Open access • Journal Article • DOI:10.1002/MAS.20072

Processing and classification of protein mass spectra — Source link

Melanie Hilario, Alexandros Kalousis, Christian Pellegrini, Markus Müller

Institutions: University of Geneva

Published on: 01 May 2006 - Mass Spectrometry Reviews (John Wiley & Sons, Ltd)

Topics: Sample collection

Processing and classification of protein mass spectra

HILARIO, Mélanie, *et al.*

Abstract

Among the many applications of mass spectrometry, biomarker pattern discovery from protein mass spectra has aroused considerable interest in the past few years. While research efforts have raised hopes of early and less invasive diagnosis, they have also brought to light the many issues to be tackled before mass-spectra-based proteomic patterns become routine clinical tools. Known issues cover the entire pipeline leading from sample collection through mass spectrometry analytics to biomarker pattern extraction, validation, and interpretation. This study focuses on the data-analytical phase, which takes as input mass spectra of biological specimens and discovers patterns of peak masses and intensities that discriminate between different pathological states. We survey current work and investigate computational issues concerning the different stages of the knowledge discovery process: exploratory analysis, quality control, and diverse transforms of mass spectra, followed by further dimensionality reduction, classification, and model evaluation. We conclude after a brief discussion of the critical biomedical task of [...]

Reference

HILARIO, Mélanie, *et al*. Processing and classification of protein mass spectra. *Mass Spectrometry Reviews*, 2006, vol. 25, no. 3, p. 409-449

DOI : 10.1002/mas.20072

PROCESSING AND CLASSIFICATION OF PROTEIN MASS SPECTRA

Melanie Hilario,[1]* Alexandros Kalousis,[1] Christian Pellegrini,[1] and Markus Müller[2]
[1]*Artificial Intelligence Laboratory, Computer Science Department, University of Geneva, CH-1211 Geneva 4, Switzerland*
[2]*Institute for Molecular Systems Biology, ETH Hönggerberg, CH-8093 Zürich, Switzerland*

Received 14 October 2004; received (revised) 10 May 2005; accepted 17 May 2005

Published online 3 February 2006 in Wiley InterScience (www.interscience.wiley.com) DOI 10.1002/mas.20072

Among the many applications of mass spectrometry, biomarker pattern discovery from protein mass spectra has aroused considerable interest in the past few years. While research efforts have raised hopes of early and less invasive diagnosis, they have also brought to light the many issues to be tackled before mass-spectra-based proteomic patterns become routine clinical tools. Known issues cover the entire pipeline leading from sample collection through mass spectrometry analytics to biomarker pattern extraction, validation, and interpretation. This study focuses on the data-analytical phase, which takes as input mass spectra of biological specimens and discovers patterns of peak masses and intensities that discriminate between different pathological states. We survey current work and investigate computational issues concerning the different stages of the knowledge discovery process: exploratory analysis, quality control, and diverse transforms of mass spectra, followed by further dimensionality reduction, classification, and model evaluation. We conclude after a brief discussion of the critical biomedical task of analyzing discovered discriminatory patterns to identify their component proteins as well as interpret and validate their biological implications. © 2006 Wiley Periodicals, Inc., Mass Spec Rev 25:409–449, 2006

Keywords: *MS preprocessing; classification; biomarker discovery; data mining; proteomics; machine learning; dimensionality reduction*

I. INTRODUCTION

Classification has a long history as a staple statistical technique but has made giant strides with recent advances in machine learning and data mining. Nevertheless, protein mass spectra—like DNA microarray data—raise a number of technical challenges that highlight the limitations of existing classification methods. First, it has been shown that mass spectra mining involves a high risk of finding patterns in noise; thus, more than most other types of data, mass spectra require meticulous and customized quality control, cleaning, and transformation prior to data analysis. Mass spectra preprocessing must take account of multiple factors that govern data production such as sample collection and handling as well as instrumentation. By practitioners' consensus, preprocessing takes around 80% of data mining time; this might well be an underestimation for mass spectra mining where preprocessing involves a complex blend of digital signal processing, data exploration, and data engineering techniques.

Second, a mass spectrum usually contains thousands of different mass/charge (*m*/*z*) ratios on the *x*-axis, each with corresponding signal intensity on the *y*-axis. For data mining purposes, each *m*/*z* ratio is represented as a distinct variable whose value is the intensity; hence each case can be seen geometrically as a single point in a very high-dimensional space. In classification for diagnosis and biomarker discovery, which is the focus of this paper, the problem of high dimensionality is compounded by small sample size: diseased specimens are relatively rare and difficult to collect, especially when invasive procedures are involved. This twofold pathology, called the high-dimensionality-small-sample (HDSS) problem, is the main issue that plagues and propels current research on protein mass spectra classification.

Dimensionality reduction is crucial to biomarker discovery. First, the curse of dimensionality must be coped with if the classification problem is to be solved at all. Whatever the classification goal, the most effective way so far to get around the HDSS problem is by reducing the size of the variable set. More importantly, extracting a handful of variables from an initial set of several thousands is not a simple preprocessing expedient but the very goal of biomarker discovery. The final variables and their interaction in the learned model constitute the proteomic signature, which the biomedical researcher must then identify, validate, and interpret. In short, dimensionality reduction and classification are the co-essential goals of mass spectra mining for biomarker discovery. A corollary requirement is model intelligibility: the selected variables and their respective roles and interactions must not only be accessible in the final classifier, they must be biologically interpretable.

This review describes how the specific characteristics and constraints of mass spectra classification have been handled in biomarker research. The remainder of the paper is structured around the main phases of the generic knowledge discovery process. Section II describes the different ways of preprocessing mass spectra for classification. Although much of the MS classification literature focuses on surface-enhanced laser desorption/ionization time-of-flight (SELDI-TOF) spectra, any

**Correspondence to:* Melanie Hilario, Artificial Intelligence Laboratory, Computer Science Department, University of Geneva, CH-1211 Geneva 4, Switzerland. E-mail: Melanie.Hilario@cui.unige.ch

TABLE 1. Abbreviations Used in This Review

Abbr	Meaning
AUC	Area under the ROC curve
BPH	Benign prostate hyperplasia
CFS	Correlation-based feature selection
ESI	Electrospray ionization
GA	Genetic algorithm
HDSS	High-dimensionality-small-sample
KDE	Kernel density estimation
KNN	K-nearest-neighbor
LC	Liquid Chromatography
LDA	Linear discriminant analysis
MALDI	Matrix-assisted laser desorption/ionization
MS/MS	Tandem mass spectrometry
PC	Prostate cancer
PCA	Principal components analysis
PFF	Peptide fragmentation fingerprinting
PMF	Peptide mass fingerprinting
QDA	Quadratic discriminant analysis
RF	RandomForest
ROC	Receiver operating characteristic
SELDI	Surface-enhanced laser desorption/ionization
SVM	Support vector machine

mass spectrometry (MS) technique can be used for sample classification, and the discussion on MS data preprocessing is, therefore, quite general involving many different types of MS spectra. If methods developed for protein identification are thought to be useful, they were also considered in this review, as well as methods from microarray data analysis. For completeness, Section III gives a brief overview of the major approaches to classification; readers familiar with the basics of building classifiers can skip this section. Sections IV and V discuss the two tasks that form the core of mass-spectra-based diagnosis and biomarker discovery, dimensionality reduction, and classification. Section VI explains how the resulting classifiers are evaluated and selected with a view to optimizing generalization performance and model stability. Section VII briefly presents the postclassification task of interpreting the learned models and patterns to extract biologically meaningful disease markers. Section VIII concludes and previews challenges that lie ahead. Table 1 gives a list of the abbreviations most often used in this review.

II. DATA PREPROCESSING

A. Introduction

Mass spectrometers register when ionized proteins or peptides hit their detector, and this information is then usually compiled into a histogram, which counts the number of detector events within small time bins (for an introduction into different techniques and their application in proteomics see Aebersold & Mann (2003)). Since each time corresponds to a mass over charge ratio (*m*/*z*), the time bins can be converted into *m*/*z* bins. These histogram data are called a "mass spectrum" and form the raw material for all further data processing. Different MS techniques measure mass spectra of different resolution and mass range. The resolution of a mass spectrum is expressed as the full-width-half-maximum (FWHM) ratio, that is, the *m*/*z* value of a signal divided by its width at half of its height ($m/\Delta m$). The resolution can vary from a few 100 for linear TOF spectra to a few 10,000 for delayed extraction/reflectron TOF or Fourier transform mass spectra. The mass range can go from 0 to a few 100,000 Da if entire proteins are measured, or it can be limited to masses smaller than a few 1,000 Da for small peptides or peptides fragments. Mass spectra have several imperfections, which can complicate their interpretation. Despite the large number of different types of mass spectra, there are some common themes a data analyst has to deal with, and some of these are listed below:

- Chemical noise: Matrix-assisted laser desorption/ionization (MALDI; Karas & Hillenkamp, 1988) spectra sometimes contain a high amount of chemical background noise produced by clusters of matrix molecules that are abundant in the sample mixture (Krutchinsky & Chait, 2002). If the protein/peptide mixture is very complex, many weak and overlapping protein/peptide signals will be assigned to the chemical noise, since they are not distinguishable from it. Many mass spectra also contain impurities, that is, molecules that are not proteins or that do not originate from the original biological sample, but from sample preparation or contamination. Examples of such contaminants are polymers, keratin (or other proteins from human skin, hair, or clothing), and trypsin (used to cleave proteins into peptides). Chemical noise is also present in electrospray ionization MS (ESI, Fenn et al., 1989) due to buffers and solvents. If coupled by means of liquid chromatography (LC), chemical noise can be very abundant at the beginning and at the end of the elution process.
- Baseline: In MALDI spectra, chemical noise can be very abundant in the lower mass range causing a strong upward drift in the baseline of the mass spectra, which falls off rapidly with increasing mass. In ESI spectra, chemical noise can form a bump in the baseline in the intermediate mass range.
- Multiple charge states: Peptide ions produced by ESI often carry a different number of elemental charges (charge state) and especially large denaturated protein ions produce a broad distribution of charge states. Since a MS instrument measures the mass over charge ratio, the corresponding protein will be found many times in the spectrum, potentially overlapping with signals of other proteins. Multiple charge states are much less important for MALDI, but can also be seen for large proteins.
- Mass-dependent sensitivity: Most of the currently used ion detectors are based on the electron multiplier technology. The signal produced by these detectors depends on the speed of the ion and not on its kinetic energy (Peng, Cai, & Chang, 2004). Since all ions of the same charge have the same kinetic energy after acceleration, heavier ions are slower and produce a weaker signal (the signal intensity should approximately diminish with the inverse square root of *m*/*z*). Additionally, the resolution of many instruments is also mass dependent.
- Chemical adducts and fragmentation: Large proteins measured by MS are often not pure, but carry chemical adduct ions (e.g., sodium and potassium, solvent, or matrix ions), which stem from the sample preparation. For large

proteins, this can create a distribution of *m*/*z* values, which is broader than the one expected for pure proteins. Especially in MALDI, a protein may also fragment, that is, lose some of its side chains or amino acids, which can also contribute to signal broadening.

- Reproducibility: In MALDI, the signal intensity depends strongly on the laser power, on the amount of sample used, and on the quality of the matrix crystals. Repeated measurements may, therefore, result in largely different absolute intensities. However, if the sample preparation conditions are carefully controlled, good reproducibility can be obtained. Similar facts hold for ESI and other techniques. Since mass spectra measure the outcome of a statistical process they are subject to statistical fluctuations even if experimental conditions are exactly the same. Therefore, peptide signal intensities as well as relative intensities of isotopic clusters can vary significantly between measurements especially if the abundances of the peptides are low.
- Ion suppression effects: The signal intensity of a protein/peptide depends strongly on its chemical composition. Especially if the analyte concentration exceeds a certain threshold, analytes producing intense signals can suppress the signals of other analytes, which are less suitable for ionization. These effects can be seen in MALDI (Kratzer et al., 1998) and ESI (King et al., 2000; Tang, Page, & Smith, 2004) experiments. The signal intensity of certain analytes does not depend linearly on the initial concentration, but is influenced in a complex way by the concentration of other analytes.
- Calibration: As the mass spectrometer measures the times of detector events, these times have to be converted into *m*/*z* values by the application of equations describing the physics of the ion separation process. Some of the parameters entering these processes are only approximately known (e.g., initial velocity and position of the ions) or are neglected in the equations. This can lead to slight shifts in the calculated masses.

The recent discussion on biomarker detection by means of surface enhanced laser desorption/ionization TOF (SELDI-MS) emphasized the relevance of data preprocessing for the classification of mass spectra from healthy and diseased patients (Fung & Enderwick, 2002; Petricoin et al., 2002; Baggerly et al., 2003; Hilario et al., 2003; Wagner et al., 2004). Baggerly, Morris, and Coombes (2004) showed that differences in data preprocessing methods could severely change the outcome of the classification task. Especially baseline correction, mass calibration, intensity normalization, and variable selection methods are crucial and should be carefully evaluated for each application. Data preprocessing is equally important in other proteomic applications. For protein identification by means of peptide mass fingerprinting (PMF) or peptide fragmentation fingerprinting (PFF or MS/MS), the quality of the peak lists will directly influence the quality of the peptide or protein identifications (Blueggel, Chamrad, & Meyer, 2004).

Many proteomic experiments are performed on a large scale, that is, hundreds of mass spectra are acquired from the original sample. This makes it possible to use correlations between these spectra in order to improve data preprocessing. In one application dubbed the "molecular scanner" (Bienvenut et al., 1999; Binz et al., 1999) proteins purified by 2-dimensional gel electrophoresis were digested and mapped onto a membrane, which was scanned on a fine grid by a MALDI-TOF instrument, that is, a PMF was measured at every grid site. The grid spacing was much smaller than the size of detectable spots, and signal intensities were correlated between neighboring grid sites making it possible to smooth out intensity variations and to implement data processing steps that were able to improve the mass calibration, to discard chemical contaminants as well as to detect and separate overlapping protein spots with high sensitivity (Muller et al., 2002a). Information gained from these steps could be incorporated into the protein identification score, which enhanced the specificity of the database search (Muller et al., 2002b).

Data preprocessing can increase specificity and sensitivity of automatic peptide/protein identification for MS/MS data as well. Gentzel et al. (2003) investigated the influence of peak clustering, contaminant exclusion, deisotoping, clustering of similar spectra, and external calibration on protein identification. The first step was necessary since the high resolution of the mass bins of Q-TOF spectra sometimes split peaks apart. Together with the next three steps this led to a reduction in complexity of the mass spectra and made the search more specific.

Besides gel electrophoresis, LC is the most important protein separation technique in proteomics. Its advantage is the speed and flexibility, which makes it possible to serialize different LC methods for multidimensional separation (Washburn, Wolters, & Yates, 2001). It can be used with ESI or MALDI, but LC-ESI MS is most often applied. In this technique, a mass spectrum is acquired for every time step, which is usually smaller than the elution time of a peptide. Since peptide signals have a specific shape in time as well as in *m*/*z* dimension, they can be distinguished from chemical noise (Hastings, Norton, & Roy, 2002). LC-MS data are often used for comparative studies, and data preprocessing and calibration methods were essential in order to obtain good results. Li et al. (2003) analyzed data with isotopically labeled peptides (isotopically coded affinity tags, ICAT), where peptide from two samples are labeled with different mass tags and then mixed for relative quantification. Wang et al. (2003) compared LC-MS data from different samples directly—a procedure, which critically depends on the correct alignment of the datasets.

Preprocessing of mass spectra can roughly be divided into several subtasks (quality assessment, baseline correction, smoothing, noise estimation, peak detection, intensity normalization, and calibration), which are described in the following sections. However, the authors are aware that these tasks are not independent, and several combinations of different solutions of these subtasks may have to be tested in order to find a good-preprocessing method. Some iterative strategies evaluate the results obtained after identification in order to refine data preprocessing. If inconsistencies or missing values appear, data preprocessing is reiterated with different settings until a consistent solution is obtained. Graber et al. (2004) described an example of such a result driven strategy for protein identification and relative quantification.

B. Spectrum Quality Assessment

Quality assessment is the first very important step in data analysis. Detecting spectra with low signal-to-noise ratio helps instrumentalists to adjust their experimental settings, and it allows the data analyst to exclude them from further processing. Data visualization is an easy and rapid way to assess measurement quality, and appropriate visualization techniques can reveal structure in the data, which would be very difficult to detect by purely computational approaches. Heat maps, in which all the spectra are plotted side-by-side and the intensity is represented on a gray scale, are very useful for comparative studies (Baggerly, Morris, & Coombes, 2004). They reveal peaks that can be detected in many spectra as vertical bands. If zoomed into, they show the alignment of the peaks and provide hints whether the masses are well calibrated. In LC-MS experiments the spectra are ordered according to their elution time, which allows a natural representation of the data. These 2-dimensional maps, sometimes called virtual gels in analogy to 2-dimensional gel electrophoresis, can be annotated with data obtained from MS/MS peptide identifications (Li et al., 2004). The maps can be depicted as 2-dimensional grey scale images or as 3-dimensional landscapes (eagle view or surface plots). Similar to microarray or electrophoretic approaches, two LC-MS maps from different samples can be color coded as red and blue images and overlaid. The resulting image shows upregulated or downregulated peptides as red or blue, respectively, and the unchanged peptides as dark magenta (Tammen et al., 2004). To make this approach work, the two LC-MS runs have to be well aligned. One way to check the alignment graphically is to calculate the covariance of all mass spectra in one run with all mass spectra in the other, and to depict the covariance matrix as a contour plot. Off diagonal signals in this matrix indicate alignment errors (Bylund et al., 2002). Multivariate data visualization techniques can be used to explore large numbers of spectra. Principal component analysis (PCA) or discriminate coordinate analysis can project the data onto a 2-dimensional subspace with minimal loss of information, and outliers can be detected visually (Hastie, Tibshirani, & Friedman, 2001; Coombes et al., 2003).

Multivariate methods can control the quality and detect outliers automatically. Coombes et al. (2003) used the Mahalanobis distance in the space of the first six principal components, which accounted for 80%, and a χ^2 test to successfully detect outliers in SELDI chip data. In the same paper, an analysis of variance (ANOVA) of a good-quality replicate dataset obtained on different chips and different days showed that the variance due to chip-to-chip, day-to-day, or spot-to-spot differences was minor compared to the peak-to-peak differences inherent in the measurement, which explained about 90% of the total variance. For replicate microarray data, Model et al. (2002) applied a statistical test based on robust PCA in order to detect failed experiments.

In large scale-protein identification experiments hundreds of MS/MS spectra are measured. Detecting and discarding low-quality spectra from further processing saves computing time and lowers the false-positive rate. On the other hand, detecting high-quality spectra that failed to be identified indicates that peptide identification should be tried with a different method. Sadygov et al. (2002) defined a score based on the number of ion pairs that add up to the parent mass and showed that low-scoring spectra are of lesser quality. A similar score and a set of other scores such as the number of peaks, total peak intensity, the number of ion pairs that have an amino acid mass difference or a neutral loss mass difference were used to classify spectra into good or bad ones (Bern et al., 2004). The performance of this handcrafted classifier was then compared to an off-the-shelf support vector classifier, and it was found that both methods gave similar results being able to correctly classify 90% of the good and about 70% of the bad spectra.

C. Baseline Correction, Smoothing, and Noise Estimation

Roughly, a mass spectrum consists of signals, baseline, and noise (Fig. 1a). The signals are produced by the peptides, proteins, and contaminants present in the sample; the baseline is the slowly varying trend under the spectrum; and the noise consists of chemical background (usually small, except for MS/MS spectra), electronic noise, signal intensity fluctuations, statistical noise, warping of the signal shapes (due to overcharging in ion traps), and statistical noise in the isotopic clusters (see below). Signals, baseline, and noise can never be totally separated. The baseline, for example, can depend on the presence of large and intense signals as well as on abundant low-intensity

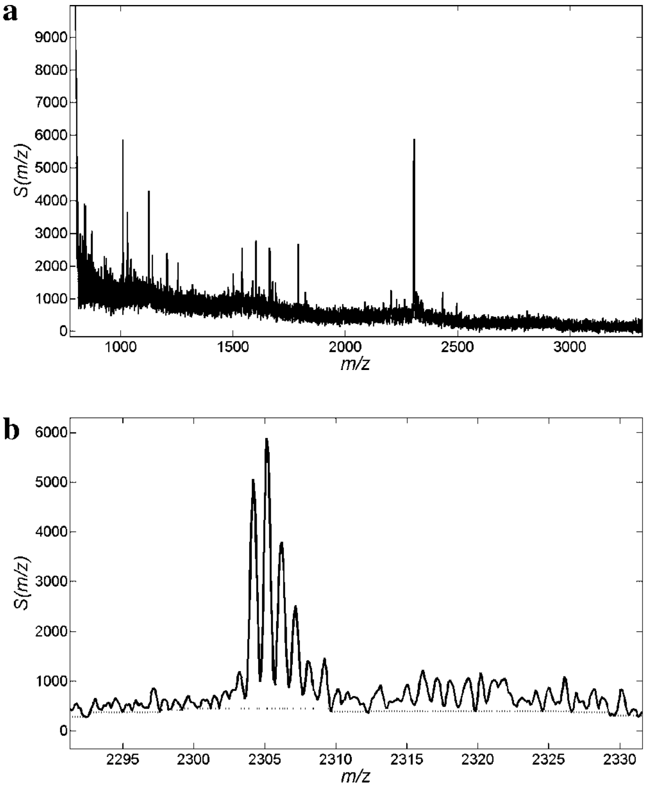

FIGURE 1. Two views of a matrix-assisted laser desorption/ionization time-of-flight (MALDI-TOF) mass spectrum. **a**: The MALDI-TOF mass spectrum clearly shows the baseline drift and the signals sticking out of the noise. **b**: Zoom of the same spectrum, which shows the signal of peptide STQVYGQDVWLPAETLDLIR surrounded by chemical noise. The dotted line indicates the baseline calculated by a Top-Hat filter.

noise. Noise can be quite intense and is sometimes impossible to distinguish from real signals.

In order to provide a model for a mass spectrum $S(t)$ or $S(m/z)$, the contributing terms have to be simplified and are considered as additive and independent. The first term is a sum of independent signals produced by peptides and contaminants. Each signal can be modeled as isotopic clusters (see below) for high-resolution spectra or have a peak-like form for low-resolution spectra and high masses. Each measurement produces spectra of different overall intensity, and the intensity of every single signal is subject to fluctuations as well, which can be modeled by a global random factor δ and an individual random factor γ_i for every signal i, both of which have a mean value of 1. The second term is the baseline b_δ, which depends on the m/z value and δ. The third term ε_δ describes additive noise (mostly chemical noise), which is also dependent on the m/z value and δ.

$$S(x) = \delta\left\{\sum_i \gamma_i \cdot I_i \cdot s_i(x; p_i, z_i, r_i)\right\} + b_\delta(x) + \varepsilon_\delta(x) \qquad (1)$$

where x is either time t or m/z, and s_i is the ideal mean signal of peptide (or contaminant) p_i with charge z_i and mean intensity I_i. The signal also depends on the resolution r_i of the MS instrument, which may be mass dependent (therefore the index i). This dependency can be accurately approximated by convoluting the isotopic cluster with the peak shape of the spectrometer (a Gaussian shape works well for MALDI spectra, FT instruments produce Lorentzian peaks). Since m/z is measured, the m/z value of a signal with charge z is reduced by a factor z, and the signal is compressed by a factor z leading to a spacing between isotopic groups of $1/z$ (Fig. 3). This spectrum model neglects warping and fluctuations of the signal shapes, but for most applications it is general enough. However, its assumptions have to be verified for each type of MS data. A discussion of the various components and how to estimate them is presented in the following sections.

1. Baseline correction

The most important parameter for baseline estimation is the maximal width a signal can have. This width is mass dependent and can depend on the presence of intense or overlapping signals. For TOF data, the signal width increases with the mass, but a logarithmic transformation of the mass values reduces this dependence, which facilitates baseline correction and peak detection (Tibshirani et al., 2004). High pass filters implemented with fast Fourier transform (Press et al., 1995) or filters from mathematical morphology (Breen et al., 2000; Soille, 2003), for example, the Top-Hat filter, can be applied. The latter method is very easy to implement since one only has to calculate the minimum intensity in a sliding window in the first run and the maximum intensity in the second run. Coombes et al. (2003) combined baseline correction and peak detection into a two-step algorithm. First, peaks are detected and subtracted from the spectrum, and the baseline is calculated as a piecewise constant interpolation of local minima. After baseline subtraction, peak detection is run again with newly calculated noise levels.

Another interesting way to design baseline filters is based on wavelet theory (Shao, Leung, & Chau, 2003). The wavelet method transforms the mass spectrum into a hierarchical representation with different scales, and it allows rapidly accessing the data at a certain mass and resolution. Filters that discard the wavelet coefficients above a certain resolution, maybe in a mass-dependent manner, could be used to estimate the baseline. It should also be possible to detect slow oscillations in the baseline, which can occur at higher masses in SELDI spectra.

