
 T = X with label of x changed;
 F(x) = N features selected by SVMFS applied to T;
end;
count = maximum number of times that a feature
        occurs in the sequence of F(x), x in SV;
W = features occurring count times  in the sequence
    of F(x), x in SV;

%INTERVAL step:
Cl = {C1, .., Cn} clustering of W, with
               Ci = (w(1),.., w(ni))
               w(1)<..< w(ni)
               s.t. idx(w(j+1))-idx(w(j)) <= 2
               for all j in [1..ni];
Int = {Int_1, .., Int_n} intervals from Cl, with
    Int_i= {w in Features s.t. w >= min(Ci)
             and w<= max(Ci)} for i in [1,n];
return Int;
```

Let us explain a bit in more detail the steps performed by FWI.

- FWI starts by skimming the number of features, by applying the Filter (F) step. Here RELIEF is employed in order to select M features. In the F step one typically retains about M=5000 m/z measurements from the initial 22572.
- In the Wrapper (W) step, robust wrapper based feature selection is performed using the features that passed the Filter selection. In the Wrapper (W) step, the support vectors of SVM trained on all the features are used for perturbing the data. More precisely, multiple runs of SVMFS are performed, where at each run the class label of one support vector is changed. Each run generates a set of N features (typical value N=1000). The resulting sequence of feature sets is then considered. The maximum number *count* of times a

feature occurs in the sequence is computed, and all features occurring *count* times in the sequence are selected.
- Finally, in the Interval (I) step, the selected m/z features are segmented as follows. The sequence of features in W, ordered by m/z values, is segmented in such a way that each pair of consecutive features in one segment C_i is at most two positions apart in the sequence of all features ordered by m/z values (in the above pseudo-code $idx(w)$ gives the position of feature w in the ordered sequence of all features). Finally each sequence C_i generates one interval I_i containing all m/z measurements between the first and last element of C_i.

The F step can be viewed as a kind of preprocessing, while steps W and I are heuristics for overcoming the problem of variability due to perturbations of the data that can possibly originate from noise. The method requires the user to specify 3 parameters, in particular the sizes M and N of the feature subsets selected by RELIEF and SVMFS, respectively. However, note that the values of M and N can be chosen to be fairly big, and the final smaller size of the feature subset selected by FWI is determined automatically by the algorithm.

5 Numerical Experiments

In order to assess the effectiveness of the modules of FWI, we consider the following four algorithms:

1. Wrapper feature selection (W), obtained by applying the W step of the FWI.
2. Wrapper Interval (WI), obtained by applying steps W followed by I.
3. Filter Wrapper (FW), obtained by applying steps F followed by W.
4. The complete Filter Wrapper Interval algorithm FWI.

Because of the small size of the data, LOOCV is used for comparing the performance of the four algorithms (cf., e.g., [9]). At each leave-one-out run, all but one element of the data is used as training set, and the left-out element is used for testing the predictive performance of the resulting classifier. Observe that the 96 samples of the considered dataset are not independent one of the other, as required for a correct application of LOOCV, because they are generated from 8 persons, and neither the 6 different storage times nor the spiking guarantee the production of independent samples. Nevertheless, the corresponding bias introduced in the LOOCV procedure affects the results of each algorithm, hence the results can be used for comparing the performance of the algorithms. However, such bias possibly affects the estimation of the generalization error.

Table 1 summarizes LOOCV performance results of the experiments. We use accuracy, sensitivity and specificity as quality measures for comparing the algorithms. Other measures, like AUC (Area Under the ROC Curve), can be used. As illustrated e.g. in [1], there is a good agreement between accuracy and AUC as to the ranking of the performance of the classification algorithms.

The results indicate that there is an improvement in predictive performance of the four algorithms, with best accuracy achieved by FWI.

Method	Accuracy	Sensitivity	Specificity
W	0.9479 (0.2234)	0.9375 (0.2446)	0.9583 (0.2019)
WI	0.9583 (0.2019)	0.9583 (0.2019)	0.9583 (0.2019)
FW	0.9792 (0.1436)	1.0000 (0.0000)	0.9583 (0.2019)
FWI	0.9896 (0.1021)	1.0000 (0.0000)	0.9792 (0.1443)

Table 1. Results: LOOCV sensitivity, specificity and accuracy (with standard deviation between brackets).

The misclassified samples over all the LOOCV runs have storage time equal to 24 or 48 hours, indicating that longer storage time affects negatively classification of proteomic samples. Algorithm W misclassifies a total of 5 samples, of which 2 spiked at t= 48, 1 spiked at t=24, and 2 normal at t=24. WI improves by correcly classifying the one spiked sample with t=24. Furthermore, FW misclassifies a total of 2 samples, the normal sample at t=24 like W and WI, and one normal at t=48, while it correctly classifies all spiked samples. Finally, FWI only misclassifies a total of 1 sample, the normal one at t=24, like W, WI, and FW.

Each algorithm selects about 120 features at each run, which are distributed (in the Interval step) in about 15 clusters.

We further analyze the results of FWI. Figure 2 shows m/z measurements versus the number of times they are selected over all LOOCV. On the x-axis the location of the spiked molecules is indicated by circles. The plot indicates that m/z measurements in proximity of spiked molecules are more often selected over the LOOCV runs, except for m/z measurements in the neighbourhood of 4000 and 5000, which do not correspond to m/z measurements of spiked molecules. In the absence of additional information (e.g. tandem mass spectra yielding sequence tags) it is difficult to know what these peak values represent. One possibility is that the higher molecular weight spiked molecules are partially degraded in serum, and these peaks are proteolytically cleaved peptides from larger proteins (due to large storage time at room temperature) in the sample itself. However, this possibility has not yet been examined in depth. Figure 3 shows a typical set of m/z measurements generated by FWI, and the mean value of the intensities of spiked and normal samples for the selected m/z measurements.

In conclusion, results indicate that FWI performs robust m/z selection, where the selected features are close to the spiked molecules, and the misclassification error is close to zero, with misclassification of only noisy (that is high storage temperature) samples.

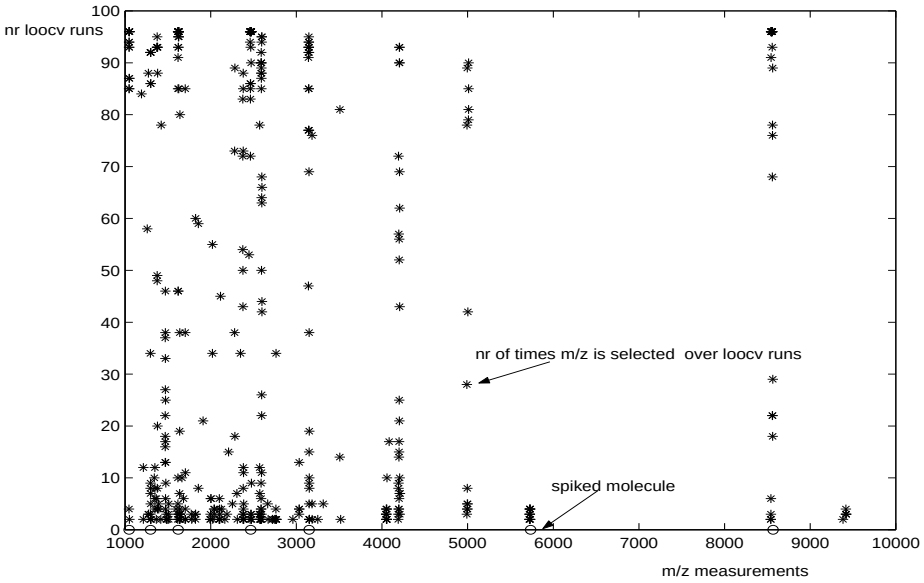

Fig. 2. Number of m/z measurements selected over LOOCV runs.