All of these non-parametric baseline detection algorithms have difficulties to correctly predict the height of the peak in the following situations: if a small peak sits on top of a large and broad peak, or if several larger peaks overlap. In the former case the envelope of the large peak forms the baseline for the smaller one, and in the latter case the baseline stays on the base level even if the width of the total signal is much larger than the expected signal width. A parametric approach that operates with a signal model could alleviate these problems, but it would be difficult to apply if spectra are overcrowded with peaks (Fig. 2).

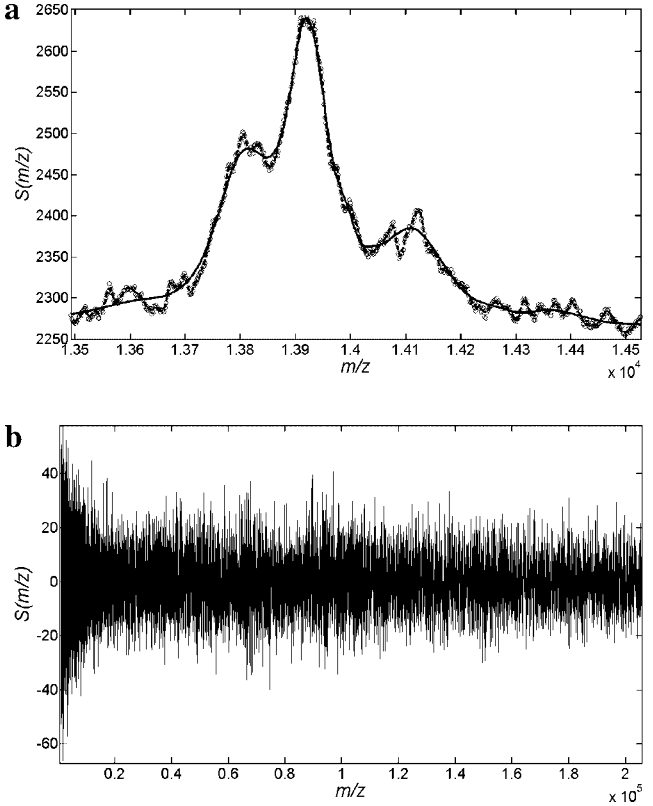

FIGURE 2. Smoothing of (SELDI) spectra: for this type of data, it is difficult to distinguish signal from noise. The strength of the smoothing defines which peaks are discarded and which ones are selected for further processing. **a**: The smoothing was carried out using the wavelet smoother described in Coombes et al. (2004) (free download from http://bioinformatics.mdanderson.org/cromwell.html). The dashed line corresponds to weak (threshold value 6), the solid line to stronger (threshold value 60) smoothing. The number of peaks changes significantly between the two methods. **b**: The estimated noise, which is the original minus the smoothed curve (threshold 60), shows some mass dependence for low masses. This could be a true feature of the noise, but it could also indicate that the smoothing is too strong for low masses. The noise estimated with threshold 6 is much smaller and does not seem to be mass dependent (data not shown).

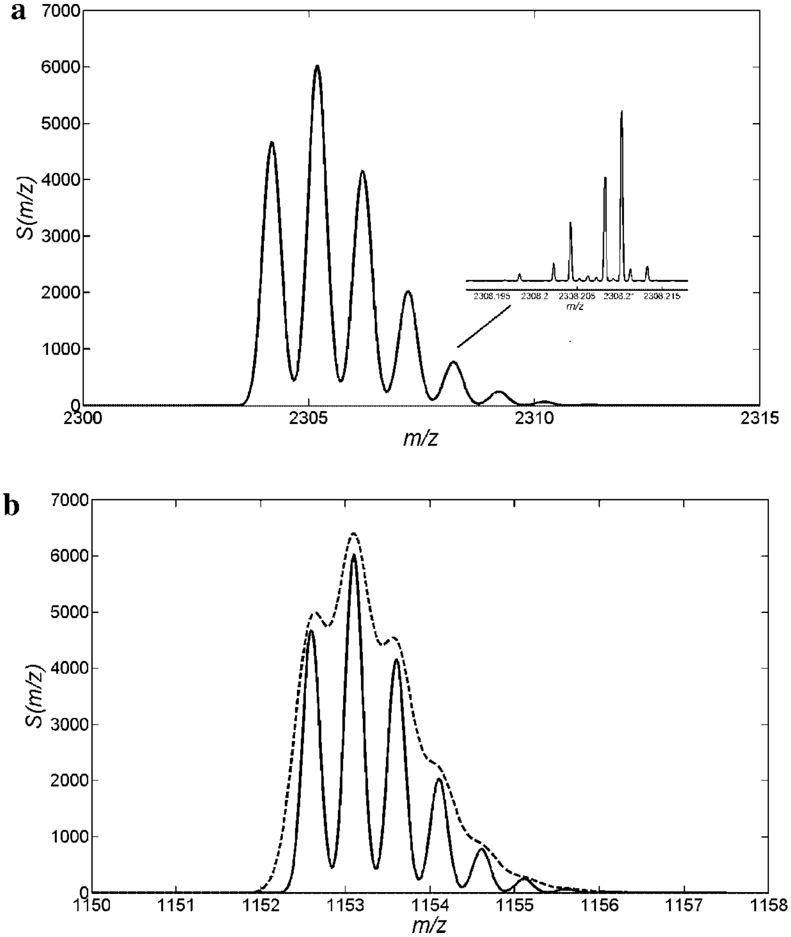

FIGURE 3. Theoretical signal shapes: The mean signals s_i of peptide STQVYGQDVWLPAETLDLIR were calculated by the methods described in Rockwood, Orden, and Smith (1995) and Rockwood and Orden (1996). **a**: The singly charged distribution convoluted with a Gaussian of width 0.2 (full width at half maximum (FWHM) $\approx$ 5,700). The inset shows the fine structure 5th isotopic group at a very high resolution, which is out of reach for standard instruments. **b**: The same peptide, but with charge $z = 2$. The dashed line corresponds to FWHM $\approx$ 3,000 and the solid line to FWHM $\approx$ 5,700.

2. Smoothing and noise estimation

First we focus on additive noise, whereas further methods to estimate multiplicative noise and remove impurities will be discussed further down. After an appropriate smoothing method has been applied, additive noise can be estimated by calculating the expected deviation of the raw data from the smoothed curve in a mass window (Fig. 2b). If the noise is estimated in a region where signals are present, it is preferable to use a robust calculation of the deviation, for example, by using percentiles (Satten et al., 2004). Various methods for smoothing spectra have been developed. It is important that a smoother preserves the peak shape or at least its mean mass and width (first and second moment), especially for high-resolution spectra where the isotopic peaks are visible. Hastie, Tibshirani, and Friedman (2001) discuss smoothing splines, wavelet smoothing, and kernel methods such as locally weighted linear regression. The last method encompasses popular smoothers such as the Gaussian or the Savitzky–Golay smoother (Savitzky & Golay, 1964). In the latter method, a polynomial is locally fitted to the data for each smoothed value, and for polynomials of degree n it can be shown that the first n moments of a peak are preserved. As an extension of the algorithm, the window width and the degree of the polynomial can be defined adaptively for every mass region (Barak, 1995). In a classic paper, Cleaveland (1979) presented a robust version of local weighted regression, where the polynomial is fitted using robust, iterative regression. Coombes et al. (2004) proposed a discrete wavelet approach with hard thresholding, which worked well for low-resolution SELDI spectra (Fig. 2). Filters taken from mathematical morphology are attractive as well, since they are fast and easy to implement with only very few parameters, although their results can be quite jagged (Breen et al., 2000).

Estimation of the multiplicative noise and its dependency on the signal intensity is a different problem. In the best case, a large number of replicate spectra are available and the variation of the signal intensities can be estimated by calculating the variance over the replicates for each signal separately. However, if only a small number of replicates are at hand, the variance estimate is not precise and often too small. If one assumes that the variance is a smooth function of the signal intensity only and does not depend on the peptide, then signals of similar intensity can be pooled, and the variance can be estimated within these pools (Jain et al., 2003). For duplicate data the intensity pairs can be represented as a scatter plot, and noise can be estimated, for example, by the deviation from a smoothed curve. Three similar methods for variance estimation in duplicates, all based on pooled estimates of the variance, are compared for microarray data in Huang and Pan (2002).

In many experiments, spectra are not independent, and the correlation between them can be used for smoothing and to discard chemical noise or contaminants. If a set of spectra is acquired under the same experimental conditions, but with different analytes, the peaks that can be detected in most of the spectra are probably due to sample preparation and not related to the analytes (Chamrad et al., 2003). In the molecular scanner application, the 2-dimensional patterns of the signal intensities revealed whether they could be attributed to chemical noise or impurities (Muller et al., 2002a,b). A similar situation holds for LC-MS experiments, where the subsequently measured spectra are highly correlated and where the elution profiles often show different patterns for chemical noise than for true analyte masses: analytes elute over a short time and show a smooth profile, whereas chemical noise either has a spiky profile or forms a slowly varying background signal. This fact has been used by a series of algorithms designed to purge chemical noise from this type of data. Andreev et al. (2003) smoothed the time domain in LC-MS data using a matched filtering technique, which suppresses the additive noise in the Fourier domain taking into account its frequency characteristics. The component detection algorithm (CODA) by Windig, Phalp, and Payne (1996) calculated statistical descriptors to discard masses showing noisy elution profiles. CODA can effectively clear chemical noise masses present at the beginning and end of a LC run. However, there is a chance that analyte signals are discarded as well if they have the same mass as chemical noise. The 2-dimensional structure of the data matrix in LC-MS experiments facilitates smoothing, since a signal must match in both dimensions. PCA separates the smoothed signals (first PCs) from noise (higher order PCs) if applied to the data matrix (Lee, Headley, & Hardy, 1991). Muddiman et al. (1995) developed the sequential paired covariance (SPC) method based on the correlation between subsequent spectra as a filtering criterion, which allows suppressing noisy spectra. However, since the original signal intensities are replaced by the correlation score, quantitative information is lost in the transformed data. Fleming et al. (1999) reviewed and compared different LC-MS smoothing methods: CODA, PCA, SPC, and their own method based on SPC, which considers only signals that are correlated over time windows of the expected signal width, but not over larger windows.

Data measured in large-scale experiments are often redundant, and several spectra of the same analyte are measured. In LC-MS experiments several spectra of a peptide are obtained during its elution. A peptide can also elute in different forms at different times and be chosen more than once for fragmentation analysis. In order to enhance the signal-to-noise ratio, the redundant information can be compiled. If all spectra contain the same analytes with similar intensities the best way to combine them is to calculate the average spectrum. However, if the spectra have different intensities low-intensity spectra contribute more noise than signal to the sum and should be omitted. Zhang and McElvain (1999) showed for Gaussian profiles and constant noise that only those spectra should be considered, which are more intense than about 40% intensity. If the shape of the elution profile is known, a matched filter can even further enhance the combined signal.

If the identity of the spectra is not known, an unsupervised clustering approach can yield groups of similar spectra, which can be combined for better signal-to-noise ratio. Gentzel et al. (2003) and Beer et al. (2004) described such an approach for LC-MS/MS data (see also Venable et al. (2004)). For 2D gel data, clustering PMF spectra can reveal the similarity of spots without knowing their identity. Additionally, it allows determining which masses stem from a spot itself and which come from overlapping spots (Schmidt et al., 2003). In the framework of the molecular scanner technique masses (and not spectra) are grouped if they have similar 2-dimensional profiles. This proved to be very useful for sensitive protein spot detection and separation of eventually overlapping spots. It could also largely improve a PMF scoring system, since peptide masses from a protein should have similar 2-dimensional profiles, and random matches to masses from chemical noise or overlapping protein spots could be discarded due to their different profiles (Muller et al., 2002a,b).

D. Peak Detection and Charge State Estimation

Mass spectra usually contain several 10,000 up to 1,000,000 sampling values. However, intensity values are correlated since mass signals are usually much broader than the sampling width. Also, mass spectra can contain large regions that do not contain useful information. Extracting the relevant signals from a mass spectrum is, therefore, a means to reduce its very large dimension to a more manageable size of several hundred features.

Attempts to classify SELDI spectra using raw data directly (Petricoin et al., 2002) have to consider the "curse of dimensionality" (Somorjai, Dolenko, & Baumgartner, 2003), and the results have to be analyzed critically (Baggerly et al., 2003; Sorace & Zhan, 2003). A difficulty with the whole spectrum approach in biomarker discovery studies is its lack of interpretability. If one detects a significant difference between two groups of spectra that lies in a noisy region, how could this be explained, and how could a potential biomarker molecule be extracted from this knowledge? Or if the differences are found only in the flanks of a peak: is this due to overlapping peaks, due to a change in the chemical composition of the ion reflected in a different peak shape or just due to different mass calibration? Therefore, many authors use peak detection before further analysis is carried out (Fung & Enderwick, 2002; Hilario et al., 2003; Wang et al., 2003; Yasui et al., 2003; Coombes et al., 2004; Tibshirani et al., 2004), although other strategies such as simple *m*/*z* binning were also applied (Purohit & Rocke, 2003). Prados et al. (2004) investigated the influence of peak detection thresholds in SELDI spectra on the classification performance.

For protein identification by means of PMF or PFF, the quality of the peak lists defines the error rate of protein identifications (Gras et al., 1999; Gentzel et al., 2003), since peak lists containing too many insignificant entries will have a low specificity in the identification process, whereas missing peaks will impair the sensitivity. The precision of the measured mass value is crucial for a good-specificity of protein identification by MS (Clauser, Baker, & Burlingame, 1999).

1. *Isotopic distribution*

The ideal signals produced by peptides or proteins are isotopic clusters, that is, the ensemble of all isotopic masses of a peptide, weighted with their frequency of occurrence in the biosphere. Isotopic masses group around m/z values of $(m_0 + i)/z$ Da, where m_0 is the so-called monoisotopic mass and $i = 1, 2, \ldots, n$. Within each group neighboring isotopic masses differ by less than $0.01/z$. The measured signal consists of the isotopic masses convoluted with the peak shape of the instrument. The width of the entire isotopic cluster goes from 1 Da for small masses (~100 Da) up to more than 50 Da for 100,000 Da (it raises approximately with the square root of the mass). Depending on the resolution of the MS instrument single isotopic peaks are either distinguishable or they melt into broader peaks containing several isotopes (Fig. 3). For singly charged peptides, modern MS instruments can clearly resolve the isotopic groups at $m_0 + i$ Da for m_0 up to several thousand daltons, and Fourier transform ion cyclotron resonance (FTICR) spectrometers are even able to resolve isotopic fine-structure within these groups (Shi, Hendrickson, & Marshall, 1998).

The isotopic distribution of peptides of known sequence can be calculated using tables containing the masses and abundances of isotopes of each element. Rockwood, Orden, and Smith (1995) and Rockwood and Orden (1996) presented an elegant and very fast solution to the problem, which represented the isotopic distribution of a peptide as a convolution product of elementary distributions and used the fast Fourier transform method for its calculation. However, in most applications the sequences of the peptides are not known before peak detection, and an isotopic distribution typical for the investigated mass range has to be used (Berndt, Hobohm, & Langen, 1999; Gras et al., 1999; Breen et al., 2000). Fortunately, peptides of similar mass but different composition usually have similar isotopic distributions (especially in the fist two isotopic groups), and a distribution averaged over peptides within a mass range (say 100 Da) provides a good approximation.

2. *Peak detection*

In the parametric approach to peptide signal detection, a model of a peptide signal is matched against the raw data, and where the match exceeds a certain threshold a signal is assumed to be present. The model may contain various parameters, such as offset from baseline (to correct errors in baseline subtraction), signal height, width of the isotopic peaks (Berndt, Hobohm, & Langen, 1999; Gras et al., 1999), and charge state for ESI spectra (see next section). Gras et al. used a matched filter approach to locate potential peaks and then performed a non-linear regression to adjust the peak width and height. The fitted models were then subtracted from the raw data, and the algorithm was run again in order to find overlapping peaks.

An isotopic distribution of finite peak width can be considered as a linear transformation of a signal consisting only of a sharp peak at the monoisotopic mass (mathematically it is the product of two convolutions: the first produces the isotopic distribution and the second blurs the sharp peaks). This transformation can be inversed by mathematical techniques, which are either based on Fourier transform methods such as matched filtering (Palmblad, Buijs, & Hakansson, 2001; Andreev et al., 2003) or on generalized inversion and regularization theory (Mohammad-Djafari et al., 2002). The latter method was applied by Zhang, Guan, and Marshall (1997), who used maximum entropy regularization, and by Samuelsson et al. (2004), who used constrained quadratic programming and a regularization term that penalizes too many overlapping signals. Linear inversion methods have the advantage that they can decompose overlapping signals directly as long as these have the same charge state, width and offset.

For high masses or low-resolution mass spectra the isotopic peaks may not be visible and collapse into a single broad peak, the shape of which may be distorted by fragmentation and chemical adducts. In order to describe such a broad peak the isotopic model may not be accurate, and more flexible approaches such as the "exponentially modified Gaussian" (Malmquist, 1994) or the very flexible "empirically transformed Gaussian" (Li, 1997) could be used. Shackmana, Watson, and Kennedy (2004) took the latter model to deconvolve overlapping peaks by means of non-linear regression.

For low-resolution peaks, one could also refrain from using parametric models. The easiest way to find broad low-resolution peaks is to smooth the raw spectrum and then take those local maxima which exceed a threshold value (Yasui et al., 2003; Coombes et al., 2004). The first derivative indicates peak flanks if it exceeds a certain threshold (Coombes et al., 2003; Shackmana, Watson, & Kennedy, 2004). Wallace, Kearsley, and Guttman (2004) presented a different technique to find summits and valleys: starting with a straight line that connects the first and last point in the spectrum, the algorithm finds the point in the raw spectrum that is farthest from this line. It adds this point as a new node in the piece-wise linear interpolation of the raw data and repeats these steps until no significant peaks are left. Jarman et al. (2003) used a statistical test to check whether the histogram within a sliding window (ion counts *vs.* time or m/z bins) resembles a uniform distribution or has a peaked shape. The test considers baseline and noise in the raw data, and it is performed for varying window width in order to cope with different peak widths. For non-parametric peak detection there are two options to quantify peaks: peak height or area above the baseline. Peak height is less sensitive to disturbance by other overlapping signals, but it neglects the width of the signal. Peak area considers the full signal and averages out random noise, but beginning and end of a peak have to be well defined.

One disadvantage of non-parametric methods is that they cannot detect strongly overlapping peaks. Various filters used in image processing, such as the "unsharp masking" and high pass filters (Carroll & Beavis, 1996) or second-derivative filters (Grushka & Israeli, 1990), allow enhancing the resolution of a mass spectrum as well as removing the background. Fast Fourier transform is a powerful tool to implement these filters and to combine them with prior smoothing of the raw data. More recently, wavelet transforms have been applied to separate overlapping signals (Shao et al. (1997); Shao, Leung, & Chau, 2003). Mohammad-Djafari et al. (2002) reviewed another way to design these filters by means of inversion theory, where suitable regularization techniques help limiting the high variance in the filtered data.

Most of these algorithms use thresholds for signal to noise ratios (or other scores) to exclude random peaks. The threshold for the signal-to-noise ratio can be obtained from statistical analysis of the noise. The distribution of the noisy peak intensities can be estimated for a certain mass window, and all intensities that have a low *P*-value with respect to this distribution can be considered as real peaks. Another option is to link the peak detection threshold directly to the identification or classification process. Gras et al. (1999) took a supervised learning procedure to obtain optimal peak detection thresholds. The MS/MS identification software Mascot (Perkins et al., 1999) evaluates several peak detection thresholds and the one with the best identification *P*-value is taken. Prados et al. (2004) investigated the influence of the peak detection threshold on mass spectra classification.

3. Charge detection and charge deconvolution

MALDI spectra have the advantage that the charge state of peptides is almost always $z = 1$. ESI, on the other hand, produces multiply charged ions, and each peptide usually appears in several charge states corresponding to different peaks in the spectrum. For native globular proteins of known structure, the charge state can be readily predicted since it correlates well with the diameter of the protein (Felitsyn, Peschke, & Kebarle, 2002), but denatured proteins or peptides can produce a broad distribution of charge states depending on their chemical composition (mainly the number of basic amino acids for the positive ion mode).

For high-resolution spectra, the charge state can be directly read from the spacing between isotopic peaks. Senko, Beu, and McLafferty (1995) investigated two commonly used techniques: Fourier transform frequency analysis and the Patterson transform, which calculates the autocorrelation in the neighborhood of a peak. The authors found that the combination of the two charge state estimators provided better results over a wide range of conditions. Zhang and Marshall (1998) replaced the autocorrelation by a more robust score in order to determine the charge state of isotopic clusters. However, especially for low-resolution spectra or noisy spectra and overlapping peaks, the frequency estimation can perform poorly, and it is better to find the charge state whose isotopic pattern fits best to the data (Gentzel et al., 2003; Li et al., 2003; Wang et al., 2003).

For low-resolution spectra, where the isotopic peaks are not distinguishable, multiple charge states can be an advantage since the uncharged parent mass can be calculated more precisely as a weighted sum of the measured *m*/*z* values of the different charge states. If the peptide mass is not known, Mann, Meng, and Fenn (1989) presented a simple charge deconvolution algorithm, that yields the peptide mass *m* from a sequence of multiply charged experimental masses. For all *m*, this algorithm simply calculates all possible *m*/*z* values within the mass range and sums up the intensities in the intensities at these values. It does this for all masses, and the mass that yields the highest intensity sum is believed to be the singly charged peptide mass. However, the algorithm is sensitive to calibration errors, baseline, and noise. It is also sensitive to outliers, and more robust measures have been introduced (Reinhold & Reinhold, 1992). If many proteins are present in the sample, it might be useful to transform multiply charged spectra into singly charged ones. Zhang and Marshall (1998) start with the most intense signal, determine its charge state, transform the signal into a charge 1 signal in an artificial spectrum, discard the processed signals in the original spectrum, and go on to the next most intense peak, until all signals above a certain intensity/noise threshold are processed.

For MS/MS experiments, the charge of the parent ion is an important parameter for peptide identification, and search time and efficiency can be improved if the charge state is known. For high-resolution spectra, the charge can be determined by the spacing between isotopic peaks, but this is often not possible for low-resolution ion trap spectra. However, ions of different charge states fragment differently, and one can try to determine the charge of the parent ion from its fragmentation spectrum. Sadygov et al. (2002) proposed a score, which counts for each parent charge the fragment ion pairs adding up to the parent mass. A different approach uses the fact that fragment masses are always smaller than the parent ion mass. However, the *m*/*z* value of fragment ions can be larger than the *m*/*z* value of the parent ion (at most by a factor of *z*), and the distribution of fragment *m*/*z* values with respect to the parent *m*/*z* value was found to be indicative for the parent charge state (Colinge et al., 2003).

4. LC-MS peak detection

LC-MS experiments record a mass spectrum for each step in the elution time from the LC column, which produces data that has a special 2-dimensional structure (Fig. 4). An ideal LC-MS peptide signal can be represented as a bilinear form: $S(t,m/z) = S_1(m/z)S_2(t)$ where S_1 is the mass signal (e.g., isotopic peak cluster) and S_2 is the elution profile (it is assumed that a peptide elutes over several time steps). A real LC-MS signal consists of a sum of isolated signals plus noise and baseline where the noise has a different elution profile than peptides, which makes it possible to distinguish it from signals. Although the elution profile of a peptide is less well defined and less reproducible than its *m*/*z* signal, a Gaussian shape is usually a rather good approximation, but more flexible refinements were proposed (Malmquist, 1994; Li, 1997). Peak detection in the time domain is basically the same as peak detection in low-resolution mass spectra, and it can be done in a non-parametric or parametric way. In order to find signals in 2-dimensional LC-MS runs, Hastings, Norton, and Roy (2002) presented a simple method dubbed "vectorized peak detection," which first smoothes the SICs by a median filter and then considers only those points in (t, m/z)-space that have a local maximum in t and match a peptide signal in *m*/*z*-direction. Andreev et al. (2003) used matched filtration in the time domain and scored each peak by a multiplication of its time and mass domain scores.