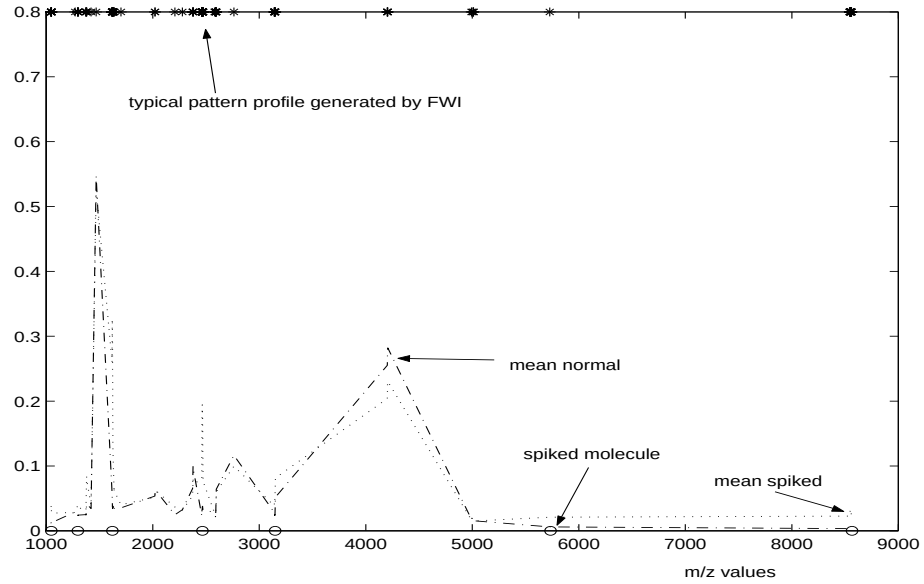

Fig. 3. A typical m/z selection generated by FWI, the corresponding values of the mean spiked and normal profile at the selected m/z values, and the spiked molecules.

6 Conclusion

We have proposed a method for robust feature selection with MS proteomic data. The method can be considered as a small step towards the development of a feature selection methodology addressing the specific issues of the underlying laboratory technology. In particular, in this paper we addressed the issue of performing robust feature selection in the presence of noisy samples which perturb the data and negatively affect sample classification. The W and I steps of the proposed FWI method provide heuristics for tackling this problem.

This issue is related to broader questions about reproducibility and validity of results in the discovery-based "omics" research [21, 24]. In a special session on genomics of a recent issue of *Science* an essay entitled "Getting the noise out of gene arrays" noted that "[t]housands of papers have reported results obtained using gene array ... But are these results reproducible?" [19]. A controversy about reprodicibility and validity of results from MS proteomic data is ongoing [3, 21] and the path for achieving such ambitious goals appears still long.

Acknowledgments

We would like to thank the anonymous referees for their constructive comments, and Jan Rutten for stimulating discussions.

References

1. A. P. Bradley. The use of the area under the ROC curve in the evaluation of machine learning algorithms. *Pattern Recognition*, 30(6):1145–1159, 1997.
2. N. Cristianini and J. Shawe-Taylor. *Support Vector machines*. Cambridge Press, 2000.
3. E.P. Diamandis. Analysis of serum proteomic patterns for early cancer diagnosis: Drawing attention to potential problems. *Journal of the National Cancer Institute*, 96(5):353–356, 2004.
4. H.J. Issaq et al. SELDI-TOF MS for diagnostic proteomics. *Anal. Chem.*, 75(7):148A–155A, 2003.
5. Petricoin E.F. et al. Serum proteomic patterns for detection of prostate cancer. *Journal of the National Cancer Institute*, 94(20):1576–1578, 2002.
6. Petricoin E.F. et al. Use of proteomic patterns in serum to identify ovarian cancer. *The Lancet*, 359(9306):572–7, 2002.
7. Qu Y. et al. Boosted decision tree analysis of surface-enhanced laser desorption/ionization mass spectral serum profiles discriminates prostate cancer from noncancer patients. *Clin. Chem*, 48(10):1835–43, 2002.