The special bilinear structure of LC-MS data can either be used to directly find the number of signals present, or for smoothing or comparison with other experiments. In the absence of noise the number of linearly independent components or the

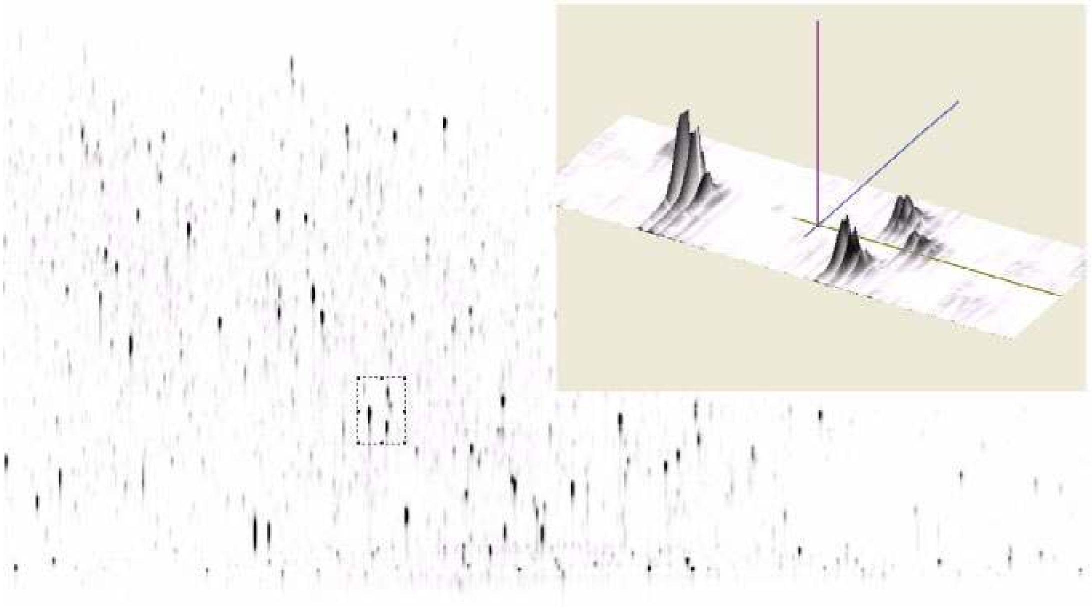

FIGURE 4. LC-MS data: this figure shows good-quality data, which contains very little noise. The elution time is along the vertical and *m/z* along the horizontal axis. The inset shows a 3-dimensional surface plot of the marked rectangle and reveals how four isotopic distributions smoothly elute over time. The images were produced by the MSight LC-MS viewer, freely available at http://www.expasy.org/MSight.

rank of the data matrix $\mathbf{S}(t,m/z)$ equals the number of signals (Fraga, Bruckner, & Synovec, 2001). Generalized rank estimation methods, which provide a robust estimate of the number of independent components, therefore provide an estimate of the number of signals. For small windows of LC-MS data that contain a mixture of bilinear signals, PCA can yield pure 1-dimensional signals (elution profiles and isotopic distribution). However, the solutions are not well defined, and additional constraints have to be defined (Kiers, tenBerge, & Bro, 1999). This problem can be avoided if multi-way data are available, that is, if the same measurement is repeated with different concentration of peptides. Then multi-way decomposition methods such as PRAFAC or Tucker3 provide the pure signals of the analytes without ambiguity under mild regularity conditions (Bro, 1997; Kiers, tenBerge, & Bro, 1999).

Certain analytes such as polymers or glycosylated peptides can produce extended 2-dimensional patterns of peaks in a LC-MS run. Polymer chains, for example, often have different lengths, and these chains differ in a number of polymer units. Since the mass is directly proportional to the number of units in the chain, and the elution time is often a nearly linear function of the chain length, these polymers form a nearly linear pattern in the $(t,m/z)$-space. Marchetti et al. (2004) used the two-dimensional autocorrelation function to detect such linear patterns.

E. Intensity Calibration and Variance Stabilization

Even after baseline correction and smoothing, it is possible that large experimental variations remain in the data, since the signal intensities can change between experiments due to different total analyte concentration or ionization efficiency, for example. In order to even out these experimental variations signal intensities are usually normalized, that is, the intensity values are transformed to new values, which are less dependent on experimental conditions. For MALDI/SELDI data the peak intensities are often divided by the sum of all intensities (total ion count or TIC) of the spectra in order to be less dependent on variations in laser intensity or matrix crystal formation. Sometimes a single very abundant protein (e.g., albumin in human plasma samples) can dominate the TIC, which should be excluded unless its concentration is known to be constant, or a more robust approach such as normalization by the median intensity of the peaks has be considered. Another normalization strategy is to replace intensities by their signal-to-noise ratios, where the noise is estimated in a window around a signal (Satten et al., 2004).

The standard deviation of peak intensities depends on the intensity itself, which makes the application of statistical tests more cumbersome. Coombes et al. (2004) and Wang et al. (2003) found a linear dependence for SELDI and LC-MS data, respectively. If the dependence is strictly linear, a logarithmic transformation of the intensities will produce constant standard deviation (it turns the multiplicative noise into constant additive noise)-a property, which is called variance stabilization. Detailed studies on the intensity dependence of the standard deviation have been performed for microarray data. Durbin et al. (2002) proposed a statistical model of fluorescence intensities, which consists of background as well as a multiplicative and an additive noise term. In this case the standard deviation σ depends on the mean intensity μ quadradically ($\sigma^2 = \mu^2\sigma_1{}^2 + \sigma_2{}^2$), and a logarithmic transformation cannot stabilize the variance anymore for small intensities. However, the authors could show that the arcsinh transformation, which approaches the logarithm for large arguments, yields constant variance. The same findings were obtained for a more general quadratic relation between variance and mean intensity (Huber et al., 2002). The authors also compare the arcsinh normalization with other approaches and show its good performance for real microarray data. In a recent paper, Anderle et al. (2004) studied noise models for LC-MS experiments, and found that a quadratic variance/intensity relation fits the data very well. On the other hand, Coombes et al. (2003) found for a SELDI dataset that a cube root transformation was most successful (among those transformations examined) at stabilizing the variance.

In many applications such as ICAT quantification experiments the signal intensities of a peptide under different biological conditions are compared. Statistical tests have to be developed in order to find out whether peptides are significantly upregulated or downregulated compared to random variations in the intensities. This situation is very similar to cDNA microarrays experiments (Yang et al., 2002) or to difference gel electrophoresis (DIGE) applications (Kreil, Karp, & Lilley, 2004). For ICAT quantification in large-scale LC-MS experiments, Li et al. (2003) assumed that the majority of peptides do not change their intensity except for a scaling factor common to all equally labeled peptides. Further they assumed that the logarithm of the intensity ratios has a Gaussian distribution for the unchanged peptides, and an unsupervised fitting procedure yields a normalized Zscore and *P*-value for each intensity ratio. In another LC-MS experiment that was performed without labeled peptides, signal intensities of peptides from different runs are compared directly (Wang et al., 2003). The intensities of each run were normalized by a constant factor in order to set the median intensity ratio equal to 1. Intensity ratios have been studied extensively in the context of microarray data. Chen, Dougherty, and Bittner (1997) deduced the intensity ratio probability distribution under assumptions that the intensities of each gene are normally distributed and the standard deviation of the intensities are proportional to their mean values, where the proportionality factor c is the same for each gene and fluorescent. Under these assumptions, which can to a certain extent be justified biologically, the intensity ratio distribution does not depend on the mean intensity of a gene making it possible to apply the same test to all genes. The authors also proposed an iterative algorithm that corrects a constant scaling factor for red or green intensities. Powell et al. (2002) used Monte Carlo simulations to demonstrate the robustness of this method with respects to violations of the tests main assumptions (normality and constant c).

These applications assume that the differences of signal intensities I of a peptide i between the two groups are mainly due to a constant scaling factor k:$I_{1,i} = kI_{2,i}$ for all i. However, small intensities can be dominated by background term b_i, and a better relation would be $I_{1,i} = kI_{2,i} + b_i$. Chen et al. (2002) provided an extension of their test including background correction terms, which have an influence on genes with a low signal-to-noise ratio. In order to calibrate the intensities, the values of k and the mean background b can be determined by a robust linear fit or a more general relation $I_{1,i} = f(I_{2,i})$ can be obtained by applying a non-linear scatterplot smoother to the intensity data (Yang et al., 2002). Zien et al. (2001) presented a maximum likelihood calibration algorithm to calculate scaling factors, which is based on a normal distribution of intensity ratios and which works as well for more than two groups.

F. Mass Calibration and Time Alignment

Calibration of MS data is a crucial step in data preprocessing. The precision of the *m*/*z* values determines the error rate of protein identifications (Clauser, Baker, & Burlingame, 1999; Chamrad et al., 2003). In comparative studies, small shifts in the *m*/*z* values can blur the distinction between groups of samples. For example, Baggerly, Morris, and Coombes (2004) showed the importance of calibration issues for SELDI-TOF classification of ovarian cancer samples. For LC-MS experiments, the relative variations in elution time are usually much higher than those in *m*/*z*. In order to compare different LC-MS runs the elution times have to be aligned (Wang et al., 2003). Since mass and time calibrations can be performed independently and since they have to deal with quite different problems, they will be discussed in separate sections.

1. Mass calibration

As already mentioned in the introduction to this chapter, the times of detector events have to be converted into *m*/*z* values. The conversion formulas contain various experimental parameters, some of which cannot be known exactly or are subject to variations leading to errors in the *m*/*z* values. Vestal and Juhash (1998) give a detailed discussion of these formulas for various TOF configurations and calculate their dependency on initial ion velocity and position, for example, two parameters, which cannot be determined with certainty. If the flight times, the conversion formulas as well as the flight times of some reference ions of known mass are available, unknown parameters in the conversion equations can be defined by a fitting procedure (Christian, Arnold, & Reilly, 2000). However, the exact conversion formulas or reference times are not always available to the data analyst, and other calibration methods have to be applied. Usually *m*/*z* errors are quite small (less than 0.2% for linear TOFs down to less than 0.001% for Fourier transform instruments), and they can be corrected approximately by a simple affine transformation (i.e., $x' = ax + b$) of the *m*/*z* values if two or more reference *m*/*z* values are provided (Egelhofer et al., 2000; Gentzel et al., 2003). Higher order correction polynomials can substantially reduce the error compared to an affine correction if many evenly distributed reference masses are present (Gobom et al., 2002). However, if only a few masses (less than 5) are present, first-order correction is the safer method. For singly charged ions, peptide masses (up to 4,000 Da) can be adjusted even in the absence of reference masses since peptide masses are not continuously distributed, but are concentrated in narrow intervals separated by 1.00045 Da (Gay et al., 1999). A mass correction can then be applied in order to find as many masses as possible within these intervals (Gras et al., 1999; Wool & Smilansky, 2002; Muller, 2003), and masses outside the intervals can be discarded as outliers (Schmidt et al., 2003). In the case of protein identification, PMF or PFF masses can be adjusted in order to give the best match with theoretical masses obtained for each candidate protein or peptide sequence, respectively. Gras et al. (1999) and Egelhofer et al. (2002) used robust linear regression to align theoretical and experimental masses for each candidate protein and to discard outlier masses. This allowed working with a much lower mass tolerance, and the specificity of the search could be greatly improved.

In many applications multiple mass spectra are measured from the same sample and data processing is facilitated if they all are aligned. Normally peaks are simply clustered together if their mass difference is less than a certain value. Fung and Enderwick (2002); Yasui et al. (2003), and Prados et al. (2004) discuss their clustering strategies in some detail. Often not all spectra are equally similar to each other, and it might be better to first align the more similar spectra, for example, by first constructing a similarity tree and then aligning the spectra following the tree structure. In the molecular scanner approach, the 2-dimensional topology of the MS data could be used. Since calibration errors were quite smoothly distributed as a function of the position on the membrane, mass deviations between neighboring spectra were small and these spectra could be aligned more easily with respect to each other. An iterative algorithm aligned each spectrum with respect to its four neighbors until the mass deviations were all evened out. Some spectra, which provided many reliable reference masses, were used as anchor points in order to force the algorithms to converge to the right values (Muller et al., 2002a).

2. *Time alignment in LC*

Shifts in LC retention time are caused by different injection timing (constant shift), slow and fast temperature fluctuations, and flow rate changes. They are more irregular than mass calibration errors, and a low-order polynomial may only be sufficient to correct the trend in the errors but not the intermittent fluctuations. An alignment strategy consists of a mapping $t' = f(t)$, which is able to correct these irregular deviations under the condition that it neither reverses time order (monotonous function) nor introduces sharp changes. Dynamic time warping (DTW) is frequently used in signal processing tasks in order to align warped signals (Aach & Church, 2001). It is based on a dynamic programming algorithm, which finds a globally optimal solution maximizing the similarity between two signals, but which can be quite time- and memory-consuming for large signal vectors. Several authors used DTW to align LC data: Wang and Isenhour (1987) used an integer valued warping function to minimize the Euclidian distance between two signal vectors under monotonicity constrains. Nielsen, Carstensen, and Smedsgaard (1998) developed the correlation optimized warping (COW) algorithm, where they divided the signal into intervals. These intervals were shifted, stretched, or compressed without violation of monotonicity and continuity constrains, and the authors used DTW to find the piecewise linear warping function, which provides the best correlation between the two signals. In order to overcome the computational burden of DTW Forshed, Schuppe-Koistinen, and Jacobsson (2003) determined the end positions of the intervals with a genetic algorithm (GA), which finds a reasonably good solution within a short time even for large chromatograms. Eilers (2004) used polynomial time warping and developed an iterative algorithm in order to find the polynomial coefficients that minimize the Euclidian distance between two chromatograms.

The methods discussed so far compare raw chromatograms (eventually baseline corrected and smoothed) without peak detection. If the complexity of the spectra is not too high and there is a clear correspondence between peaks in the two chromatograms, then the time shifts can be directly measured, and a time alignment can be obtained by linear interpolation of these shifts (Johnson et al., 2003). Malmquist (1994) proposed a similar method where the chromatograms are first aligned using the highest peaks, and then smaller peaks with a good correlation between the two chromatograms are taken into account to refine the calibration.

For LC-MS experiments peaks in the LC chromatogram can be identified by their corresponding mass spectra, which should provide a clearer distinction. Wang et al. (2003) used information from both time and *m/z* dimensions in order to align elution times by means of a DTW algorithm. Bylund et al. (2002) presented a modified version of the COW algorithm adapted for LC-MS data, where they used the covariance of mass spectra as the similarity score used in the alignment.

III. A BIRD'S EYE VIEW OF CLASSIFICATION METHODS

Biomarker discovery is aimed at finding a set of discriminatory proteins to diagnose different states with respect to a given disease. Such a diagnostic model can be built from mass spectra of biological samples (e.g., serum), which have been labeled by biomedical specialists, that is, assigned to one of several predefined classes or disease states, for example, in the simplest case, diseased (positive) *versus* control (negative). After the preprocessing operations described in "Data Preprocessing" section, a collection of mass spectra is represented by an $n \times p$ matrix $\mathbf{X}$. Each of the n rows is a spectrum of p selected peaks, and each cell M_{ij} contains the normalized intensity of the jth peak of spectrum i. Associated with matrix $\mathbf{X}$ is a vector $\boldsymbol{Y}$ of n class labels, which can be viewed as the $(p+1)$th column of $\mathbf{X}$; label Y_i $X_{i(p+1)}$ is the class or disease state of spectrum X_i. The model induced (or learned) from this labeled dataset will serve to diagnose new cases (spectra), that is, assign them to one of the prespecified classes. In data mining or machine learning terms, diagnosis and biomarker discovery can be cast as a classification task. The generic model of classification assumes:

- a generator of random vectors $\boldsymbol{x}$, which are drawn according to an unknown but fixed probability distribution, P($\mathbf{X}$).
- a supervisor which assigns output values, class labels, $\boldsymbol{y}$, to the $\boldsymbol{x}$ random vectors, according to an unknown but fixed conditional probability distribution $P(Y|\mathbf{X})$.

The pairs $(\boldsymbol{x},\boldsymbol{y})$, drawn from the probability distribution $P(\mathbf{X},Y) = P(Y|\mathbf{X})P(\mathbf{X})$, constitute the learning space. The task of the learner is to build a classifier, in other words, an approximation of $\boldsymbol{Y}$ as a function of $\mathbf{X}$ and a set of model parameters θ in a space of hypotheses (Fig. 5).

Generative approaches model the class-conditional densities $p(x|\boldsymbol{y}_i)$ and the priors $p(\boldsymbol{y}_i)$, and then use Bayes' theorem to estimate posterior class probabilities $p(y_i|\boldsymbol{x}) = p(x|\boldsymbol{y})p(\boldsymbol{y}_i)/p(\boldsymbol{x})$ where $p(\boldsymbol{x})$ serves to normalize the result to the [0,1] interval. Discriminative approaches make no attempt to model the underlying joint data distribution but model posterior class probabilities directly; that is, they assume some functional form for $p(Y|\mathbf{X})$ and estimate its parameters directly from the training data.

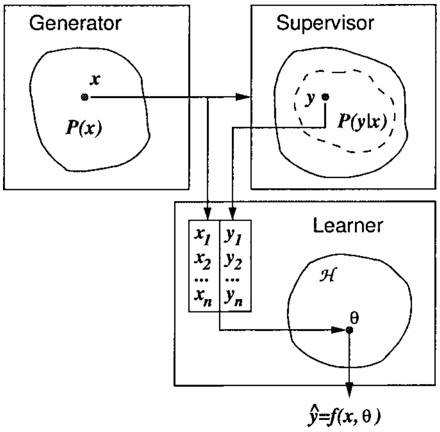

FIGURE 5. Classification as a supervised learning task. The training data are assumed to be drawn from an unknown probability distribution $P(\mathbf{X})$ and the class labels from $P(Y|\mathbf{X})$. A learner builds a function that estimates the joint distribution $P(\mathbf{X},Y)$ to predict a class $\hat{y}$ for a new instance x.

The generative/discriminative distinction applies to base level learning approaches or models. Very often, ensemble approaches improve classification performance by building committees of such base models and combining their responses in some way to output a common, more informed decision.

A. Generative Approaches

Generative models are so-called because they express a hypothesis about how the data were generated. Naïve Bayes is a simple classifier that assigns a case x to the most probable class given x. The method uses Bayes' theorem to compute the posterior probability of each candidate class y_i. Naïve Bayes owes its name to the simplifying hypothesis that all variables are mutually independent. Thus the class-conditional density of the data is computed as the simple product of the individual class-conditional densities of the variables. In short, learning a classifier reduces to estimating class priors and class-conditional densities; classifying a new case consists in using these estimations to compute the posterior of each class and selecting the class with the highest posterior probability.

Other density estimation methods come in different flavors based on initial assumptions about these densities. Parametric approaches assume the data to have been generated according to a given probability distribution specified by a set of parameters. For instance, linear (LDA) and quadratic discriminant analysis (QDA) assume that the class densities are Gaussian. The distinction between the two arises from a second assumption regarding class covariance. If the classes have a common covariance matrix, the class boundaries (or decision boundaries) become linear in x; the discriminant function for a given class k is defined as

$$\delta_k^{\mathrm{LDA}}(x) = x^{\mathrm{T}} \sum^{-1} \mu_k - \frac{1}{2}\mu_k^{\mathrm{T}} \sum^{-1} \mu_k + \log \pi_k$$

Otherwise the discriminant remains quadratic in x:

$$\delta_k^{\mathrm{QDA}}(x) = -\frac{1}{2}\log\left|\sum_k\right|\frac{1}{2} - (x-\mu_k)^{\mathrm{T}} \sum_k^{-1} (x-\mu_k) + \log \pi_k$$

In both cases the Gaussian parameters are estimated from the training data, and x is assigned to $\mathrm{argmax}_k \delta_k(x)$ (see Hastie, Tibshirani, & Friedman (2001) for further details).

For less well-behaved data, non-parametric approaches make no prior assumptions and estimate densities in a purely data-driven manner. Kernel density estimation and K-nearest-neighbor classifiers are non-parametric approaches that estimate densities in the local vicinity of a new case. In kernel density estimation (KDE, also known as Parzen windows), a kernel function (e.g., a Gaussian) is centered on each training case; the width of the kernel, a user-specified parameter, determines the region of influence of each case. To classify a new example, the class-conditional density at its precise location in instance space is estimated as the sum of all other individual densities whose region of influence encompasses the new location. In K-nearest neighbors (KNN), the size of the local vicinity is determined by the user-defined parameter K, the number of neighbors to be considered. No internal model is built; learning is simply storing the training cases. To classify a new case, its KNN (i.e., the K cases most similar to it in terms of predictive variable values) are identified using a similarity metric such as Euclidean distance. The new case is assigned to the most frequent class among these K neighbors. While no probability densities are explicitly computed as in kernel density estimation, KNN classification can be viewed as delimiting a sphere-like region centered on the query case and estimating the posterior probability of the class within that region (Duda, Hart, & Stork, 2000).

B. Discriminative Approaches

Discriminative approaches build a direct mapping from inputs to class labels or model posterior class probabilities without modeling the underlying joint probability density. Logistic regression models class posteriors using a function that is linear in x:

$$P(Y = y_i|\mathbf{X} = x) = \frac{1}{1 + \mathrm{e}^{-(\alpha+\beta^{\mathrm{T}}x)}}$$

In the binary case, the logit transform of the above model yields the linear discriminant function:

$$\delta(x) = \alpha + \beta^{\mathrm{T}}x$$

such that x is assigned to the positive class if $\delta(x) > 0$. Algorithmically, the parameters (α, β) can be fit to the data either by maximizing the conditional likelihood $\sum_{i=1}^{n} \log p(y^{(i)}| x^{(i)}; \alpha, \beta.)$or by minimizing the 0–1 loss $\sum_{i=1}^{n}[I(I(\delta(x^{(i)}) > 0) \neq y^{(i)})]$, where the indicator function $I(.) = 1$ if its argument is true, 0 otherwise.

The perceptron is another simple discriminative classifier. To separate two classes y_1 and y_2, it computes a linear combination of its inputs. Each input variable is assigned a weight or coefficient; if the sum of these weighted inputs is above a given threshold, the example is assigned to class y_1, otherwise it is assigned to class y_2. Learning consists in finding the appropriate weights so that the resulting hyperplane (i.e., $(d-1)$-dimensional surface in d dimensions, for example, a line in 2 dimensions) effectively separates the classes. The perceptron learning algorithm starts with a random initialization of these weights and iteratively adjusts them until a prespecified criterion is met (e.g., error is below a given threshold). Since perceptron learning builds hyperplanes to separate classes, it fails whenever the data are not linearly separable. Many of the artificial neural networks (NNs) (Bishop, 1995) now available are extensions that overcome this limitation. For instance, multilayer perceptrons add hidden units with non-linear (e.g., sigmoid) activation functions in order to build arbitrary non-linear class boundaries and solve more complex classification problems.

Support vector machines (SVMs) (Vapnik, 1998) are a more recent and extremely powerful example of the discriminative approach. Their underlying principle (called structural risk minimization) defines the true risk or error of a classifier as the sum of the empirical (or training) error and a term that quantifies the capacity or complexity of the learned model. There is a trade-off between the two terms: overly simple models incur high-training error but increasing model complexity can entail overfitting and hurt generalization. To minimize generalization error, we need to attain the lowest empirical error with the lowest-capacity model suited to the available training data. For 2-class problems, it has been shown that the model, which meets this requirement is a hyperplane that produces the maximal margin of separation between the two classes. Such a hyperplane can be uniquely constructed by solving a constrained quadratic optimization problem; the solution can be expressed exclusively in terms of the data points that lie on the margin, the so-called support vectors. This technique can be applied even if the data are non-linearly separable; the basic idea is to transform the data *via* a non-linear mapping onto a higher dimensional feature space where they become linearly separable. Thus, a linear boundary in feature space is equivalent to a non-linear decision surface in the original input space. Remarkably, there is no need to actually perform this mapping and carry out the computations in high-dimensional space; the use of an appropriate kernel (e.g., polynomial) function allows us to compute the final decision function using dot products between patterns in input space. SVMs have achieved impressive results in many biomedical applications (Brown et al., 2000; Schölkopf, Guyon, & Weston, 2003; Schölfkopf, Tsuda, & Vert, 2004); introductory texts can be found in (Burges, 1998; Cristianini & Shawe-Taylor, 2000).

Decision trees (DT) and rules comprise a distinct sub-category of discriminative learners. From the point of view of knowledge representation, they can be qualified as logical (Langley, 1996) or non-metric (Duda, Hart, & Stork, 2000) approaches as opposed to the other methods described above. Moreover, they are sequential approaches (Quinlan, 1994) in the sense that they examine one variable at a time whereas the preceding learners consider all input variables simultaneously. To determine the order in which the variables should be considered, all decision tree algorithms have built-in feature selection strategies, as we shall see in ‘‘In-Context Variable Selection’’ subsection. Sequential learning methods are most appropriate for tasks which can be solved by exploring only a small subset of the available variables; ‘‘simultaneous’’ learners are best suited for tasks where variable interactions should be taken into account, for example, when variables taken individually are only weakly correlated with the class variable but are collectively relevant. Neither alternative is perfect for mass spectra based biomarker discovery, which requires finding the smallest variable set possible while maximizing sensitivity to variable interaction.

The simplest models in this category are single-node trees called decision stumps and single-condition rules. They build the simplest possible class boundaries, which are single axis-parallel lines. However, more elaborate DTs and rules can carve out regions of arbitrary complexity as assemblages of piecewise hyperrectangles in p-dimensional space. A decision tree is built by recursively partitioning the training data with the aim of maximizing the class homogeneity of the resulting subsets. At each node, the remaining data are further subdivided based on the values of a test variable. The selected variable is that which ensures the maximal reduction of class heterogeneity as measured by the Gini index $G(t) = \sum_i p_i(1 - p_i)$ in CART (Breiman et al., 1984) or by entropy $H(\mathbf{X}) = -\sum_{x \in x} p(x) \; \log p(x)$ in C4.5 (Quinlan, 1993). The recursion process continues until all terminal nodes are homogeneous or all variables have been used, after which the tree is pruned to avoid overfitting.

A decision rule is typically a conjunction of a number of conditions: if $\text{cond}_1 \wedge \text{cond}_2 \wedge \ldots \wedge \text{cond}_N$ then conclusion. A rule classifier can be built by recursive partitioning, that is, by building a decision tree, which is then reexpressed as a rule set in a straightforward fashion. A rule is simply a path from the root to a terminal node, and the tree itself is a disjunction over all these rules (paths). An alternative way of inducing decision rules is by set covering. In this approach, rules are created one at a time, and the examples covered by the new rules are removed from the training set. As with DTs, rule conditions are added successively as tests on the values of individual variables. Examples of set-covering rule induction methods are Ripper (Cohen, 1995) and logical analysis of data (LAD) (Boros et al., 2000).

C. Ensemble Approaches

Contrary to the single-model classifiers described in the previous subsections, aggregate classifiers comprise multiple models whose decisions are combined in some way in order to classify a new case. There are two main approaches to building aggregate models. Resampling-based methods generate multiple models by training a single learning algorithm on multiple random replicates or subsamples of a given dataset whereas heterogeneous ensemble methods (also called multistrategy methods) train several different learning algorithms on the same dataset.

Resampling-based ensemble methods, which have been applied to mass spectra include boosting and bagging. Both these methods achieve model diversity by running the same learning algorithm on different samples of the training data. In both, the number of iterations is fixed by the user, and a new case is classified by taking a simple or weighted vote among the base classifiers. The basic idea of boosting is to focus the learning process on the more difficult cases by iteratively reweighing the training cases. Initially all cases are equally weighted. At each iteration, a classifier is built and tested; the weights of all misclassified cases are increased and those of correctly classified cases decreased (Freund & Schapire, 1997). In bootstrap aggregation, more popularly known as bagging, the different training sets are generated by randomly drawing with replacement samples of the same size as the original training dataset. A given learning algorithm is applied to the different bootstrap replicates to produce a committee of *m* models, which assign a new instance to a class by a simple majority vote (Breiman, 1996). While bagging builds diverse models by randomly resampling training *instances*, the RandomForest algorithm grows an ensemble of DTs by randomly resampling *features*. The algorithm has two user-defined parameters: the size of the candidate feature set *F* and the number of iterations *T*. In the standard tree-building procedure, the feature to be tested at the current node is selected from all the remaining candidate features; in RandomForest, the selection process is restricted to a subset of *F* features drawn randomly drawn from the full candidate set. *T* different trees are thus built, and a final decision is reached *via* a majority vote on their predictions (Breiman, 2001).

While the above methods vary the training sets on which to apply a given algorithm, multistrategy approaches build heterogeneous ensemble classifiers by varying the algorithms to apply on a given training set. An early example of this approach is stacked generalization, whereby *K* base level models are built on the training data and their predictions on test samples input as training data to a metalevel learner, together with the actual class labels of these samples. The metalearner's task is to build a model, which will predict the outcome of a new sample based on the predictions of the base level learners (Wolpert, 1992). In other forms of multistrategy learning, the predictions of the different base classifiers can be combined without metalearning, for instance, *via* a simple or weighted vote.

D. Which Classification Algorithm?

There is an overwhelming number of classification algorithms which can be combined in an exponential number of ways. The question of which learning approach works well for a given classification problem is still an open question and will probably remain so for sometime. Different classification algorithms have their specific biases, which should match the problem structure, that is, the concept that governs class assignment. Unfortunately the problem structure is not known *a priori*; in fact it is precisely what should be discovered. Even in a circumscribed domain such as mass spectrometry, different learning algorithms could be appropriate for seemingly related problems, depending on the concept that underlies the data and how the features interact together to determine the class.

To illustrate the match or mismatch between the problem structure and the biases of the learning algorithms, we created three very simple artificial classification problems involving only two features. These were fed into two different classification algorithms, a linear discriminant, and a decision tree. In Figure 6, the leftmost column shows the training sets, the other two columns visualize the decision boundaries drawn by the two algorithms on the plane defined by the two features. The three problems are characterized by different types of feature interaction and feature relevance to the class label. Both features, m/z_1, m/z_2, are rescaled within the interval $[-1,1]$. In the simplest problem (a), only one feature (m/z_1) is relevant for classification. If this takes values within the intervals $[0.6,0.7] \cup [0.9,0.95]$ then the specimen belongs to one class, otherwise it belongs to the other class. Class distributions are multimodal. The decision tree algorithm approximates these class boundaries correctly, thanks mainly to its inherent feature selection strategy and the fact that it can capture multimodal distributions when the boundaries are orthogonal to the axes defined by the features. The linear discriminant, however, fails to find the correct decision surface due to multimodality (one of its main assumptions is that class distributions are unimodal). Problem (b) shows a more complicated situation where both features are relevant but the decision boundaries are not orthogonal to the axes. This is an easy problem for the linear discriminant but harder for the decision tree algorithm, which tries to approximate the decision surface piecewise, thus producing a staircase effect. The decision tree algorithm could in principle approximate the decision boundary given enough training examples. The third problem, (c), is the equivalent of a logical exclusive-or between the two features in determining the class label. Both features are relevant, but none, taken alone, is adequate to completely define the class. The decision tree manages to approximate the decision boundaries due to its divide-and-conquer local approach; this approximation could be improved provided enough examples were used for training. However, the decision tree's sequential feature selection mechanism could be misled completely if these two features and their discriminatory interaction were hidden in a much larger feature set.

To summarize, there is no universally superior learning algorithm. The question is not which algorithm is best overall, but rather under which conditions a given algorithm is appropriate for a given learning task. Among the factors to be considered are the complexity of the concept to be learned, the availability of domain knowledge, and the nature, quality, and distribution of the available data. Beyond the idiosyncrasies of individual problems and datasets, however, we can search for commonalities that characterize a clearly delimited task domain. For instance, high data dimensionality is a generic issue to be tackled in all applications involving mass-spectra classfication, whatever their specific objectives. Algorithms that can cope with high dimensionality, are therefore most appropriate for this task. However, such methods are extremely rare; an alternative solution consists in reducing dimensionality prior to the learning process.

IV. DIMENSIONALITY REDUCTION

After the preprocessing phase described in "Data Preprocessing" section, the mass spectra are ready to be mined. They have been denoised, aligned, normalized, and otherwise transformed to facilitate the modeling or learning task. In particular, the dimensionality of raw spectra has been reduced, often by several orders of magnitude. Such drastic reduction might still prove insufficient; if the number of variables is greater than the sample size, certain modeling algorithms like linear or quadratic

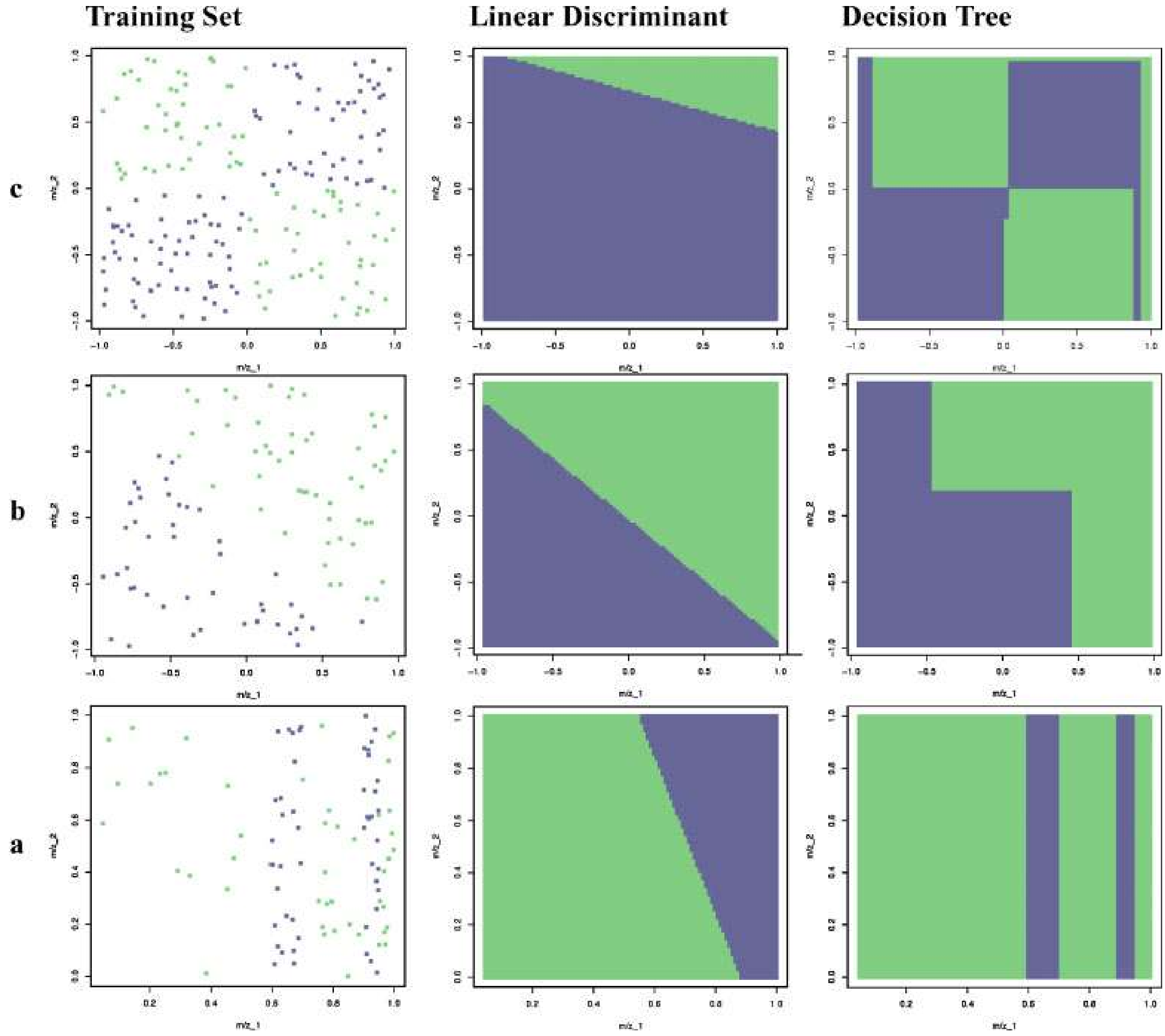

FIGURE 6. Three artificial learning problems with two features. Each problem (row) is represented by the training set (**left column**), the decision surface induced by a linear discriminant algorithm (**middle**), and the decision surface induced by a decision tree algorithm (**right**).

discriminant analysis will fail. In addition, we have seen that in diagnosis with biomarker discovery, selection of a small panel of *m*/*z* values is as important a goal as classification itself.

Dimensionality reduction methods can be classified into three main groups based on the perspective adopted to reduce dimensionality. Individual variable selection methods rank and select single variables assuming mutual independence among them. In-context variable selection methods also rank or evaluate individual variables, but do so in the context of others, that is, taking account of certain interdependencies among variables. Variable subset selection methods assess and select variables sets collectively, thus integrating all possible correlation or other forms of interaction among them into the evaluation function. Finally, variable transformation methods reduce dimensionality by constructing new variables as combinations of the old.

Another classification scheme is the distinction between filter, wrapper, and embedded methods. Filter methods perform dimensionality reduction as a preprocessing step to the learning phase, independently of the learning method. Wrapper methods wrap feature selection around the learning process and use the estimated performance of the learned classifier to select feature subsets; the utility of the selected variable set is tied to the learning method used in feature selection. Embedded methods are programmed as subroutines of the learner and are, therefore, inseparable from specific learning algorithms. Since the filter/wrapper distinction was introduced (Kohavi & John, 1997), there has since been a proliferation of new approaches, which defy classification under this scheme. Filter methods have been used as wrappers, and wrapper or embedded methods as filters, so that this distinction has become confusing at best. For this reason, we organize this section based on the first-classification scheme, which has the advantage of clarity.

A. Individual Variable Selection

Typically used as a filter, individual variable selection assumes mutual independence of all predictive variables. It relies on some scoring or ranking function to quantify variable relevance or discriminatory power; the final variable set is selected by defining a threshold on the computed scores or ranks. Many of the classical statistical tests and measures have been used to determine significant differences in variable importance. These tests rely on the same basic procedure to evaluate each variable: partition the sample according to classes (e.g., healthy *vs.* diseased), compute a test statistic of the variable for each class, and then check for significant differences in the values of this statistic. Statistics that have been used to rank mass spectral peaks are the *t*-statistic (Liu, Li, & Wong, 2002; Wu et al., 2003; Papadopoulos et al., 2004), the *F*-ratio (Liu, Li, & Wong, 2002; Wagner, Naik, & Pothen, 2003), and the χ^2-statistic (Liu, Li, & Wong, 2002; Rogers et al., 2003). Intuitively, the *t*-statistic quantifies differences between the class-conditional means of a variable whereas the *F*-statistic expresses the ratio of its between-class variance to its within-class variance. The χ^2 statistic measures the strength of association between two qualitative variables; to test a peak's association with the class variable, its intensity must be discretized or binned. In all three cases, the higher the value of the statistic, the higher the variable's rank. The Wilcoxon test ranks variables directly according to the absolute value of differences in their class-conditional means; it has been used for peak selection in (Kozak et al., 2003; Sorace & Zhan, 2003). The precise definitions and formulas of these different statistics and tests can be found in standard statistical textbooks.

Alternative variable ranking/selection criteria have been borrowed from information theory and technology. A well-known entropy-based criterion is the mutual information between a predictive and a class variable (Cover & Thomas, 1991), computed as the initial entropy of the class variable minus its entropy after observing the predictive variable. The difference quantifies the information about the class gained from observing the variable. For this reason, mutual information is also known as information gain in the Machine Learning community. It has been shown to be an effective variable ranking criterion in MS-based lung cancer prediction (Hilario et al., 2003). Another increasingly popular criterion from information technology is the AUC or area under the receiver operating characteristic (ROC) curve. The ROC curve is a plot of the true *versus* false-positive rates associated with all possible thresholds for classifying a sample as positive (see "Overview of Model Evaluation Methods and Matrics" subsection for a more detailed explanation). It has been shown that, for certain distribution patterns, the AUC is a more reliable indicator of a biomarker's ability to discriminate between cancer and control (Pepe, 1995). The AUC has been used to rank peaks in prostate cancer detection prior to learning with individual or boosted DTs (Adam et al., 2002; Qu et al., 2002).

B. In-Context Variable Selection

The main advantage of individual variable selection is its efficiency, since it requires no more than computing p_0 scores (where p_0 is the original dimensionality or number of raw variables). However, it has a number of drawbacks: it cannot detect redundant or correlated variables, or variables which are irrelevant by themselves but highly discriminatory in combination with others. To alleviate these shortcomings, machine-learning research has given rise to novel variable selection algorithms which we shall call *in-context variable selection methods* because they take (limited) account of variable interaction while ranking/selecting individual variables.

Like most in-context variable selection methods, those described in this section are filters unless specified otherwise. Relief-F (Kononenko, 2004), an extended version of Relief (Kira & Rendell, 1992), computes the relevance of each predictive variable *via* a method based on KNN. For simplicity, we describe the algorithm for the case of two classes. The algorithm maintains a relevance score for each variable. At each iteration, it picks a case at random and identifies the case's nearest neighbor from the same class and its nearest neighbor from the other class. It then adjusts feature weights to reward features, which discriminate neighbors from different classes and penalize those which have different values for neighbors of the same class. The result is an estimate of features' merit in circumscribed local regions of the instance space. This allows Relief-F to take into account feature interaction, that is, their conditional dependence given the class, whereas other methods lose sight of such dependencies as an effect of averaging over all the training instances. Since Relief-F is a self-contained feature selection method, it can and has been used as a filter for a variety of learning algorithms. It was been used for *m/z* value ranking and selection in lung cancer diagnosis, where it outperformed purely univariate variable selection methods such as that based on information gain (Hilario et al., 2003). Relief was also shown to outperform a variety of other feature selection methods in a comprehensive comparative study by Guyon et al. (2003).

To select individual variables while integrating the impact of other variables, Wu et al. (2003) use a measure of variable importance given by the RandomForest (RF) learning algorithm (Breiman, 2001) (Subsection V.C.1). This measure is derived by averaging over several iterations of the following process: a classifier is built to compute a reference accuracy; the value of each variable is then permuted randomly in turn, and the decrease in performance with respect to the reference accuracy measured. The magnitude of the decrease is taken as a measure of the discriminatory power of that variable: the higher the decrease, the more important the variable's contribution to classifier performance. In a filter set-up, Wu et al. used RF scores to select a subset of 15–25 peaks, which were deemed most discriminatory between diseased and healthy samples. The selected peak set was used to train diverse classifiers such as linear discriminants, nearest-neighbor classifiers, DTs, SVMs, and RF itself. In general, all these learning algorithms obtained better performance on variable sets filtered using RF rather than the *t*-statistic.

A similar in-context feature selection technique was used in a study on stroke diagnosis (Prados et al., 2004). Three variable ranking methods were used: one based on information gain, Relief-F, and one SVM-based technique. In this last approach, a linear SVM classifier was trained using the set of peaks produced in the preprocessing phase. The weights of the variables in the linear classifier were used to rank them in decreasing order of importance, and the p' top-ranked variables (where p' is a user-selected parameter) were retained for the actual training phase. Though the hyperplane weights played the same role as information gain in ranking the variables, the manner in which these criteria were computed spells the difference between single variable selection and in-context variable selection. Information gain is computed as the mutual information between the class variable and a predictive variable, taken in isolation from all other predictors. On the other hand, all variable weights are computed simultaneously in building an SVM classifier, so that the weight of each variable is computed in strict interdependence with those of all others.

Genetic algorithms have also been used for variable ranking in experiments on ovarian cancer (OC-HC4). Working on the original 15,154 variables (*m*/*z* values) of the mass spectra, Li et al. (2004) apply GAs to select 10,000 different subsets of 20 variables using KNN ($K = 5$, consensus rule) as the fitness function. A subset was considered discriminative if it led to an accuracy of at least 90%. The 15,154 variables were then ranked based on the number of times each was selected into the 10,000 discriminative subsets. Finally, this ranked list was used to train nearest neighbor classifiers using successively increasing numbers of top-ranked variables (Subsection V.A.2). While dimensionality reduction is based on individual variable ranks, the ranking criterion does not examine each variable separately, but considers classification decisions made in interaction with 19 other variables each time; the procedure is thus a case of in-context variable selection.

Certain methods that rank and select variables in isolation when used as filter become in-context variable selection methods when embedded in learning algorithms. An example is the mutual information criterion used in C5.0 under the name of information gain. Though the criterion is explicitly applied to candidate variables taken individually, interaction with previously selected variables in the ancestor nodes is implicitly taken into account. Stepwise discriminant analysis embeds forward, backward, or bidirectional feature selection into linear discriminant analysis; the backward and bidirectional variants are more sensitive to variable interaction than forward selection. More recent methods embed variable selection or variable weighting techniques into linear classifiers. Examples are Yanasigawa et al.'s (2003) modification of Tukey's compound covariate method, Tibshirani et al.'s (2004) shrunken centroids, and Yasui et al.'s (2003) boosted univariate discriminants; as they are inextricable from the learning process, these methods will be discussed in Section V.

C. *Variable Subset Selection*

Variable subset selection requires evaluation criteria that are specifically adapted to groups of variables as a whole. It also introduces an additional difficulty: the number of possible variable subsets increases exponentially with the number of variables. This precludes exhaustive search for all but trivial datasets; heuristic or stochastic search strategies are needed. GAs are increasingly popular stochastic strategies while forward or backward selection methods are examples of heuristic search. Forward selection starts with an empty variable subset S and selects the variable that maximizes a predefined scoring function. Thereafter, it selects from the remaining variables the one which, added to S, maximizes the score of the resulting subset. The process continues until a predefined criterion is met, for example, until no single variable addition improves the merit of the subset. Backward elimination proceeds in the reverse direction; it starts with the full variable set and at each step removes the variable whose elimination yields the highest score for the remaining subset.

A number of variable subset selection strategies have been used as filters prior to the learning process. Forward selection has been used with different scoring functions in two mass-spectral applications. In one experimental study on prostate cancer detection (Qu et al., 2003), a discrete wavelet transform reduced the initial mass set to 1,271 variables. Stepwise forward selection was then applied to find a subset that maximized the Mahalanobis distance between the cancer cases and controls. Intuitively, the Mahalanobis distance quantifies the separation between two groups in terms of the Euclidean distance between their centers (group means), normalized by their covariance to correct for the effect of correlated variables. Mathematically, it is computed as $D_{\mathrm{M}} = (\bar{X}_1 - \bar{X}_2)^{\mathrm{T}} S^{-1} (\bar{X}_1 - \bar{X}_2) D$, where S is the unbiased estimate of the covariance matrix. This method resulted in a subset of 11 variables, which were then used to build a linear discriminant model.

Correlation-based feature selection (CFS) also relies on forward selection. Its evaluation criterion is based on the idea that good variable subsets contain variables highly correlated with the class yet uncorrelated with each other (Hall & Holmes, 2003). The merit of a variable set is directly proportional to the mean strength of correlation between the member variables and the class, and inversely proportional to the mean correlation among the variables themselves. Correlation between variables is measured in terms of their symmetrical uncertainty, a normalized form of mutual information. Use of this criterion requires preliminary binning of continuous variables. CFS uses stepwise forward selection to find a variable subset that maximizes the merit criterion. CFS has been found to yield best performance in a comparative study of variable (subset) selection methods for mass-spectra-based ovarian cancer diagnosis (Liu, Li, & Wong, 2002); however this should be taken with caution due to a technical flaw in the Liu et al.'s experimentation methodology (Subsection VI.B.1).

In Baggerly et al. (2003), mass spectra preprocessing yielded samples with 506 peaks. A much smaller variable set was needed to build a classifier using Fisher's linear discriminant analysis. Search was restricted to subsets of size $N = 1$ to 5 of this initial peak set. A peak set was considered optimal if it maxmized the Mahalanobis distance between the lung cancer and control groups. Exhaustive search was used for $N = 1$ and 2. For $N = 3$ to 5, 50 GA runs were performed using different initial populations of 200 sets of N peaks; 250 generations were evolved before halting.

Variable subset selection has been shown to be most effective in a wrapper setup, targeted to a specific learning algorithm. In a study on ovarian cancer diagnosis and biomarker discovery, mass spectra preprocessing was limited to baseline subtraction and intensity scaling to [0,1]; no dimension reduction was performed. The input to the data mining process thus contained 15,154 *m*/*z* values or variables (Petricoin et al., 2002). GAs were used to evolve an initial population of 1,500 sets, each containing between 5 and 20 *m*/*z* values, into meaningful biomarker patterns. To determine the discriminatory power of each set, the training samples were expressed in terms of its variables and used to generate a self-organizing map. Self-organizing maps are NNs which cluster input samples in a way that preserves the topology of the input space, with the result that the distances between samples in the map reflect their actual relative distances (Kohonen, 1995). A variable set was deemed fit if it produced a map with homogeneous cancer and control clusters. The sets that passed the fitness test were used to spawn new variable sets through crossover and mutation. The learning process halted after 250 generations or when a map was found that perfectly separated the cancer and control cases.

In Alexe et al. (2004), feature subset selection is wrapped around the LAD (Logical Analysis of Data) algorithm, a set-covering method for rule induction. The process starts with an initial pool of *k* features selected from the raw set on the basis of five individual feature-scoring criteria (e.g., entropy, Pearson correlation with the class variable). The feature pool is then reduced iteratively; at each iteration, a subset composed of the half top-ranked features is used to train a LAD classifier. If classification accuracy using the reduced pool is higher than that obtained with the parent pool, the reduced pool becomes the current pool, and iteration continues. Otherwise an attempt is made to find a better performing feature set by generating variants of the current reduced pool. If such a feature set is found, it becomes the basis for further feature reduction, otherwise the iteration process stops and returns the current feature pool. The feature selection process is, however, marred by a methodological inconsistency: while cross-validation is used to select the most appropriate feature subset, the initial feature set is selected on the entire dataset prior to cross-validation, thus resulting in the use of test samples for what should be considered an integral part of the training process (Subsection VI.B.1).