8. Zhu W. et al. Detection of cancer-specific markers amid massive mass spectral data. *PNAS*, 100(25):14666–14671, 2003.
9. T. Evgeniou, M. Pontil, and A. Elisseeff. Leave one out error, stability, and generalization of voting combinations of classifiers. *Mach. Learn.*, 55(1):71–97, 2004.
10. I. Guyon and A. Elisseeff. An introduction to variable and feature selection. *Machine Learning*, 3:1157–1182, 2003. Special Issue on variable and feature selection.

11. I. Guyon, J. Weston, S. Barnhill, and V. Vapnik. Gene selection for cancer classification using support vector machines. *Mach. Learn.*, 46(1-3):389–422, 2002.
12. George H. John, Ron Kohavi, and Karl Pfleger. Irrelevant features and the subset selection problem. In *International Conference on Machine Learning*, pages 121–129, 1994.
13. K. Jong, E. Marchiori, M. Sebag, and A. van der Vaart. Feature selection in proteomic pattern data with support vector machines. In *IEEE Symposium on Computational Intelligence in Bioinformatics and Computational Biology*, 2004.
14. K. Kira and L. A. Rendell. The feature selection problem: Traditional methods and a new algorithm. In *Tenth National Conference on artificial intelligence*, pages 129–134, 1992.
15. J. Li, Z. Zhang, J. Rosenzweig, Y.Y. Wang, and D.W. Chan. Proteomics and bioinformatics approaches for identification of serum biomarkers to detect breast cancer. *Clinical Chemistry*, 48(8):1296–1304, 2002.
16. H. Lie and editors H. Motoda. *Feature Extraction, Construction and Selection: a Data Mining Perspective*. International Series in Engineering and Computer Science. Kluwer, 1998.
17. H. Liu, J. Li, and L. Wong. A comparative study on feature selection and classification methods using gene expression profiles and proteomic patterns. *Genome Informatics*, 13:51–60, 2002.
18. E. Marchiori, N.H.H. Heegaard, M. West-Nielsen, and C.R. Jimenez. Feature selection for classification with proteomic data of mixed quality. In *Proceedings of the 2005 IEEE Symposium on Computational Intelligence in Bioinformatics and Computational Biology*, pages 385–391, 2005.
19. E. Marshall. Getting the noise out of gene arrays. *Science*, 306:630–631, 2004. Issue 5696.
20. Il-Seok Oh, Jin-Seon Lee, and Byung-Ro Moon. Local search-embedded genetic algorithms for feature selection. In *16 th International Conference on Pattern Recognition (ICPR'02)*. IEEE Press, 2002.
21. D.F. Ransohoff. Lessons from controversy: Ovarian cancer screening and serum proteomics. *Journal of the National Cancer Institute*, 97:315–319, 2005.
22. M.L. Raymer, W.F. Punch, E.D. Goodman, L.A. Kuhn, and A.K. Jain. Dimensionality reduction using genetic algorithms. *IEEE Transactions on Evolutionary Computation*, 4(2):164–171, 2000.
23. L. A. Rendell and K. Kira. A practical approach to feature selection. In *International Conference on machine learning*, pages 249–256, 1992.
24. Michiels S., Koscielny S., and Hill C. Prediction of cancer outcome with microarrays: a multiple random validation strategy. *The Lancet*, 365(9458):488–92, 2005.
25. V.N. Vapnik. *Statistical Learning Theory*. John Wiley & Sons, 1998.
26. M. West-Nielsen, E.V. Hogdall, E. Marchiori, C.K. Hogdall, C. Schou, and N.H.H. Heegaard. Sample handling for mass spectrometric proteomic investigations of human sera. *Analytical Chemistry*, 11(16):5114–5123, 2005.
27. E.P. Xing. Feature selection in microarray analysis. In *A Practical Approach to Microarray Data Analysis*. Kluwer Academic, 2003.
28. L. Yu and H. Liu. Feature selection for high-dimensional data: A fast correlation-based filter solution. In *ICML*, pages 856–863, 2003.