Recursive feature elimination or RFE (Guyon et al., 2002) is a feature subset selection method, which was applied in gene expression analysis to identify biomarkers for cancer diagnosis. In RFE a given feature set (initially the set of all variables, scaled if necessary) is used to train a linear SVM; the features are ranked in decreasing order of their (squared) weights in the hyperplane, and the lowest ranked features are eliminated. The algorithm generates a set of nested feature subsets, one for each iteration. The selected subset is that which minimizes $\mathrm{score}(F) = \mathrm{err}(\mathrm{SVM}_{|F}) + |F|/N$, where $\mathrm{err}(\mathrm{SVM}_{|F})$ is the error of the SVM classifier trained on feature subset F, $|F|$ is the size of F, and N is the total number of original features. The second term penalizes large feature sets. In the original version of RFE, one feature was eliminated at a time; to reduce the number of iterations, a natural variant consisted in eliminating the *t*% lowest ranked features. Multiple runs were needed to explore different values of *t*. Instead of selecting a single threshold/subset, Jong et al. (2004) proposed two ways of combining feature subsets produced from these multiple runs. First, Join gathers all the features occurring at least *x* times in the different feature subsets and trains a single classifier based on the resulting feature set. Second, Ensemble builds a separate classifier for each feature subset and classifies a test sample by a majority vote of the committee of classifiers. These methods were tested on ovarian and prostate cancer diagnosis (Subsection V.B.3).

D. Variable Transformation

Variable (subset) selection reduces data dimensionality by selecting from a preexisting set of variables. Variable transformation techniques create new variables by combining or transforming the old. Dimensionality is reduced if a small number of these new variables can replace the old without loss of discriminating information.

Many of these variable transformation/extraction methods are commonly used during the preprocessing stage (Section II): examples are spectral transforms such as Fourier, wavelet, or kernel convolution transforms. In addition, principal components analysis (PCA) is a statistical technique for reexpressing raw variables in terms of new variables, called components, which are linear combinations of the original variables. These components are computed through an eigenvalue decomposition of the covariance matrix of the original data, with the eigenvalues (and their corresponding eigenvectors) ordered in decreasing order of magnitude. Though theoretically there can be as many components as original variables, very often a much smaller set of components can explain most of the variability in the data. Substantial dimensionality reduction can thus be attained by describing the data in terms of a few principal components. (Lilien, Farid, and Donald 2003) used PCA to reduce the raw *m*/*z* ratios of three ovarian cancer datasets and a prostate cancer dataset (around 15,000–16,000 variables) in view of classification by linear discriminant analysis. In compliance with a precondition of LDA, the number of components was selected to be lower than the number of examples available for each problem.

A study on human African trypanosomiasis (Papadopoulos et al., 2004) compared PCA and *t*-tests as tools for dimensionality reduction. These methods resulted in a reduced set of 41 principal components and 19 peaks, respectively. Each reduced variable set was used to train classifiers based on DTs, NNs, and GAs, as well as a combined model, which classified test cases *via* a majority vote of all the three base models. PCA led to higher accuracy than *t*-statistic-based variable selection on all learning methods used except DTs, where both achieved equivalent performance (Subsection V.C.2).

Partial least squares (PLS) projection to latent structure can be viewed as the supervised counterpart of PCA. It extracts latent variables as linear combinations of the original explanatory variables such that most of their association with the response variable is explained. Dimensionality is reduced when the first few linear combinations of predictors explain most of the association with the response. In a study on lung cancer diagnosis, PLS was used as a filter before classification *via* logistic regression or linear discriminant analysis. The number of factors retained was chosen on a separate tuning set but was not reported (Purohit & Rocke, 2003).

TABLE 2. Datasets which have been made public and used in classification experiments by different teams

Disease	Name	Tech	Sample	Data

Lung cancer	LC-Duke	MALDI-TOF	serum	D=24
				C=17

Ovarian cancer	OC-H4	SELDI	serum	D=100
				C= 100
				B=16

	OC-WCX2a	SELDI	serum	D=100
				C= 100
				B=16

	OC-WCX2b	SELDI	serum	D=162
				C = 91

	OC-NWUH	MALDI-TOF	serum	D=47
				C=42

Prostate cancer	PC-CPPD	SELDI	serum	D=69
				C=253

	PC-EVMS	SELDI	serum	D=197
				B=93
				C=96

The datasets are referred to in the text by their identifiers as shown in the Names column. The Data column gives the number of spectra for each class or disease state: *D*, diseased, *C*, controls, *B*, benign.

V. CLASSIFYING MASS SPECTRA FOR DIAGNOSIS AND BIOMARKER DISCOVERY

This section surveys work on diagnosis and biomarker discovery following the taxonomy of classification approaches outlined in "A Bird's Eye View of Classification Methods" section.[1] Much of this work concerns mass spectral data that have been made available by the originating institutions; for conciseness these datasets are summarized in Table 2 and will be referred to in the remainder of the text by their short names, given in column 2 of the table.

A. Generative Approaches

1. Linear and Quadratic Discriminant Analysis

Discriminant analysis (Subsection III.A) is one of the most widely used approaches to mass spectra classification in spite of the HDSS problem. Discriminant analysis typically relies on the covariance matrix, which becomes singular when the number of variables p is greater than the number of examples n. To guarantee a non-degenerate solution, it is necessary that $p \leq (n-k)$, where k is the number of classes; in addition, to avoid overfitting, it is recommended that $n \gg p$, for example, $n \geq 2p$ (Tukey, 1992), $n \geq 5p \ldots 10p$ (Somorjai, Dolenko, & Baumgartner, 2003).

The most popular way of having $p < n$ is by using any of the dimensionality reduction methods described in "Dimensionality Reduction" section prior to discriminant analysis. Working on an ovarian cancer dataset with only 89 samples (OC-NWHU, Table 2), Wu et al. (2003) selected variable sets of size 15 and 25 using two alternative measures, the *t*-statistic and Random-Forest scores (Subsection IV.B). They then applied a number of classification algorithms including LDA and QDA. On the 15-variable sets, LDA was second only to SVM in classification accuracy, but this advantage diminished on the 25-variable sets (however, see "Generalization Performance" subsection for remarks on their evaluation methodology). In addition, the use of 25 variables to build quadratic discriminants often resulted in singular covariance matrices when a resampling strategy was followed. Similar behavior was observed on the Duke lung cancer dataset ($n = 41$): the leave-one-out cross-validation error of LDA and QDA on a 4-peak set almost tripled when a 13-peak set was selected; degradation was worse for QDA since the covariance matrices became nearly singular on very small samples (e.g., only 17 cases for the lung cancer group) (Wagner, Naik, & Pothen, 2003).

LDA has been coupled with variable subset selection in lieu of individual variable selection. As discussed in "Variable Subset Selection" subsection, Qu et al. (2003) used the Mahalanobis distance to select the most discriminatory set composed of 11 wavelet coefficients. They then applied Fisher's linear discriminant to project this 11-dimensional vector onto a hyperplane which allowed for maximal separation of the prostate cancer and control groups in their 248-case training sample (PC-EVMS, Table 2). The resulting classifier attained 96.7% sensitivity and 100% specificity on an independent test set of 45 samples. Similarly, Baggerly et al. (2003) applied linear discriminant analysis to draw a hyperplane for each of the 1- to 5-peak sets selected by GAs from MALDI-TOF spectra in a study on lung cancer (LC-Duke, Table 2).

Variable transformation techniques have also been used to yield a small variable set appropriate for discriminant analysis. To build linear discriminants for one prostate cancer dataset (PC-EVMS, Table 2) and three ovarian cancer datasets (OC-H4, OC-WCX2a, OC-WCX2b, Table 2), Lilien, Farid, and Donald (2003) used PCA to transform the *original* p-dimensional mass-spectral space (e.g., $p = 16{,}382$ in PC-EVMS) into an $(n-k)$-dimensional space (e.g., $n = 386$, $k = 3$ for PC-EVMS). Results of this PCA-LDA learning configuration will be discussed further in "Evaluation Results: A Comparative Study" subsection. The supervised counterpart of PCA, PLS, has also been used as a filter for LDA on the LC-Duke dataset (Table 2). In leave-out-out cross-validation experiments, LDA achieved significantly higher predictive accuracy when variables were filtered with PLS than with principal components regression (Purohit & Rocke, 2003).

Rather than filtering variables prior to discriminant analysis, other researchers have used variations on the discriminant analysis algorithm itself to circumvent the $p \gg n$ problem. The best known of these is stepwise discriminant analysis, which can be viewed as the straightforward embedding of forward, backward, or bidirectional variable subset selection into the discriminant algorithm. Sorace and Zhan (2003) built several diagnostic models for ovarian cancer (OC-WXC2b, Table 2) by combining stepwise discriminant analysis with a variable selection filter based on a two-sided Wilcoxon test. Among the peaks which had a P-value $< 10^{-6}$, 100 were selected, sorted, and binned by requiring a separation of at least 1 *m/z* value to start the next bin. In one case, this procedure produced 12 bins; the peak site with the lowest P-value in each bin was selected. Stepwise discriminant analysis was applied to these 12 peaks and built a

[1] This excludes the large body of attempts to classify mass spectra based solely on differentially expressed peaks identified by standard statistical significance tests such as *T*-stat, Wilcoxon, Mann–Whitney.

diagnostic model using 7 peaks. Three different models were built by varying the lower bound on the range of *m/z* values to retain; tests on an independent holdout set led to the intriguing observation that perfect classification was achieved by models involving peaks from the low molecular weight range, generally taken to represent noise.

A variant of LDA is diagonal linear discriminant analysis (DLDA), which assumes mutual independence of the explanatory variables, resulting in a diagonal covariance matrix $\Delta = \mathrm{diag}(\sigma_1^2, \sigma_2^2, \ldots, \sigma_p^2)$ for all class densities. Closely linked to DLDA are methods that can be viewed as weighted voting of linear univariate classifiers. An example is Tukey's (1992) compound covariate method which was applied by Hedenfalk, Duggan, and Chen (2001) to gene expression analysis. This method uses a standard *t*-test at level α to select a set of genes and forms a linear classifier with these genes weighted by their *t*-statistic:

$$H(x*) = \sum_{i \in S(\alpha)} t_i \left(x_i^* - \frac{\bar{x}_{i1} + \bar{x}_{i2}}{2} \right)$$

where x^* is the example to be classified, $S(\alpha)$ the set of genes with a significant *t*-statistic at level α, and t_i the *t*-statistic for gene *i*:

$$t_i = \frac{\bar{x}_{i1} - \bar{x}_{i2}}{s_i \sqrt{1/n_1 + 1/n_2}}$$

where s_i is the pooled standard deviation for gene *i*. If $H(x^*) > 0$, the sample x^* is assigned to class 1, otherwise to class 2. In other words, an example is assigned to the class with the nearest centroid, the distance to the centroids being weighted by the summed *t*-statistics of the discriminatory variables. In a study of mass-spectra-based diagnosis of lung cancer, Yanasigawa et al. (2003) used a variant of the compound covariate method where the *t*-statistic was replaced by a battery of six different statistical tests including Kruskal–Wallis and Fisher's exact test. Eighty-two peaks that met 3 of these 6 criteria were selected from a training set of 50 samples; the resulting compound covariate model correctly classified all 43 samples of a blinded test cohort as either tumor or normal. However, the method did less well in discriminating histological subgroups such as adenocarcinoma *versus* large-cell (94% test set accuracy) or mediastinal nodal involvement (75% test set accuracy).

Stepwise discriminant analysis and the compound covariate method use hard thresholds to select or eliminate variables, and hence often exhibit high variance. In contrast coefficient shrinkage methods, which assign continuous weights to variables, operate less abruptly and eliminate variables only when the coefficient is reduced to 0. Such an approach was developed by Tibshirani et al. (2002) as a modification of the nearest-centroid method. In the 2-class case, the discriminant function is defined as:

$$S(x*) = \log \frac{\pi_1}{\pi_2} + \sum_{i=1}^{p} \tau_i \left(x_i^* - \frac{\bar{x}'_{i1} + \bar{x}'_{i2}}{2} \right)$$

with normalization factor τ_i defined as,

$$\tau_i = \frac{\bar{x}'_{i1} - \bar{x}'_{i2}}{(s_i + s_0)}$$

where $\bar{x}'_{i1}$ and $\bar{x}'_{i2}$ are possibly shrunken means, and s_0 is a value common to all genes, for example, the median value of the s_i over the set of genes. Thus shrunken centroids use a standardized squared distance in contrast to LDA which uses the Mahalanobis distance to class centroids. The Mahalanobis distance becomes problematic in an HDSS context, as it uses the pooled within-class covariance matrix to normalize deviations from the mean; when $p \gg n$, the covariance matrix becomes singular. To circumvent this problem, the shrunken centroids method assumes a diagonal within-class covariance matrix. In addition, centroid shrinkage by soft thresholding introduces a way of reducing the number of variables. The soft threshold function is $s(t, \Delta) = \mathrm{sign}(t)(|t| - \Delta)_+$, where $t_+ = t$ if $t > 0$ and otherwise 0. If $|t| \leq \Delta$, *t* is set to 0, otherwise it is moved closer to 0 by the quantity Δ, a user-tuned parameter (typically by cross-validation). If the centroids for a given variable are shrunken so that they coincide for all classes, the variable is in effect eliminated.

The shrunken centroids approach, applied to gene expression analysis in Tibshirani et al. (2002), was integrated into the "peak probability contrasts" method for mass-spectra-based diagnosis of ovarian cancer (OC-NWUH, Table 2). Peak extraction from raw mass spectra yielded 14,067 *m/z* sites; these were then binned *via* hierarchical clustering to identify 192 peaks that were common to all spectra. A split point $\alpha(i)$ was estimated for each peak *i* so as to maximize $|p_{i2}(\alpha) - p_{i1}(\alpha)|$, the difference in the proportion of samples from each class having peaks higher than $\alpha(i)$. For a given class *k*, p_{ik} was set to $p_{ik}(\alpha(i))$. A new spectrum can then be encoded as a binary vector based on the set of common peaks and their individual split points. A peak in the new spectrum is deemed to correspond to a common peak if its center lies within 0.005 of the position of the common peak, and its value set to 1 if it is higher than the split point of the common peak, otherwise to 0. The result is a binary vector, which can then be compared to the probability centroid vectors of each class and assigned to the class that is closest in overall squared distance.

While stepwise discriminant analysis, compound covariates, and shrunken centroids embed variable elimination into the classification process, Lee et al. (2003) merge variable transformation and classification through the use of PLS. Working on the LC-Duke dataset (Table 2), they first used a wavelet transform to reduce the initial 60,000 *m/z* values of the raw spectra to 545 wavelet coefficients. These were the inputs to PLS which produced a 2-component discriminant model to separate lung cancer cases from controls. Each component was a linear combination of wavelet coefficients, which were then inverse-transformed to the original variates. *m/z* ratios corresponding to large PLS coefficients can be used to investigate proteins that may possibly be upregulated or downregulated in diseased specimens.

2. *Non-parametric generative approaches*

Discriminant analysis assumes a Gaussian distribution although it has been shown to work in practice in a much broader range of cases. Non-parametric generative approaches make no such assumption; examples that have been used for mass spectra are KNN, kernel density estimation (Subsection III.A), or versions of Naive Bayes which do not assume a specific probability distribution for continuous variables[2]. KNN was explored in the context of two studies on ovarian cancer. Using a training sample from the OC-H4 dataset, Li et al. (2004) applied the GA-based process described in Subsection IV.B to rank the 15,154 *m/z* values of the raw spectra. They then explored successively increasing variable subsets consisting of the *i* top-ranked variables, for $i = 1$ to 100, applying KNN ($K = 5$) each time to measure classification performance on a separate test set. Across 50 iterations of the whole GA/KNN process, it was observed that performance grew initially as the variable subset size was increased, and then reached a plateau at size 10, with average classification performance on the test set hovering around 97% (93–100%). In findings reminiscent of those of Sorace and Zhan (2003) on another ovarian cancer dataset, the authors observed that all 10 top-ranked variables belonged to the <500 Da region, and then repeated the same analysis after excising this low-molecular weight region. Performance on the new, supposedly less noisy 10-variable set fell to 90% (78%–96%). Explaining these observations remains an open problem.

Zhu et al. (2003) built a classifier for ovarian cancer diagnosis (OC-WCX2a, Table 2) after a two-step dimensionality reduction process. First, they selected discriminatory variables using a *t*-test with a significance level adjusted for multiple comparisons. The 563 variables that passed the test were ranked and subjected to further reduction. The second step was variable subset selection wrapped around KNN ($K = 5$) using the Mahalanobis distance to identify nearest neighbors. Starting with the top-ranked variable, they successively added the next top-ranked variable until KNN classification performance reached a plateau. This resulted in a final variable set of 18 *m/z* values. The 18-variable KNN classifier achieved 100% accuracy on a holdout subset of the OC-WCX2a dataset as well as on an independent dataset, OC-WCX2b (Table 2).

Non-parametric generative models have been compared with other approaches to mass spectra classification. In a comparison involving five different variable selection schemes and four learning algorithms, Naïve Bayes and KNN displayed roughly equivalent performance in ovarian cancer diagnosis (OC-WCX2b, Table 2); both did clearly better than C4.5 but were outperformed by SVM on variable sets of size 17–20 (Liu, Li, & Wong, 2002). On a different ovarian cancer dataset (OC-NWUH), KNN ($K = 1$ and 3) also outperformed LDA and QDA on variable sets of size 15 and 25 selected by RF (Subsection IV.B). On the 25-variable set, 1NN achieved the highest accuracy along with the RF classifier, which had the undeniable advantage of having preselected variables adapted to its learning bias (Wu et al., 2003). On the Duke lung cancer data, however, 6NN achieved significantly lower accuracy than LDA and QDA on a 4-peak set selected using the *F*-statistic, while kernel density estimation using a Gaussian kernel showed equivalent performance with QDA. Results of these two comparative studies should however be taken with caution due to methodological shortcomings discussed in Subsection VI.B.1.

B. Discriminative Approaches

1. Logistic regression

Multivariate logistic regression was used in two similar studies on cancer diagnosis. In a study on ovarian cancer based on SELDI mass spectra, dimensionality reduction was performed in two alternative ways (Rai et al., 2002). The first approach used unified maximum separability analysis (UMSA), a variant of classical discriminant analysis which projects the training samples onto a 3-dimensional component space. Components are linear combinations of the original variables/peaks determined to achieve maximum separation between cancer and control cases. The individual variables were then ranked according to their contribution to the separation of the two groups, and seven peaks were thus selected. In the second approach, CART was used to grow a decision tree, which selected two peaks, in fact a subset of the UMSA-selected set. Multivariate logistic regression was then applied to build a classifier using these two common peaks at 60 and 79 kDa. Finally another classifier was built based on the combination of these two peaks with CA125, the tumor marker traditionally used for ovarian cancer diagnosis. This classifier produced improved performance over those based on CA125 alone or the two selected peaks alone. However, these results are subject to caution since performance was measured over all specimens, including the training data. The second study (Li et al., 2002) focused on breast cancer diagnosis and followed a strategy very similar to that described by Rai et al. Peak extraction using Ciphergen software yielded 147 qualified mass peaks ($S/N > 5$). The UMSA procedure, followed by stepwise selection, reduces these 147 peaks to a final set of 3 candidate biomarkers at 4.3, 8.1, and 8.9 kDa. A logistic regression classifier built from these peaks achieved 93% sensitivity and 91% specificity, averaged over 20 evaluation runs using a 70%–30% train-test split.

2. Neural networks

An investigation on renal cancer carcinoma (RCC) used a dataset composed of 106 samples (48 RCC, 38 healthy controls (H), and 20 benign cases (B)) to train different multilayer perceptrons (Rogers et al., 2003). Peak detection and clustering on the raw data yielded a set of 368 variables/peaks, which was sorted in order of decreasing relevance as measured by the χ^2 criterion. The resulting data were then encoded using boolean (peak presence/absence) or continuous features (peak signal intensities). Different fully connected multilayer perceptrons were built with varying numbers of the top-ranked variables. All models had five hidden units with a sigmoid activation function. Weights were randomly initialized to values in $[-1, +1]$, and back-propagation training was pursued until a limit of 100 epochs or an error of 0 was attained. To overcome chance results due to randomness, each model was initialized ten times, and the average performance over these ten runs reported. On one

[2] Naïve Bayes can be parametric or non-parametric depending on how probabilities of continuous variables are computed. Non-parametric versions include those which discretize continuous variables or which use kernel density estimation methods; parametric implementations assume a specific probability distribution, for example, Gaussian, for continuous variables.

blinded test set (12 RCC *vs.* 11 healthy controls), NNs using both boolean and continuous features were able to discriminate RCC from healthy controls fairly well. On the RCC *versus* benigns/controls task (12 RCC *vs.* 20 benigns/controls), performance degraded for models trained on boolean inputs but remained essentially the same for those with continuous features. Results concerning model stability are reported in Subsection VI.B.

Multilayer perceptrons were also used in a study aimed at detecting hepatocellular carcinoma (HCC) in patients with chronic liver disease (CLD) (Poon et al., 2003). The data included serum samples from 38 patients with HCC at various stages and 20 patients with CLD (controls). Preprocessing of raw mass spectra yielded 2,384 candidate peaks, which were reduced by significance analysis of microarray (a technique borrowed from gene expression analysis (Tusher, Tibshirani, & Chu, 2001)) to a set of 250 differentially expressed peaks. Intensities at these peaks were normalized and fed into a multilayer perceptron with 250 inputs, 7 hidden units, and 1 output unit. The output was a diagnostic score between 0 and 1, with 0 = CLD and 1 = HCC. The NN was trained using standard backpropagation, with the learning rate and momentum automatically selected by the software used. Training halted when the error <0.02 or when the number of epochs reached 300. Generalization error was estimated using tenfold cross-validation. ROC analysis of estimated errors showed that NN diagnostic scores were useful in differentiating HCC and CLD cases. However, NN interpretability remains a non-trivial problem; though sensitivity analysis can be used to determine which of the 250 model variables play a major role in diagnosing HCC, the presence of bias and hidden units complicates the task of determining the precise nature of interactions between the peaks.

3. Support vector machines

Support vector machines (Subsection III.B) have been applied extensively to mass spectra, both for classification and for dimensionality reduction. To filter the variable set prior to SVM learning for prostate cancer diagnosis (PC-CPPD, Table 2), Jong, Marchiori, and van derVaart (2004) evolved a large number of variable sets using GAs with SVM accuracy as the fitness function, and then selected features that were present in more than ten runs. This led to the selection of 47 features for SVM training. In another set of experiments, RFE was run several times with different thresholds, and the resulting variable subsets were combined according to the Join and Ensemble approaches described in Subsection IV.C. To train the SVM classifier on the OC-H4 data, linear SVM was used with the regularization parameter *C* set to 10, and the weights of the diseased and control cases set to 10 and 0.5, respectively, to compensate for class imbalance. On the PC-CPPD dataset, which is far more skewed that the ovarian cancer dataset, *C* was set to 1, and its weights for diseased and controls set to 1,000 and 0.005, respectively. These two approaches led to significantly higher sensitivity rates than use of the full feature set or simply selecting the variable subset which minimized error on a tuning set.

SVMs have also been used in conjunction with other filter methods like variable subset selection methods such as CFS (Liu, Li, & Wong, 2002) or individual variable selection based on criteria such as information gain (Prados et al., 2004), the *F*-statistic (Wagner, Naik, & Pothen, 2003), χ^2, and entropy (Liu, Li, & Wong, 2002). Whichever feature selection method is used, and despite methodological caveats formulated in Subsection VI.B.1, there is widespread agreement on the behavior of SVMs for mass-spectra classification: SVMs are competitive with top-performing algorithms when the number of features is very small; as dimensionality increases, their advantage over all other methods becomes more pronounced. For instance, on a 4-variable version of the LC-Duke lung cancer dataset, five algorithms (LDA, QDA, KDE, KNN, SVM) displayed error rates varying between 10% and 17%, with linear SVM rating 15%. On the 13-variable version of the same dataset, however, all other algorithms displayed error rates between 27% and 34%, far above linear SVM's 2% (Wagner et al., 2003). In another comparative study involving SVM, Naïve Bayes, KNN, and C4.5, Liu, Li, and Wong (2002) found that SVMs were constantly in the top two ranks for variable set sizes between 17 and 20, independently of the variable selection method used. However, on the 15,154 raw variables of the OC-WCXb dataset (Table 2), the error rates of KNN and Naïve Bayes tripled at the least, that of C4.5 remained stable at 3.5% while SVM attained perfect accuracy. Jong, Marchiori, and van derVaart (2004) reported the same performance for SVM on the same dataset. This resilience of SVM to high-dimensional data makes it one of the most appropriate techniques for mass spectra classification.

4. Decision trees and rules

As explained in Subsection IV.B, DTs and rules are sequential approaches, which use embedded feature selection methods to determine the next variable to test on. Hence they are relatively resilient to high dimensionality and have been used in mass spectra classification without a preliminary variable selection phase, even in cases where $p \gg n$. In a study on renal cell carcinoma (Won et al., 2003) for instance, 36 SELDI mass spectra were preprocessed using Ciphergen's built-in software PBS to select 119 peaks. No further dimension reduction was performed prior to the application of the C4.5 algorithm, which created a classification tree using five selected variables. CART was also applied directly to PBS-detected peaks to identify candidate biomarkers for ovarian cancer (Rai et al., 2002) or renal transplant rejection (Clarke et al., 2003). Yet another example concerns a sample of 106 cases of prostate cancer and 56 controls (Bañez et al., 2003). Two sets of SELDI mass spectra were produced from these samples using the weak cation exchange array (WCX2) and the immobilized metal affinity capture-copper array (IMAC3-Cu). PBS detected 89 peaks for the WCX2 set and 97 for the IMAC3-Cu set. Again, there was no further variable selection. On a training set of 44 cancer cases and 30 controls, the WCX2 set produced a tree with six test nodes/variables and the IMAC3-Cu set a tree with five test nodes. Combining the data from the two arrays resulted in a simpler tree with three test nodes, which achieved higher classification performance on a blinded test set than either of the two larger trees.

However, it has been shown that prior dimensionality reduction can also enhance performance of DTs and rules. On the

EVMS prostate cancer data (PC-EVMS, Table 2), for example, the 779 peaks produced after mass spectra preprocessing were ranked according to the area under the ROC curve; 124 peaks with AUC ≥ 0.62 were retained, of which CART used only nine to build a decision tree (Adam et al., 2002). After a discrete wavelet transform on the raw mass spectra of the Duke lung cancer dataset, Zhu, Yu, and Zhang (2003) selected nine wavelet coefficients that maximized the *F*-ratio. The decision tree built from this variable set contained only two test nodes. The approach described by Alexe et al. (2004) differs from the three others on two counts. First, it creates decision rules *via* set covering rather than recursive partitioning; second, it combines a variable selection filter with a variable subset selection process wrapped around the rule induction process, using the accuracy of the induced model to score the candidate variable sets (subsection IV.C).

Controlled comparative studies give us a rough idea of how DTs fare on mass spectra with respect to other learning algorithms. On the Duke lung cancer data, DTs were found to perform worse than both linear discriminants and logistic regression in intensive 500 split-sample runs (Neville et al., 2003). Similarly, DT ranked last or next to last in a comparative study involving six different variable selection methods and four learning algorithms including SVM, Naïve Bayes, and KNN (Liu, Li, & Wong, 2002). One possible explanation for this poor performance is that DT sequential approach takes little account of variable interaction and thus fails to exploit useful information concerning the relative abundance of proteins in mass spectra. The second explanation is that the non-metric representation of DT falls short of the finer information concerning relative abundance of proteins by testing single thresholds which effect simple binary splits on peak intensities. On the other hand, DTs and rules dominate all other learning approaches in terms of model intelligibility; the biomarker patterns detected can be interpreted directly as sets of constraints on the intensity levels of the peaks identified by *m*/*z* values.

C. Ensemble Models

1. Resampling-based ensembles

Homogeneous ensembles are aggregates of base classifiers built using the same learning algorithm but different versions of the training data. The most straightforward way of obtaining diverse training subsets is by instance resampling, for example, *via* boosting or bagging. A prostate cancer dataset (PC-EVMS) has been repeatedly used as a testbed for boosting univariate base classifiers. After individual variable selection based on the AUC (Subsection IV.A), 194 peaks remained available for model building. These were used by Qu et al. (2002) to build ensembles of decision stumps following the boosting procedure described in Subsection III.C. A first ensemble with 400 base classifiers and 62 distinct peaks separated prostate cancer (PC) from non-cancer with 100% accuracy on both training and test sets; a second ensemble comprising 100 base classifiers also perfectly separated healthy (H) men from those with benign prostate hyperplasia (BPH). These two ensemble classifiers were combined to form a 3-class ensemble, which again achieved perfect accuracy on the PC *versus* BPH *versus* H problem. However, this final classifier had 500 decision stumps and 74 peaks. To build a more parsimonious and hence more interpretable classifier, the same procedure was followed with one difference: only new peaks can be selected at each iteration. The resulting 3-class classifier had 21 peaks instead of 74, but sensitivity and specificity on the test set dropped to 96.7%. This accuracy-interpretability trade-off is a recurrent observation in many of the reviewed studies.

Yasui et al. (2003) follow a technique similar to that of Qu et al. except that their base classifier is a univariate linear discriminant. At the outset, all *N* cases are assigned equal weights of 1/*N*. At each iteration *i*, each of the candidate variables is used to build a logistic regression model based on the weighted cases, and the model (variable) that maximizes the likelihood ratio is selected. The linear part of the selected logistic regression model, that is, the exponent in the sigmoid function $1/(1 + e^{-(\alpha+\beta^T s)})$, becomes the base classifier. Its predictions on the training set are evaluated, and the weights of misclassified (correctly classified) examples are increased (decreased) for the next iteration. Boosting halts when observed sensitivity and specificity exceed predefined thresholds. The resulting aggregate classifier after the last iteration *M* can be written as the sum of *M* univariate linear classifiers. While each linear classifier is univariate, selection of the sole variable needed for each model is done in interaction with other variables by virtue of the boosting. After the first base classifier is built, all previously selected variables influence each new variable selection step *via* the weight updates entailed by the performance of their respective classifiers. This method produced an aggregate classifier using 26 peaks for the PC/BPH *versus* controls problem and 25 for the cancer *versus* BPH problem.

A less common way of resampling the training data is by taking different subsets of the features instead of the instances. The best-known representative of this approach is the RF algorithm (Breiman, 2001), which has been described in Subsection III.C. RF was used by Wu et al. (2003) on ovarian cancer data, both for dimensionality reduction (Subsection IV.B) and for classification. As a variable selection mechanism, RF was compared with *t*-statistic-based variable ranking/selection; as a learning method it was compared with LDA, QDA, and KNN. For all learning algorithms used, variable selection was more effective using RF than the *t*-statistic. As a learning algorithm, RF outperformed the three other methods; it led to an overall lower error rate as well as to a more stable assessment of classification errors across different model evaluation strategies. These observations suggest that RF is one of the more promising techniques for mass spectra classification. A thorough exploration of the potential of the RandomForest algorithm for cancer diagnosis can be found in (Izmirlian, 2004).

In Li et al.'s (2003) cascaded DTs, as in RandomForest, model diversity is achieved by varying the variables instead of the instances. However, while RandomForest draws candidate variables randomly at each node, Li et al. obtain diverse trees by non-randomly selecting a different variable at the root of each tree. At the outset, all candidate variables are ranked using C4.5's default criterion. For $i = 1$ to T (the number of trees to be grown as specified by the user), the *i*th tree is initialized using the *i*th top-ranked variable on the candidate list as the root node. Tree construction is then pursued following the standard procedure, but the diversity of root nodes ensures diversity and complementarity of the different trees in the ensemble. Decisions are combined by a weighted majority vote. In tenfold cross-validation experiments on an ovarian cancer dataset (OC-H4), cascaded C4.5 and SVM achieved perfect accuracy while standard, bagged, and boosted C4.5 scored 10, 7, and 10 errors, respectively. However, SVM used all 15,154 variables whereas Li's model comprised 20 trees with 2–5 variables each. In this particular case, a relatively more intelligible model was obtained without sacrificing accuracy. However, identification and validation of the 72 variables in the entire tree ensemble remains a non-trivial task.

2. *Heterogeneous ensembles*

In contrast to the homogeneous ensembles described in the preceding section, heterogeneous ensembles combine different learning algorithms trained on the same dataset. An example is the aggregate model built by Papadopoulos et al. (2004) to diagnose sleeping sickness or trypanosomiasis. After preprocessing, the dataset of 85 diseased samples and 148 controls included 206 peak sites. Two dimensionality reduction strategies were explored. Individual variable selection based on the *t*-statistic yielded 19 potential biomarkers with a *P*-value $<10^{-5}$ while transformation of these 206 peaks into principal components produced a set of 41 derived variables which explained most of the variation in the data. On each variable set, an ensemble classifier was built combining three models built by recursive partitioning or DTs, NN training, and GAs. Classification of test instances was done by a majority vote of the base classifiers. On the 19-peak set, the ensemble classifier achieved an accuracy of 96.3% on an independent test set while the best individual base classifier DT attained 94.5%. On the 41-principal component set, the ensemble classifier achieved 99.1% accuracy *versus* 98.2% (NN), 97.2% (GA), and 94.5% (DT) for the base classifiers. These results illustrate once again the often observed trade-off between generalization performance and model understandability; better performance was obtained with 41 principal components, each of which is a linear combination of 206 detected peaks. Similarly, the multistrategy classifier outperformed all individual classifiers on either variable set, but analysis of this complex model is far from straightforward. One of the many promising research paths in biomarker discovery is the development of techniques for interpreting the knowledge distilled from data by ensemble models.

VI. EXPERIMENTATION AND MODEL EVALUATION

A classification algorithm can be evaluated along different dimensions like prediction or generalization error, understandability or novelty of the models produced, robustness, and training and classification computational requirements. For the task of classifying mass spectra we will focus on two dimensions, generalization error, and model stability (the latter can be considered as a special case of robustness). Assessment of model novelty requires the direct intervention of domain experts; in the case of mass spectra it is directly related to pattern interpretation and biomarker identification (Section VII).

Generalization Error: Recall that our definition of the classification task (Section III) assumes an instance space governed by a joint probability distribution $P(\mathbf{X},\boldsymbol{Y})$, in which data have been generated according to probability $P(\mathbf{X})$ and class labels $y=S(x)$ assigned according to a conditional probability distribution $P(Y|\mathbf{X})$. In the real world we have access to a dataset D with a limited number of examples, drawn from the distribution $P(\mathbf{X},\boldsymbol{Y})$. The classification process will be trained on this dataset in order to construct the approximation of $S(\mathbf{X})$.

The generalization error of the model, $M(\mathbf{X})$, induced by C on the dataset D, is the probability that M will misclassify an example, x, drawn at random from $P(\mathbf{X})$. That is:

$$\begin{aligned}\mathrm{Err}_M(D) &= P(M(x)\neq S(x)) = E_P[M(x)\neq S(x)]\\ &= \int_x I(M(x)\neq S(x))P(x)\mathrm{d}x\end{aligned}$$

where E_P denotes expectation with respect to P and the indicator function $I(.)$ returns 1 if its argument is true and 0 otherwise. The generalization error of the classification process C on datasets with a given number of instances N is simply the average generalization error of the classification models derived on datasets of size N; it is given by:

$$\mathrm{Err}_C(N) = E_F[\mathrm{Err}_M(D)]$$

where F is the distribution from which the datasets D of size N are drawn.

Since we cannot draw an infinite number of new examples from $P(\mathbf{X})$ in order to compute the exact generalization error of a model M, we have to rely on estimations computed by some error estimation procedure using the available data D. For this purpose we distinguish between training error (aka resubstitution error) and test error. The former is the percentage of misclassifications incurred when the model is applied to the set on which it was trained, while the latter is misclassification rate when applied to an independent set that was not used for training. Both are given by:

$$\frac{1}{N}\sum_{x_i\in D} I(M(x_i),S(x_i))$$

D is either the train or the test set, depending on which error we are computing. It is easy to construct a confidence interval for the above quantity if one considers the result of the application of the model to an example as governed by a binomial distribution.

Training error is not a good estimate of the generalization error; depending on the complexity of models that the classification process is inducing it can typically drop to zero for highly complex models. Nevertheless, a model that overfits the training data will usually have high generalization error. The test error is the best way we have to estimate the true generalization error, and it approaches the true generalization error as $N\rightarrow\infty$.

Stability: The following definition of stability was given by Turney (1995):

The stability of a classification algorithm is the degree to which it generates repeatable results, given different batches of data from the same process.

Ideally different datasets D_i from the same instance distribution $P(\mathbf{X},\boldsymbol{Y})$ should result in the same or at least very similar models. Stability is crucial when the goal of the classification task is not limited to providing accurate prediction but includes knowledge discovery, that is, pinpointing those factors that affect the classification decision, as it is definitely the case in biomarker discovery.

Imagine an application scenario in which we are asked to produce a classification model for predicting whether a given individual has developed a disease or not. As part of the analysis, we might induce a number of classification models by applying the same classification process to different subsets of the initial dataset. In an alternative scenario, two different teams working on the same problem use the same classification process but different samples; each team comes up at the end with its own classification model. The problem appears when the induced models are different, emphasizing different features of the individual's description. Which one should be trusted and given as the result of the knowledge discovery process to the experts?

The answer to this question requires the detection of the source of model instability. In the two scenarios described above there are two sources of instability: one related with the classification process that is applied, and the other with the experimental conditions under which the samples were collected and prepared. In the first scenario, the classification process we are using is sensitive to sampling variations from data collected from the same instance distribution $P(\mathbf{X},\boldsymbol{Y})$; we could lift this instability by altering the classification process by selecting more stable constituents. In the second case (assuming that the classification process is stable) different classification models might be an indication that the distributions from which we sample our training examples are in fact different due to variations of the experimental protocols used or simply because samples come really from different instance distributions.

In the following sections, we will describe how we can estimate the generalization error and the stability of a classification process.

A. Overview of Model Evaluation Methods and Metrics

1. Generalization performance

a. Principles and techniques. In what follows we will use the term classification process instead of classification algorithm. This distinction is important since the first one involves all preprocessing steps, like feature selection and parameter tunning of the classification algorithm, and the actual application of the classification algorithm to the preprocessed dataset. These should be evaluated as a single component, otherwise the error estimation would be flawed, resulting in optimistic estimates of the generalization error (more on this later in the same section).

In the estimation of generalization performance there are two notions that should be clearly identified and distinguished. The first is the generalization error of the finally induced classification model, $\text{Err}_M(D)$, and the second is the generalization error of the classification process used, $\text{Err}_C(N)$. Working with specific applications we are mostly concerned with the generalization error of the induced model since it is the one that will be applied in practice. Nevertheless the most often used error estimation procedures provide an estimation of the generalization error of the classification process and not that of a single classification model; however, under appropriate stability assumptions these are also good estimates of the $\text{Err}_M(D)$.

The general idea underlying all error estimation procedures is the division of the available set of examples into two disjoint sets. One is used for training, and the other is used for testing/evaluating the generated model. The test set should not contain examples that have been used in the training set, as this would provide optimistically biased estimates of the error. Various methods are used for obtaining the division to train/test sets and estimating the error.

Error estimators are typically characterized by two quantities, their bias and their variance. Let $\overline{\text{Err}}(C,D)$ be the error estimation produced by an error estimation method for a given dataset D with N instances and a given classification procedure C, and $\text{Err}_M(D)$ the true generalization error of the model that we wish to estimate. Then the bias of the error estimation method is simply $E_F[\text{Err}_M(D) - \overline{\text{Err}}(C,D)]$ and the variance $E_F[(\text{Err}_M(D) - \overline{\text{Err}}(C,D))^2]$, where as before the datasets D are drawn from the distribution F with respect to which the expectation is taken. Bias captures the systematic error of our error estimation method in establishing the true error, while variance measures the dispersion of our estimation. In Figure 7, we give an example of two fictional estimators, one unbiased with high variance, A, and the other biased but with low variance, B. Even though estimator B is slightly biased we might be tempted to use it because its values are less dispersed than those of A, so we can have higher confidence in its estimations.

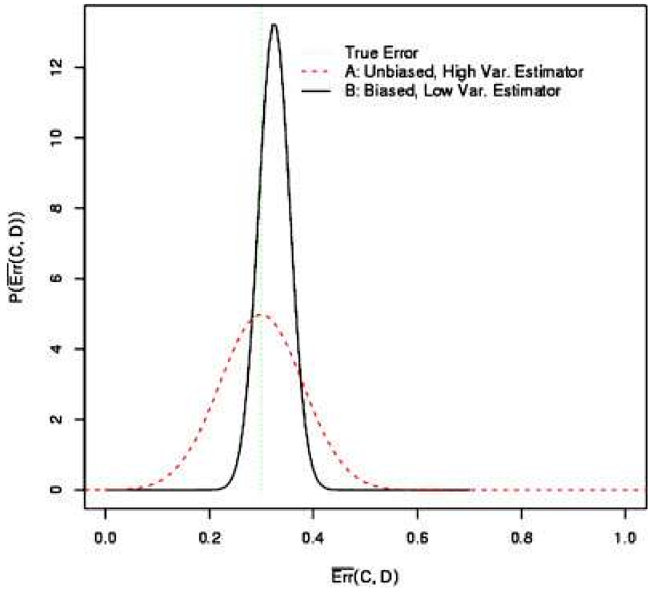

FIGURE 7. Examples of two hypothetical error estimators with different bias-variance profiles.

In an ideal scenario there are enough data available, and the error estimation procedure is relatively straightforward. In this case we set aside a number of samples that will constitute our test set, which should never be used during training. The data analyst will be working only with the training data where he can perform whatever tasks he considers necessary, like feature selection or parameter tuning, in order to derive the final classification model. One part of the training data could be used to train different models using different parameter settings of the classification algorithm or different feature sets, and another part as a validation set on which the performance of the induced models will be validated. Based on the performance on that validation set a single classification model is selected and applied to the test set. The test error is then a reliable estimation of the generalization error of our classification model, $\mathrm{Err}_M(D_{\mathrm{tr}})$, where D_{tr} is the training set (including the validation set). In fact, this is the only method that provides an estimation of the generalization error of the classification model. This approach to error estimation is referred as the holdout method. Usually 2/3 of the initial examples are used for constructing the classification model and the remaining 1/3, called the holdout set, is used for testing. Nevertheless in practice the availability of data is rather limited: using the holdout method we will test our classification model on only one third of small datasets; moreover the size of the training set will be also small. As a result the final estimation might be pessimistically biased, since better performance could probably be achieved if we used more training data, but also unreliable, due to the small amount of testing data which will result in large confidence intervals for the generalization error.

In order to get more reliable estimates of error in cases of limited data availability we have to rely on resampling techniques. Resampling is based on repeatedly separating the available data into training and test subsets, and then running the classification process on the training set and testing it on the test set. During the resampling procedure a number of classification models, possibly different, are created whose generalization error is estimated on a small part of the whole dataset. The final error estimation is an average over the different test errors coupled with a confidence interval. One of the advantages of resampling techniques is that they take into account error variations due to different training and test sets, so to some extent they can detect sensitivity to different samples coming from the same instance distribution. Resampling techniques estimate the generalization error of the classification process and not of a specific classification model. By getting an estimate of the average performance of the classification models that the classification process induces we hope that the classification model that will be finally employed will have a similar generalization error since it will be the result of the same process.

In k-fold cross-validation the available set is split into k disjoint sets. The inducer is then trained on the union of $k-1$ sets and tested on the remaining set. The whole process is repeated k times, each time a different set from the k is used as a test set. The estimation of the error is simply the average of the observed errors over the k-folds. When k equals the number of examples then the method is called leave-one-out. A variant of cross-validation is stratified cross-validation, where the partitions are constructed in such a way, that the distribution of the classes in the initial dataset is preserved. Leave-one-out provides unbiased estimates of $\mathrm{Err}_C(N-1)$ and the k-fold cross-validation unbiased estimates of $\mathrm{Err}_C(N-N/k)$.

In the bootstrap method the initial set of examples is sampled with replacement, so that a new set of the same size is established. The instances not chosen in the sampling process will form the testing set. The whole process is repeated a number of times, k, usually between 50 and 200, each time using a different sample of the examples. The estimation of the error is given by the following formula:

$$\mathrm{Err} = \frac{1}{k}\sum_{b=1}^{k}\left(0.632\varepsilon_{\mathrm{test}_b} + 0.368\varepsilon_{\mathrm{train}}\right)$$

where $\varepsilon_{\mathrm{test}_b}$ is the error of the model on the b test set, and $\varepsilon_{\mathrm{train}}$ the error of the model on the complete initial set.

Leave-one-out produces almost unbiased estimates of the true error, but with high variance. The size of the training sets is almost the same as the size of the complete dataset. The variance is reduced when we move to k-fold cross-validation, with k in the area of five to ten, and it is further reduced when we are using stratified cross-validation, though remaining relatively high. One method to reduce the variance of cross-validation is to repeat the whole procedure for a number of times. For both cross-validation and stratified cross-validation the estimates of the mean are almost unbiased. In bootstrap the error estimates are highly biased, but they have a very low variance. Bootstrap's bias is high especially when evaluating algorithms that fit the training data perfectly, for example, a one-nearest neighbor algorithm or an unpruned decision tree. In that case, $\varepsilon_{\mathrm{train}}$ is zero, leading to optimistic estimation of the error. Efron and Tibshirani (1995) propose a bootstrap version, which they call the 632+ rule, which is designed to provide less biased estimates of the error. A comparative study of cross-validation, stratified cross-validation, and bootstrap can be found in Kohavi (1995). The author concludes that the use of tenfold stratified cross-validation is appropriate for algorithm selection, even if the computational power available is sufficient for more computational intensive methods of error evaluation. In a similar study, Bailey and Elkan (1993) compared the performance of bootstrap and leave-one-out cross-validation; they also concluded that the use of cross-validation is preferable, since it exhibits much smaller bias than bootstrap. They noted though that the best choice of error estimation method depends on which algorithm is evaluated. The same observation was made by Braga-Neto and Dougherty (2003), when they examined the performance of cross-validation, bootstrap, and resubstitution estimation on small sample sizes. They found that cross-validation exhibited high variance in small sample sizes, variance that increased with the classification algorithm's ability to produce highly complex models. The variance of cross-validation decreased with an increase of the sample size, and it became comparable with that of other estimators for samples that contained more than 100 instances. The resubstitution estimation was strongly biased. Finally, bootstrap was shown to have relatively low variance, and low bias in some cases, which overall makes it an interesting evaluation strategy if one overlooks its high computational cost.

Since resampling involves the creation of multiple classification models, the problem is deciding which classification model should be finally employed. One can choose to rerun the classification process, this time on the complete dataset; in this case we cannot estimate the final model's generalization error which might possibly be less pessimistic than the initial estimation which was based on smaller training sets[3]. Alternatively one may choose one of the classification models developed as the result of the resampling process; in this case we have more confidence in the accuracy of our error estimation since it is computed on training sets of the same size. Hastie, Tibshirani, and Friedman (2001) propose what they call the one standard-error rule for selecting which model to apply: "Choose the most parsimonious model whose error is no more than one standard error above the error of the best model." The choice of the final classification model can prove irrelevant if the classification models produced during resampling are quite similar among themselves and to the classification model induced from the complete dataset, bringing up again in front the question of stability of the classification process.

Determining which error estimation method to use is a complicated task. In any case it is clear that the greater the number of training instances the more reliable the evaluation results. The following rule could serve as a rough guide: use holdout testing for large datasets and tenfold cross-validation for average-sized datasets; if the dataset has less than a couple of 100 examples then bootstrap could provide more reliable estimations due to its lower variance.

b. Methodological pitfalls. Building a classification model is a complex process which involves: preprocessing steps like feature selection, feature extraction, and/or feature combination; tuning of parameters that control the behavior of the classification algorithm, for example, the complexity of the models it induces; and only lastly, the actual training of the classification algorithm on the chosen data representation and parameters. To evaluate the predictive performance of a classification model all these steps should be done exclusively on the training data; it is only when a final classification model is built that it can be applied to the test data to estimate its performance. This rule applies to all error estimation methods; though quite clear for the holdout method, it is often violated when resampling methods are used.

A common error consists in performing feature selection on the entire dataset before evaluating the classification algorithm on the new dataset of reduced dimensionality. This entails an information leak since test examples, that is, those contained in the test folds, have been used in feature selection, which is an integral part of the model building process. This can result in overly optimistic estimations of the error as demonstrated clearly by Simon et al. (2003). Using leave-one-out cross-validation, they compared a scenario with feature selection performed only once using the complete dataset (partial cross-validation) with the correct scenario where feature selection was redone within each training fold (complete cross-validation). Experiments were conducted with artificial datasets of very high dimensionality and few instances, constructed from the same generating distribution, which contained no information. No classification algorithm could do better than random guessing on such data. Nevertheless partial cross-validation reported zero classification error in 90.2% of the 2000 artificial datasets used in the study. For complete cross-validation the median error estimation was correct, though with a relatively high variance. A similar study by Ambroise and McLachlan (2002) led to the same conclusion. When feature selection is done outside the cross-validation loop, the higher the dimensionality to sample-size ratio, the higher the odds of finding features that by pure chance discriminate the data perfectly. However such optimistic results are completely misleading; a true measure of generalization performance can be obtained only with complete cross-validation, that is, using test instances that have not been used for feature selection.

Suppose now that we have a classification algorithm, C, whose behavior is controlled by a set of parameters α. We can estimate the generalization error of C for a given value of α by using one of the error estimation techniques described above. This will yield a more or less[4] unbiased estimation of the error of C for the given α. However, if we examine a set of different values for using the same evaluation method and select the value that minimizes the error, this minimized error will not be an unbiased estimate of $\min\{\mathrm{Err}_{C_\alpha}\}$ since the selection of the appropriate value for α is done by looking *a posteriori* at the testing performance. As in feature selection parameter tuning should be redone for each training fold. This can be achieved by nesting within each training fold another resampling loop whose purpose is to estimate the performance of different parameter settings on that fold. Once a choice is made the classification algorithm should be retrained on the complete training fold using the selected parameter setting and then evaluated on the test fold following the standard procedure. The average over all the folds will now be an unbiased estimate of $\min\{\mathrm{Err}_{C_\alpha}\}$. The final classification model to be applied can be derived in two different ways. The first is to directly select the classification model produced by the most parsimonious parameter setting according to the one standard-error rule given above. The second is to simply run the classification algorithm C with each of the different parameter settings (without the nested resampling loop), select the setting that exhibits the lowest error, and then rerun the classification algorithm C on the complete training set with the selected setting. Note that in both cases we do not get the actual error of the finally produced classification model but only an error estimation as this is given by $\min\{\mathrm{Err}_{C_\alpha}\}$.

Simply comparing error estimates is not sufficient, as observed differences might not be significant in a statistical context. The estimates of the errors are sample estimates of the true error. Two inducers can have the same true error but different sample estimates. In order to establish whether the differences in the sample estimates reflect a difference in the true error or are simply the result of random fluctuations of the sample estimates around the same mean, the use of statistical significance tests is essential. A number of studies have been done on the validity of different statistical tests for detecting significant differences in classifier error (Feelders & Verkooijen, 1995; Dietterich, 1998). Among the best performing are McNemar's test and the *t*-test

[3] With the exception of the leave-one-out cross-validation where the amount of the training data used at each repetition is almost the same as the complete set of instances, nevertheless in leave-one-out the problem comes from the high variance of the estimate.

[4] Depending on the properties the error estimator.

TABLE 3. Confusion matrix for binary classification

		predicted class		
		positive	negative	(line totals)
True	positive	True Positives	False Negatives	TP + FN = P
class	negative	False Positives	True Negatives	FP + TN = N

based on the results of a twofold cross-validation repeated five times.

Depending on the type of the application we might be interested in different views of the generalization error, for example, measuring the generalization error for a given class. For these cases a number of alternative measures have been proposed. We will focus on two class problems since these are the most often met in medical diagnostic problems. We can build the confusion matrix shown in Table 3.

Then based on the above the following performance measures can be defined:

- sensitivity: TP/*P*, the percentage of positive instances correctly classified, indicates how good our classifier is in identifying the positive examples, also known as TP rate or recall.
- specificity: TP/*N*, the percentage of negative instances correctly classified, indicates how good our classifier is in identifying the negative examples.
- precision: TP/(TP+FP), the percentage of instances classified as positive that were really positive, indicates how accurate our classifier is when it predicts the positive class.

All of the above performance metrics can be estimated using exactly the same estimation methods that we described for evaluating the generalization error.

Another way of evaluating and visualizing the performance of a classification model is *via* the use of ROC graphs (Fawcett, 2003). ROC graphs are 2-dimensional graphs where the *X*-axis corresponds to the FP rate = FP/*N* (the percentage of negative instances incorrectly classified as positive) and the *Y*-axis to the TP rate. Complete ROC curves can be constructed only for classification models that output a probability or a score for their prediction. The ROC curve will then give the trade-off between TP-rate and FP-rate for every possible value of a threshold on the score. Classifiers that do not output a score become single points in a ROC graph. An example of a ROC graph with four ROC curves is given in Figure 8. Some explanations are in order. The point (0,0) corresponds to the strategy of never predicting the positive class and the point (1,0) to the strategy of always predicting the positive class. Perfect performance corresponds to the (0,1) point. The more we move to the right along a ROC curve the more we increase the TP rate, but at the same time we also increase the FP rate. Informally a point *A* is better than a point *B* if *A* appears more to the left of B (lower FP-rate) and higher (higher TP-rate). A classifier completely dominates another one if its ROC curve is always above the ROC curve of the second; in Figure 8, for example, the DT classifier is completely dominated by Naïve Bayes and the 5-nearest-neighbor. The diagonal line $y = x$ corresponds to a random classifier that predicts randomly positive with probability *p*. This random classifier has a TP rate of *p* but also a FP rate of *p*. The cost of a classifier for a given point (FP rate, TP rate) of its curve is given by:

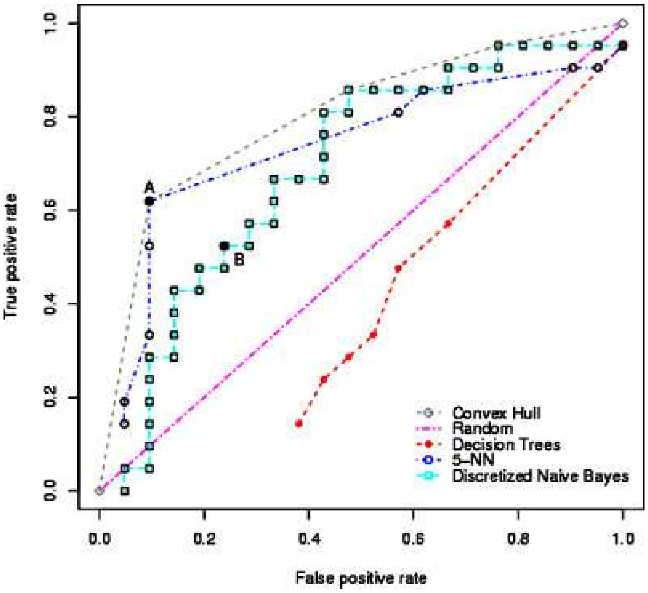

FIGURE 8. ROC curves of four different classifiers on a mass-spectrometry problem. For this specific problem, the convex hull contains discretized Naïve Bayes and 5-nearest-neighbors which can thus be optimal under certain conditions. Note that decision trees (DTs) fare worse than random, probably because their sequential variable selection mechanism assesses variables individually, in effect ignoring all interactions among peak intensities.

$$p(\text{positive}) \times (1 - \text{TP rate}) \times \text{cost}(\text{negative}, \text{positive})$$
$$+p(\text{negative}) \times (\text{FP rate}) \times \text{cost}(\text{positive, negative})$$

where *p*(positive) and *p*(negative) refer to the probability of the positive and the negative class, respectively, and cost(*x*,*y*) is the cost assigned to misclassifying *y* as *x*. One advantage of ROC curves is that they allow us to visualize the performance of the classification problem regardless of the class distribution and the misclassification costs. One can compute the convex hull of a set of classifiers (also given in Fig. 8). Classifiers on the convex hull are optimal for a range of class distributions and misclassification costs (Provost & Fawcett, 2001). Finally another measure of classifier performance is the area under the ROC curve, abbreviated as AUC. The AUC can be defined as $P(x_+ > x_-)$, that is, the probability that a classifier will rank a randomly chosen positive instance higher than a randomly chosen negative instance (Fawcett, 2003).

2. *Stability*

Stability of classification algorithms is an important issue when one is concerned with the reproducibility of results. We do not want to discover different potential biomarkers each time we analyze a different set of data. The notion of stability also provides a framework for uncovering differences in collected data that should be attributed to different experimental conditions.

To measure the stability of a classification process one needs to define a measure of similarity among classification models.

One way to do this is to use a syntactic measure of similarity. The problem with this approach is that it depends on the representation language used in the classification process. Moreover models, which seem very different based on a syntactic measure of similarity might in fact be logically equivalent, delivering exactly the same predictions over all possible inputs.

Turney (1995) proposed a measure of stability based on the agreement of two classification models. He defined the agreement of two classification models as the probability that they will produce the same predictions over all possible instances drawn from $P(\mathbf{X})$. Note here that instances are drawn from $P(\mathbf{X})$ and not $P(\mathbf{X},Y)$; the underlying reason is that the agreement of two concepts should be examined in all possible input worlds. In order then to measure the stability of a classification process he gave a simple algorithm based on $m \times$ twofold cross-validation. In twofold cross-validation the available data are split in two, each part used once for training and once for testing. However, testing is not done on a test set but on an artificial set constructed by sampling uniformly over all possible values of X. The classification process is run on each of the subsets, and the classification models produced are applied to the artificial instances. Stability is simply the percentage of times that the two models agree, independently of the instances' actual classes. The final result is the average over all the m runs (each run results in a different random split of the initial dataset). This approach provides an empirical estimation of the logical agreement of two concepts.

Another measure for the stability of a classification process comes from the bias-variance error decomposition (Domingos, 2000). For a given test instance classification error is decomposed into three components:

$$\mathrm{Err}(x) = c_1 N(x) + B(x) + c_2 V(x)$$

A first-irreducible component is due to the inherent noise, $N(x)$. The second, $B(x)$, is the bias of the classification process and measures the systematic error of the classification process. It measures the distance from the optimal prediction of the "average" prediction of the classification models constructed from different training sets. The optimal prediction is the class of the given instance in case there is no noise. In case there is noise it is the most common class label with which the instance is seen. Finally the variance term, $V(x)$, measures the variance of the predictions of the different classification models around the "average" prediction. c_is are multiplicative factors. The general procedure for the computation of the bias-variance decomposition is also based on resampling. A number of different training sets should be constructed and the classification process run on each of them in order to generate a classification model. Then each of these models will be applied to the same fixed test set, and the above quantities will be computed. What the variance term depicts is the variation in predictions, due to the differences in the training set, around the most common prediction. A completely stable classification process would have zero variance, that is, its predictions for a given instance would not change with changes in the training set.

Both of the above approaches couple error evaluation with an estimate of the stability of the resulting classification models. They could be used to select among different classification processes based on a combination of error and stability, that is, select a classification process which yields both low error and high stability. Stability is an issue that is relatively ignored in classification performance studies. Nevertheless it is important due to the way error evaluation is done, that is, *via* multiple resamples from the same dataset that usually give rise to different classification models. If the resulting models are relatively stable then first, we can have more confidence in the result of the error estimation procedure, and second, the problem of selecting the final model for deployment in the real world becomes less critical since all of them will be more or less logically equivalent.

The concept of stability can also be used to detect differences in data collection, experimental protocols, equipment etc. For example, assume that two teams work remotely on the same mass-spectrometry application. One team produces a classification model with low predictive error and good stability, and sends it to the other team for testing. Nevertheless the testing results show a high predictive error. This is simply an indication of a change in the joint distribution $P(\mathbf{X},Y)$ that can be attributed to any of the factors mentioned above. Alternatively one could use the framework proposed by Turney (1995) to assess the degree of agreement of the classification models when tested on a collection of artificial instances drawn from $P(\mathbf{X})$. Again high instability would mean that there is a difference in the generation of the training data in the two laboratories.

B. A Critical Perspective on Evaluation Practices for Biomarker Discovery

We will undertake a short review of biomarker discovery on mass spectrometry based on machine learning and data mining techniques with respect to the two aforementioned dimensions, that is, error evaluation and stability-reproducibility of results, and try to pinpoint the most common methodological flaws and how these should have been tackled.

1. Generalization performance

One of the most common methodological flaws in mass-spectra classification concerns the selection of the most discriminating features or *m*/*z* ratios. There are many cases where feature selection is kept outside the evaluation loop of the classification process. To highlight the extent of confusion, even within the same paper this can be done correctly for one dataset and wrongly for another one. For example, Liu, Li, and Wong (2002) examine feature selection methods on two different datasets, a microarray and a mass-spectra dataset. On the first set they rely on a holdout evaluation procedure where feature selection and learning are done only on the training data, and the learned model is correctly tested on a blind test set. However, on the second dataset (the mass-spectra dataset) they use cross-validation, but instead of tightly coupling the learning algorithm with feature selection within each cross-validation fold, they apply feature selection to the complete dataset prior to cross-validation. The cross-validated error is estimated only for the classification algorithm using features selected on the complete dataset, thus leading to optimistic estimations of error. Wu et al. (2003) and Tibshirani et al. (2004) worked on the same mass-spectra problem, used the same feature selection strategy, the same error evaluation procedure, and examined among others two classification algorithms that were common to both studies, support vector machines, and linear discriminants. The first study reports errors that are in the range of 12%–14% for the two learning algorithms. However the errors that the second study reports are more than double, 30%–35%. The difference is that in the first study feature selection was done outside the cross-validation loop once using the complete dataset while on the second it was correctly redone within each fold of the cross-validation. This is a very clear demonstration of how a flawed error estimation procedure can provide optimistically biased results.

Feature selection is not the only stage of learning that can create problems during the error estimation procedure. The same type of optimistic error estimation can appear also when tuning the parameters of a given learning algorithm. Alexe et al. (2004) examine the performance of logical analysis on mass-spectrometry data. They report sensitivities and specificities up to 100%. Their system requires setting up a number of input parameters that affect the learning behavior. In order to select the best parameter setting they performed a systematic search on the parameter space using information from the complete dataset. More precisely during the search each parameter setting was evaluated by k-fold ($k = 2,5,10$) cross-validation on the complete dataset. The search was continued until the results were deemed satisfactory, that is, until a parameter setting was found with low predictive error. As explained above, this procedure overfits the model to the given dataset and does not provide an unbiased error estimate. Sound error estimation can be achieved in two ways. A holdout test set can be kept aside on which the final model resulting from the selected parameter setting will be tested; alternatively, cross-validation can be used, but during each training phase of the cross-validation loop, a cross-validated parameter search can be done using the current training fold. An example of sound error estimation including systematic parameter setting is given in Tibshirani et al. (2004). There cross-validation is used to estimate the error of a number of classification algorithms; the classification process includes feature selection and extensive parameter tuning of some of the algorithms. Both feature selection and parameter tuning are redone for every training fold of the cross-validation.

Work with machine learning and data mining techniques is relatively new in the area of mass spectrometry. Nevertheless the fields of both machine learning and data mining are relatively mature fields with well-established strategies for performance evaluation. For reported results on mass-spectra mining to be meaningful and allow for valid comparisons, a strict methodological framework should be followed. More stringent review policies concerning data mining methodology might help prevent the proliferation of results of flawed data mining experiments.

2. *Stability*

Stability and reproducibility of results is an issue that has been largely neglected in the analysis of mass-spectrometry data. The only exceptions to our knowledge are the articles of Rogers et al. (2003) and Papadopoulos et al. (2004). Both of these examine whether a classification model produced at one point in time from a given dataset is still valid with respect to its predictions when applied to another dataset collected at a different moment and possibly even under different experimental conditions. Neither of the two author teams examined the effect that different data samples could have in the construction of the classification model.

Papadopoulos et al. (2004) did a small-scale study of reproducibility. They used an ensemble of classification models consisting of NNs, DTs, and GAs. They tested the predictive performance of their ensemble model on the same sample whose mass-spectra was rerun 28 different times over a period of 2 weeks. Unfortunately no conclusion can be drawn since the testing data consisted only of a single control sample (it was always correctly classified). They also examined the effects of hemolysis using eight samples (three controls and five patients) all correctly classified. Finally they examined the effect of sample degradation on 18 samples (ten patients and eight controls), which were reprocessed after having been thawed. In the latter case all of them were systematically classified as controls. Rogers et al. (2003) created a classification model using a NN with five hidden layers. Classification performance on a blind holdout set was found to be in the range of 81.8%–83.3% in terms of sensitivity and specificity. Nevertheless when the same model was tested on an independent dataset collected approximately 10 months later its performance was significantly lower with sensitivities and specificities in the range of 41.0%–76.6%. This performance discrepancy can be due to two factors: one is a poor error estimation strategy and the second is a difference between the distribution on which the classification model was trained and the distribution on it was tested. Performance estimates on the first-data sample were taken, as already mentioned, using holdout. One problem of holdout is that it does not measure variations of performance due to differences in training and test set, especially when, as here, the number of available instances is relatively small (218 instances including both stages of the study). It could be simply that the specific train-test split used initially was favorable by pure chance. If a resampling evaluation method had been used it might have revealed a lower performance estimate. Learning distribution differences could be attributed to differences in the experimental protocol used to collect the second set of data; different chips were used to generate the mass-spectra. Declining performance of the laser and the detector with time could have also played a role. Because of the number of varying factors no safe conclusion can be drawn on the source of instability.

Altogether both studies clearly demonstrate the need for a careful examination of the stability and reproducibility of the results. Any such study should first exclude the eventuality that non-reproducibility is due to the unstable nature of the algorithms used. This can be easily done by performing a stability analysis of the classification algorithm along the lines described in ‘‘Stability’’ subsection under ‘‘Overview of Model Evaluation Methods and Metrics’’ subsection. Only then can one examine reproducibility within the same experimental protocol. An issue that should be closely examined is how predictive performance is affected by the time scale but also by the site where data collection took place. A number of questions arise here: would rerunning the same testing samples over different time points and then feeding them to the classification model still produce the same predictions? Would rerunning the same training data and reconstructing the classification model still produce the same classification models? More importantly, is the predictive performance stable over time, that is, in a different batch of testing data collected at a later point in time, in the same but also in different laboratories, do we get a similar predictive performance with the already constructed classification model? Equally important, does rerunning our classification process with a new batch of training data, again taken at a later point in time from the same or different laboratories, produce the same or at least a very similar classification model? All these questions, especially the last two, should be studied systematically and answered in an affirmative manner for the results of biomarker discovery on mass-spectrometry application to be trustworthy and reliable. The goal of the Early Detection Research Network validation study (Grizzle et al., 2004) was precisely to determine the portability and reproducibility of mass-spectrometry and more specifically of the SELDI technology in the context of prostate cancer prognosis and diagnosis. Among the different issues that were examined in that initiative are: the reproducibility of mass-spectra of the same samples when mass-spectrometry is performed in different sites and whether the predictive power of classification models acquired within a specific site remains valid for mass-spectrometry data of different samples collected in the different sites.

C. Evaluation Results: A Comparative Study

Comparing performance results among different studies is not a straightforward task. First of all, comparison should take place on exactly the same dataset. Second, there should be no methodological flaws that would invalidate the evaluation results. Third, the same error evaluation strategy should be used for results to be comparable; ideally even the separation into training and test folds should be the same, however this is more easily done within the same study than across different studies. In the comparisons given below we will try to keep fixed as many factors of the experimental evaluation as possible. We will state clearly when and for which reasons a comparison is not possible. Finally, the results reported below from the studies of Liu, Li, and Wong (2002) and Wu et al. (2003) should be interpreted as approximations given the flawed evaluation practices described in ‘‘Generalization Performance’’ subsection.

1. Ovarian cancer

For ovarian cancer, three different datasets have been made available by the FDA Clinical Proteomics Databank. Two studies have been run on the first version, OC-H4 (Table 2). Petricoin et al. (2002) heuristic machine learning approach led to the extraction of a 5-marker, which achieved 100% sensitivity and 95% specificity. These performance measures were observed on a blind test set of 116 (50 diseased, 66 benign/control) out of 216 samples, that is, with a 46%/54% train/test split. Lilien, Farid, and Donald (2003) ran Q5 on the same dataset under a variety of experimental conditions. To ensure fair comparison, we selected the experimentation settings closest to those used by Petricoin: 50%/50% train/test split and a probability classification threshold of 0.5 which led to the classification of 98.04% of the test set (Petricoin et al.'s method classified all test samples). Under this setup, Q5's closed-form, exact statistical approach obtained a sensitivity of 87.57% and a specificity of 90.15%. The authors report a classification threshold that classifies 90% of the OC-H4 samples with a sensitivity of 97.5% and a specificity of 96.8% but the 10% unspecified training/test ratio for these results preclude any meaningful comparison. In contrast to the precise subset of *m*/*z* values harvested by Petricoin et al., the final pattern extracted by Lilien et al.'s linear discriminant was a linear combination of the principal components that had been used as variables for the learning phase. The discriminant was back-projected onto mass-spectral space and reexpressed in terms of the original *m*/*z* values. Those with the highest coefficients were then selected for further investigation.

Four different teams experimented with ovarian cancer dataset OC-WCX2b (Table 2). All four report 100% sensitivity and 100% specificity as their best results. With a 50% probability classification threshold and a training proportion of at least 75%, Q5 classified all test samples with perfect accuracy (Lilien, Farid, & Donald, 2000). Sorace and Zhan (2003) used Wilcoxon variable ranking followed by stepwise discriminant analysis to train three linear models on 49% if the dataset, then tested these on the remaining data. Two models with different sets of seven *m*/*z* values each achieved perfect classification, while a third model with 13 *m*/*z* values scored 96.25% sensitivity and 91.11% specificity. Jong, Marchiori, and van derVaart (2004) studied the performance of linear SVMs on the full feature set as well with feature selection using RFE, Join, and Ensemble (‘‘Variable Subset Selection’’ subsection). They also report best results of 100% sensitivity and specificity; however their final feature set had 187 features, significantly more than the seven features reported by Sorace and Zhan (2003). Interestingly enough SVMs with no feature selection yielded a sensitivity of 100% and a specificity of 99.55%. The evaluation strategy that they used was ten times holdout testing with around 25% of the total dataset kept for the holdout test set. Liu, Li, and Wong (2002) used tenfold cross-validation to compare different combinations of variable selection and learning methods. Among the variable selection methods, the subset selection algorithm CFS consistently achieved the best performance for each of the four learning algorithms used. Among the learning algorithms, SVM scored the lowest average error over the different variable selection methods used. Two configurations achieved perfect accuracy: CFS-SVM and CFS-KNN. Another remarkable result is that SVM achieved perfect classification even without variable selection (using all 15,154 *m*/*z* values).

Wu et al. (2003)'s comparative study of methods for ovarian cancer diagnosis was based on MALDI-TOF spectra (OC-NWUH, Table 2). Two feature selection algorithms—*t*-statistic-based variable ranking and RandomForest variable scoring—were explored in conjunction with five learning methods: linear and quadratic discriminant analysis, KNN, SVM, boosted CART, and RandomForest. Classification accuracy was estimated using both tenfold cross-validation and bootstrap for the first five algorithms and a 2:1 train/test split for the combined models. Only variable subsets of size 15 and 25 were considered. On both sizes, SVM achieved best performance when variables were ranked according to the *t*-statistic; however, with RandomForest-based variable selection, RandomForest and boosted CART did better than SVM. Overall, RandomForest-based variable selection not only led to higher accuracy, it also proved to be more stable than *t*-statistic-based variable ranking. Tibshirani et al. (2004) worked on the same dataset and compared peak probability contrasts (PPC, "Linear and Quadratic Discriminant Analysis" subsection) to a number of classification algorithms coupled with a *t*-statistic-based feature selection. Error evaluation was done using tenfold cross-validation with feature selection and parameter tuning always correctly redone within every cross-validation fold. Quite surprisingly, best performance (23.6% error) was achieved by an SVM model which used the complete feature set (91360 features), followed closely by PPC which used only seven features with an error of 25.8% SVM and LDA coupled with a *t*-statistic-based feature selection were found to have errors between 30.3% and 34.8%, more than double the errors estimated by Wu et al. (2003); as explained in "Generalization Performance" subsection, the discrepancy was due to the poor evaluation methodology followed in the latter study.

2. *Prostate cancer*

We analyzed five studies based on the PC-EVMS data (Table 2). Two studies (Lilien, Farid, & Donald, 2000; Qu et al., 2003) used different subsets of the data and are not comparable. The remaining three (Adam et al., 2002; Qu et al., 2002; Yasui et al., 2003) followed the same 85%–15% decomposition of the 386-specimen dataset into a training set and an independent test set. Two employed an AUC-based variable ranking method to reduce the set of candidate markers to 779 *m*/*z* values. Adam et al. (2002) used CART to produce a 9-node decision tree. Qu et al. (2002) used boosting to create two committees of decision stumps (1-node DTs), which classified cases *via* a weighted majority vote. For the first committee Adaboost generated an aggregate classifier comprising 500 base classifiers and 74 peaks. To reduce model complexity, a variant called boosted decision stump feature selection (BDSFS) required that each variable be used exactly once or not at all; the result was an aggregate classifier with 21 base classifiers and 21 peaks. Yasui et al. (2003)'s approach, described in Section V.C.1, is likewise based on boosting but combines marker selection with linear discriminant analysis within the boosting cycle. Unlike DTs, which can handle any number of classes directly, linear classifiers are basically binary. Thus, at least two linear classifiers were needed for these 3-class problems. One classifier was trained to distinguish PC/BPH *versus* control, a second to separate PC from BPH. The final classifier combined 2 linear classifiers comprising 26 and 25 peaks, respectively. We follow Yasui et al.'s (2003) decomposition to compare the three solutions to the prostate cancer diagnostic problem in Table 4.

TABLE 4. Performance measures on the prostate cancer problem (%)

Trained Model	PC/BPH vs C		PC vs BPH	
	Sensit.	Specif.	Sensit.	Specif.
AUC+CART	91	100	83	93
AUC+Adaboost	100	100	100	100
AUC+BDSFS	100	93	97	100
Boosted Linear Discriminant	98	100	93	47

PC, prostate cancer; BPH, benign hyperplasia; *C*, controls.

3. *Lung cancer*

The lung cancer data set was the object of a data mining challenge (Campa, Fitzgerald, & Patz, 2003) that elicited more than a dozen experimental studies. Three solutions were selected for this review with the aim of illustrating the diversity of approaches explored. Lee et al. (2003) built several partial least squares discriminant (PLS-DA) models, each with a different experimental strategy. A first model built on the complete data achieved 100% accuracy—a result both unsurprising and unreliable, as the model was trained and tested on the same data. The dataset was then partitioned into a design set of 28 cases and a test set of 13. A PLS discriminant was built from the design set by sevenfold cross-validation. The resulting two-component model produced one false positive and one false negative, yielding a sensitivity of 87.5% and specificity of 80%, or an overall accuracy of 85%. The same process using leave-one-out cross-validation led to a final two-component model with an accuracy of 76%. The differences in these three accuracy rates illustrate the impact of the error estimation strategy on model assessment and selection.

Wagner, Naik, and Pothen (2003)) used *F*-ratio based variable ranking followed by a comparative study of five learning algorithms—linear and quadratic discriminant analysis, kernel-based density estimation, KNN, and SVM. They tested two different experimental protocols to select between 3 and 15 peaks: the first used the full dataset to rank variables before cross-validation whereas the second-integrated variable selection into the leave-one-out cross-validation loop. We ignore the results of the first strategy, which has the methodological flaw of using test sample labels in variable selection. The second strategy produced best results with 13-peak models. Linear SVM outperformed all other classifiers with an accuracy of 98% (96% sensitivity, 100% specificity) as opposed to 73% for the closest runner-up.

Baggerly et al. (2003) built biomarker patterns of 1–5 peaks using a GA/linear discriminant hybrid. The accuracy of these peak sets was then estimated *via* leave-one-out cross-validation. The best single peak, which appeared in all the best 1- to 5-peak sets, scored 74%; accuracy increased with peak size, the best 5-peak set attaining 98%. Again, these results should be taken with caution; since peak selection involving a supervised learning technique (LDA) was done on the full data before cross-validation, the accuracy rates reported are likely to be optimistic.

D. Discussion

Meaningful comparisons could be done on very few of the studies presented above. For the ovarian cancer dataset OC-H4, the studies of Petricoin et al. (2002) and Lilien, Farid, and Donald (2000) could be compared after finding the specific experimental setting in the second study that was most similar to that of the first. For dataset OC-WCX2b each of the four studies examined used a different way to partition the available data into train-test sets so no fair comparison was possible. Nevertheless an interesting observation reported by both Jong et al. (2004) and Liu, Li, and Wong (2002) was the excellent performance of linear SVMs on the complete feature set. While this is good enough when we want to perform only classification, it does not help us when the goal is biomarker discovery. For the MALDI-TOF ovarian cancer data one of the two studies had methodological flaws so again comparison could not take place. However in Tibshirani et al. (2004) a number of methods were compared under the same framework, which allowed for some meaningful comparisons. Here too linear SVMs on the full feature set achieved top performance, though again sidestepping the problem of biomarker discovery. Very good performance with a small feature set was achieved by the PPC method introduced by the authors. For the prostate cancer dataset PC-EVMS it was possible to compare the results of three studies (Adam et al., 2002; Qu et al., 2002; Yasui et al., 2003) while two others (Lilien, Farid, & Donald, 2000; Qu et al., 2003) were excluded because they used different subsets of the initial data. The clear winner was a combination of Adaboost with AUC-based individual variable selection. For the lung cancer challenge there could not be any comparison since the different studies either had methodological problems or used different evaluation methods.

As discussed in "Which Classification Algorithm?" subsection, there is no algorithm that works best for all types of problems. If domain knowledge is available about the type of feature interactions that are sought or are expected to be found, this should motivate the choice of learning approaches that match the requirements of the problem. Otherwise the only avenue is systematic experimentation and evaluation of different learning paradigms. In fact the relative superiority of a given learning paradigm for a specific problem is informative of the form of the concept that underlies the classes, that is, the target concept probably requires the type of decision boundaries that the learning algorithm is able to discover.

VII. MODEL INTERPRETATION AND BIOMARKER IDENTIFICATION

After the final classifier has been validated from a data mining perspective, in particular with respect to predictive accuracy and model stability, the reins are handed to the biomedical researcher whose task is to validate and interpret the biological implications of the computer model. A detailed discussion of this essential phase is beyond the scope of this review; descriptions of the approaches used and problems encountered by biologists in identifying, validating, and interpreting mass spectra-based biomarker patterns can be found in Watkins et al. (2001), Adam et al. (2003), Allard et al. (2004), and Koopmann et al. (2004). The twofold purpose of this short section is to highlight the need for a human-readable diagnostic model and to sum-up ongoing debate in the biomedical community concerning the methods, assumptions, and validity of current biomarker research.

In applications such as handwriting recognition, data-driven pattern recognition models can be black boxes provided that predictions remain reasonably accurate. In contrast, model intelligibility is an indispensable requirement for medical applications in general. Producing a readable model is more or less difficult depending on the induction algorithm used and the number of variables in the final model. Classification models can be situated along a spectrum depending on the human effort required to interpret them. At one endpoint, symbolic models such as DTs and rules are straightforward to interpret. A bit further down the scale, linear discriminant classifiers remain relatively easy to understand since the relative importance of each variable is reflected by the magnitude of its coefficient (recall that the variables have been normalized (Section II) and reduced (Section IV) to the p' most discriminatory variables, obligatorily with $p' < n$ for linear discriminants). Neural networks and support vector classifiers with non-linear kernels can be grouped at the high-opacity extreme; despite intensive research on NN interpretation in the 1990s (see Andrews, Diederich, & Tickle, 1995 for an overview), translating them into readable form remains a non-trivial task.

In general, model complexity impedes understandability. With the advent of systems biology, however, extremely elaborate models have been produced to explain biological systems and processes, which are naturally and overwhelmingly complex. Such models are not completely devoid of utility, as they can provide functional definitions of systems properties, which can be tested against observed facts. Given two models of equivalent explanatory power, however, the more parsimonious one should be preferred—and parsimony, in mass spectra classification, concerns above all the size of the variable set. Eminently readable models such as DTs and rules can quickly become incomprehensible as the number of variables increases; this provides yet another justification for aggressive feature selection in biomarker discovery.

Unfortunately the model interpretability issue has been relatively neglected in computer scientists' work on mass spectra classification. Many of the studies reviewed above have focused on reporting generalization performance without providing the minimal information required to make classifiers useful to biomedical researchers—the list of discriminatory *m/z* values. Though it is beyond data miners' competence to investigate the identities and roles of the selected *m/z* values or peaks, a clear idea of the direction and magnitude of the impact of certain peaks would ease considerably the burden of interpretation that awaits the domain experts. Given such leads, biomedical researchers can review related work and single out the masses or peaks on which there is an emerging consensus. They can then undertake, on this highly reduced candidate set, the laborious process of identifying the associated proteins and discovering how these are related to the disease process.

The issue of interpreting biomarker patterns mined from mass spectra has been the subject of recent debate within the biomedical community. The defenders of proteomic pattern diagnostics claim that biomarker or proteomic patterns mined from mass spectra could be used directly as biomarkers with no need to identify the component proteins (Wulfkuhle, Liotta, & Petricoin, 2003). Adversaries of this school of thought contend that without knowing the identity of the individual proteins, it is unlikely that the method will be useful for cancer diagnosis. Diamandis (2003, 2004) observes that the patterns found in five different prostate cancer studies were completely different, even in those conducted by the same team. On the other hand, the best prostate cancer marker to current knowledge, PSA, did not appear in any of the published patterns. He raised the hypothesis that MS technology may have trouble detecting validated cancer markers which are low-abundance proteins and was instead picking up molecules present in serum at much higher levels of concentration. In addition, he surmised that the discriminatory molecules found did not originate from prostate, but were actually epiphenomena of cancer, that is, they were produced by other organs in response either to the presence of cancer or to the patient's general condition.

The same concern over reproducibility was aired by other researchers who examined datasets on ovarian cancer published by Petricoin after their Lancet publication in 2002. Sorace and Zhan (2003) and Baggerly, Morris, and Coombes (2004) noted that most discriminatory peaks belonged to the low-molecular weight region; several confirmatory experiments revealed that the patterns found in this region indeed displayed a discriminatory ability that was well beyond that expected of random noise. Both teams concluded, contrary to Diamandis, that this structure in noise had nothing to do with the underlying biology, but was due to artefacts of flawed experimentation (e.g., mid-experiment protocol shift, suspect mass calibration (Baggerly, Morris, & Coombes, 2004)). In reply to critics, Petricoin and his colleagues remarked that the initial paper was a proof of feasibility and that the methodology has undergone significant refinement since then (Check, 2004). They however contest the claim that low molecular weight values always represent noise. Using MS to identify the entire low-molecular weight region of the proteome, they have found that the region contains thousands of whole proteins and fragments including oncogenes. The ultimate goal of this identification effort is to be able to investigate extracted proteomic patterns by searching directly the corresponding identities in a database (Petricoin & Liotta, 2003).

There is in fact greater agreement on the basic issues than is apparent at first sight. Petricoin and Liotta agree that knowing the identities of the distinguishing proteins can lead to insights concerning their relationship to the underlying pathology; however, it is not an absolute precondition for the clinical evaluation of proteomic patterns: for example, CA-125 was used for cancer testing for many years before it was sequenced and characterized. On the other hand, Diamandis (2003) agrees that knowing the identities of the discriminatory molecules "is not absolutely necessary for their use as biomarkers, but without this knowledge, the method will remain empirical and probably difficult to validate, reproduce, standardize, and quality control." Wulfkuhle, Liotta, and Petricoin (2003) establish the same requirements before proteomic pattern diagnostics can be incorporated into routine clinical practice: "Standard operating procedures must be established for sample handling and processing. Reproducibility standards for proteomic patterns and a universal reference standard for quality control of MS instruments must also be developed. Equivalent reproducibility and quality control/quality assurance release specifications, spectral quality measures, machine-to-machine, lab-to-lab and process-driven variability measures must be identified and controlled for." In summary, the seminal paper of Petricoin et al. (2002) and the publication on the web of the related datasets have aroused both interest in and founded criticism of the methodology used as well as the underlying assumptions of proteomic pattern diagnostics. Beyond often intensive debate which is part of the growing pains of MS-based clinical proteomics, what appears today is a basic concurrence of views on the priority tasks for the coming years.

VIII. CONCLUSION

Despite intensive ongoing research on preprocessing and classification of protein mass spectra for biomarker discovery, the field is still very much in its infancy. Data analysts are only starting to unravel the computational difficulties involved in building accurate predictive models from extremely noisy, high dimensional, and often very small samples. Digital signal processing and statistical techniques need to be combined in order to assess the quality of raw mass spectra and transform these into a representation appropriate for knowledge discovery. As for the classification task itself, the major challenge remains the high-dimensionality small-sample or $p \gg n$ problem common to mass spectra and microarray classification. Much of existing work has focused on applying off-the-shelf classification algorithms and reporting predictive performance. However, there has been a recent trend to devise approaches tailored to the specific idiosyncrasies of mass spectral data, either by innovative combinations of known methods (Baggerly et al., 2003) or by the introduction of novel algorithms (Tibshirani et al., 2004). Differences in experimental conditions and even blatantly flawed evaluation strategies preclude a comprehensive assessment of the relative merits of the methods used, whether old or new. However, several comparative studies on specific datasets have led to independent and corroborating observations of the resilience of SVM to the $p \gg n$ problem, even where p is on the order of several thousands. This is, however, no panacea in the case of mass-spectra classification for biomarker discovery, for the opacity of the resulting SVM classifiers render them inexploitable for subsequent biological validation. Interpretability is a condition of verifiability by domain experts, and model parsimony in terms of the number of variables used is a condition of interpretability.

It therefore seems that aggressive dimensionality reduction is an indispensable requisite for biomarker discovery. It is essential to any approach that would provide a solution to the $p \gg n$ problem while satisfying stringent requirements on model interpretability. Added to these constraints is the practical impossibility, for the biomedical researcher, of experimentally identifying and validating the impact of several hundreds/thousands of candidate markers. In this context, the most promising approaches to date include SVMs coupled with filter/wrapper variable (subset) selection as well as methods which embed variable selection into the learning process to produce either a single model, e.g., shrunken centroids (Tibshirani et al., 2002) or an ensemble of base level models, e.g., RandomForest (Breiman, 2001; Izmirlian, 2004). While the predictive performance of each of these approaches depends on the characteristics of the dataset and the concept underlying the class structure, there is general agreement on relative interpretability. Linear models such as shrunken centroids and linear SVMs give a clear indication of the importance of each discriminatory peak; this is not the case for ensemble classifiers and SVM models based on non-linear kernels. A priority research issue is finding ways to decipher these models and translate them into human-readable form if their recognized predictive power is to be put to full use in biomarker discovery.

REFERENCES

Aach J, Church GM. 2001. Aligning gene expression time series with time warping algorithms. Bioinformatics 17:495–508.

Adam BL, Qu Y, Davis JW, Ward M, Clements MA, Cazares L, Semmes O, Schellhammer PF, Yasui Y, Feng Z, Wright GL. 2002. Serum protein fingerprinting coupled with a pattern-matching algorithm distinguishes prostate cancer from benign prostate hyperplasia and healthy men. Cancer Res 62:3609–3614.

Adam PJ, Boyd R, Tyson KL, Fletcher GC, Stamps A, Hudson L, Poyser HR, Redpath N. 2003. Comprehensive proteomic analysis of breast cancer cell membranes reveals unique proteins with potential roles in clinical cancer. J Biol Chem 278:6482–6489.

Aebersold R, Mann M. 2003. Mass spectrometry-based proteomics. Nature 422:198–207.

Alexe G, Alexe S, Liotta LA, Petricoin E, Reiss M, Hammer PL. 2004. Ovarian cancer detection by logical analysis of proteomic data. Proteomics 4:766–783.

Allard L, Lescuyer P, Burgess J, Leung K, Ward M, Walter W, Burkhard P, Corthals G, Hochstrasser D, Sanchez JC. 2004. ApoCI and CIII as potential plasmatic markers to decipher ischemic and hemorrhagic stroke. Proteomics 4:2242–2251.

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

Anderle M, Roy S, Lin H, Becker C, Joho K. 2004. Quantifying reproducibility for differential proteomics: Noise analysis for protein liquid chromatography-mass spectrometry of human serum. Bioinformatics 20:3575–3582.

Andreev VP, Rejtar T, Chen HS, Moskovets EV, Ivanov AR, Karger BL. 2003. A universal denoising and peak picking algorithm for LC-MS based on matched filtration in the chromatographic time domain. Anal Chem 75:6314–6326.

Andrews R, Diederich J, Tickle A. 1995. A survey and critique of techniques for extracting rules from neural networks (Technical Report). Neurocomputing Research Centre Queensland.

Baggerly KA, Morris JS, Coombes KR. 2004. Reproducibility of SELDI-TOF protein patterns in serum: Comparing datasets from different experiments. Bioinformatics 20:777–785.

Baggerly KA, Morris JS, Wang J, Gold D, Xiao LC, Coombes KR. 2003. A comprehensive approach to the analysis of matrix-assisted laser desorption/ionization time of flight proteomics spectra from serum samples. Proteomics 3:1667–1672.

Bailey T, Elkan C. 1993. Estimating the accuracy of learned concepts. Proceedings of the 13th International Joint Conference on Artificial Intelligence (p 895–900). Morgan Kaufman.

Barak P. 1995. Smoothing and differentiation by an adaptive-degree polynomial filter. Anal Chem 67:2758–2762.

Bañez LL, Prasanna P, Sun L, Ali A, Zou Z, Adam B, McLeod DG, Moul JW, Srivastava S. 2003. Diagnostic potential of serum proteomic patterns in prostate cancer. J Urol 170:442–446.

Beer I, Barnea E, Ziv T, Admon A. 2004. Improving large-scale proteomics by clustering of mass spectrometry data. Proteomics 4:950–960.

Bern M, Goldberg D, McDonald WH, Yates JR III. 2004. Automatic quality assessment of peptide tandem mass spectra. Bioinformatics 20:i49–i54.

Berndt P, Hobohm U, Langen H. 1999. Reliable automatic protein identification from matrix-assisted laser desorption/ionisation mass spectrometry peptide fingerprints. Electrophoresis 20:2521–2526.

Bienvenut WF, Sanchez JC, Karmime A, Rouge V, Rose K, Binz PA, Hochstrasser DF. 1999. Toward a clinical molecular scanner for proteome research: Parallel protein chemical processing before and during westernblot. Anal Chem 71:4800–4807.

Binz PA, Muller M, Walther D, Bienvenut WV, Gras R, Hoogland C, Bouchet G, Gasteiger E, Fabbretti R, Gay S, Palagi P, Wilkins MR, Rouge V, Tonella L, Paesano S, Rossellat G, Karmime A, Bairoch A, Sanchez JC, Appel RD, Hochstrasser DF. 1999. A molecular scanner to highly automate proteomic research and to display proteome images. Anal Chem 71:4981–4988.

Bishop CM. 1995. Neural networks for pattern recognition. Oxford: Oxford University Press.

Blueggel M, Chamrad D, Meyer HE. 2004. Bioinformatics in proteomics. Curr Pharm Biotech 5:79–88.

Boros E, Hammer PL, Ibaraki T, Mayoraz E, Muchnik I. 2000. An implementation of logical analysis of data. IEEE Transac Knowledge Data Eng 12:292–306.

Braga-Neto U, Dougherty E. 2003. Is cross-validation valid for small-sample microarray classification? Bioinformatics 20:374–380.

Breen EJ, Hopwood FG, Williams KL, Wilkins MR. 2000. Automatic poisson peak harvesting for high throughput protein identification. Electrophoresis 21:2243–2251.

Breiman L. 1996. Bagging predictors. Machine Learn 24:123–140.

Breiman L. 2001. Random forests. Machine Learn 45:5–32.

Breiman L, Friedman JH, Olshen RA, Stone CJ. 1984. Classification and regression trees. Belmont, CA: Wadsworth.

Bro R. 1997. PARAFAC. Tutorial and applications. Chemom Intell Lab Sys 38:149–171.

Brown MPS, Grundy WN, Lin D, Cristianini N, Sugnet CW, Furey TS, Ares M, Haussler D. 2000. Knowledge-based analysis of microarray gene expression data by using support vector machines. Proc Natl Acad Sci USA 97:262–267.

Burges CJC. 1998. A tutorial on support vector machines for pattern recognition. Data Mining Knowledge Discov 2:121–167.

Bylund D, Danielsson R, Malmquist G, Markides KE. 2002. Chromatographic alignment by warping and dynamic programming as a pre-processing tool for parafac modelling of liquid chromatography-mass spectrometry data. J Chromatogr A 961:237–244.

Campa M, Fitzgerald M, Patz E. 2003. Exploring the proteome with MALDI-TOF (Editorial). Proteomics 3:1659–1660.

Carroll JA, Beavis RC. 1996. Using matrix convolution filters to extract information from time-of-flight mass spectra. Rapid Commun Mass Spectrom 10:1683–1687.

Chamrad DC, Koerting G, Gobom J, Thiele H, Klose J, Meyer HE, Blueggel M. 2003. Interpretation of mass spectrometry data for high-throughput proteomics. Anal Bioanal Chem 376:1014–1022.

Check E. 2004. Running before we can walk? Nature 429:496–497.

Chen Y, Dougherty ER, Bittner ML. 1997. Ratio-based decisions and the quantitative analysis of cDNA microarray images. J Biomedical Optics 2:364–374.

Chen Y, Kamat V, Dougherty ER, Bittner ML, Meltzer PS, Trent JM. 2002. Ratio statistics of gene expression levels and applications to microarray data analysis. Bioinformatics 18:1207–1215.

Christian NP, Arnold RJ, Reilly JP. 2000. Improved calibration of time-of-flight mass spectra by simplex optimization of electrostatic ion calculations. Anal Chem 72:3327–3337.

Clarke W, Silverman B, Zhang Z, Chan DW, Klein AS, Molmenti EP. 2003. Characterization of renal allograft rejection by urinary proteomic analysis. Ann Surg 237:660–665.

Clauser KR, Baker P, Burlingame AL. 1999. Role of accurate mass measurement (+/− 10ppm) in protein identification strategies employing ms or ms/ms and database searching. Anal Chem 71:2871–2882.

Cleaveland WS. 1979. Robust locally weighted regression and smoothing scatterplots. J Amer Statist Assoc 74:829–836.

Cohen WW. 1995. Fast effective rule induction. Proceedings of 11th International Conference on Machine Learning, p 115–123.

Colinge J, Magnin J, Dessingy T, Giron M, Masselot A. 2003. Improved peptide charge state assignment. Proteomics 3:1434–1440.

Coombes KR, Fritsche HA, Clarke C, Chen JN, Baggerly KA, Morris JS, Xiao LC, Hung MC, Kuerer HM. 2003. Quality control and peak find for proteomics data collected from nipple aspirate fluid by surface-enhanced laser desorption and ionization. Clin Chem 49:1615–1623.

Coombes KR, Tsavachidis S, Morris JS, Baggerly KA, Hung MC, Kuerer HM. 2004. Improved peak detection and quantification of mass spectrometry data acquired from surface-enhanced laser desorption and ionization by denoising spectra with the undecimated discrete wavelet transform (Technical Report UTMDABTR-001-04). The University of Texas M. D. Anderson Cancer Center.

Cover T, Thomas J. 1991. Elements of information theory. New York: Wiley.

Cristianini N, Shawe-Taylor J. 2000. An introduction to support vector machines. Cambridge University Press.

Diamandis EP. 2003. Proteomic patterns in biological fluids: Do they represent the future of cancer diagnostics? Clin Chem 49:1272–1275.

Diamandis E. 2004. Analysis of serum patterns for early cancer diagnosis: Drawing attention to potential problems. J Natl Cancer Inst 96:353–356.

Dietterich T. 1998. Approximate statistical tests for comparing supervised classification learning algorithms. Neural Comput 10:1895–2024.

Domingos P. 2000. A unified bias-variance decomposition for zero-one and squared loss. Proceedings of the Seventeenth National Conference on Artificial Intelligence, p 564–569.

Duda R, Hart P, Stork D. 2000. Pattern classification. New York: Wiley.

Durbin BP, Hardin JS, Hawkins DM, Rocke DM. 2002. A variance-stabilizing transformation for gene-expression data. Bioinformatics 18:S105–S110.

Efron B, Tibshirani R. 1995. Cross-validation and the boostrap: Estimating the error rate of a prediction rule (Technical Report TR-477). Departement of Statistics, Standford University.

Egelhofer V, Bu1ssow K, Luebbert C, Lehrach H, Nordhoff E. 2000. Improvements in protein identification by MALDI-TOF-MS peptide mapping. Anal Chem 72:2741–2750.

Egelhofer V, Gobom J, Seitz H, Giavalisco P, Lehrach H, Nordhoff E. 2002. Protein identification by MALDI-TOF-MS peptide mapping: A new strategy. Anal Chem 74:1760–1771.

Eilers PHC. 2004. Parametric time warping. Anal Chem 76:404–411.

Fawcett T. 2003. ROC graphs: Notes and practical considerations for data mining researchers (Technical Report). HP Labs.

Feelders A, Verkooijen W. 1995. Which method learns most from the data. Proceedings of the 5th International Workshop on AI and Statistics, p 219–225.

Felitsyn N, Peschke M, Kebarle P. 2002. Origin and number of charges observed on multiply-protonated native proteins produced by esi. Int J Mass Spectrom 219:39–62.

Fenn JB, Mann M, Meng CP, Wang SF, Whitehouse CM. 1989. Electrospray ionization for mass spectrometry of large biomolecules. Science 246(4926):64–71.

Fleming CM, Kowalski BR, Apffel A, Hancock WS. 1999. Windowed mass selection method: A new data processing algorithm for liquid chromatography-mass spectrometry data. J Chromatogr A 849:71–85.

Forshed J, Schuppe-Koistinen I, Jacobsson SP. 2003. Peak alignment of NMR signals by means of a genetic algorithm. Anal Chim Acta 487:189–199.

Fraga CG, Bruckner CA, Synovec RE. 2001. Increasing the number of analyzable peaks in comprehensive two-dimensional separations through chemometrics. Anal Chem 73:675–683.

Freund Y, Schapire R. 1997. A decision theoretical generalization of on-line learning and an application to boosting. J Comput Syst Sci 55:119–139.

Fung E, Enderwick C. 2002. Proteinchip clinical proteomics: Computational challenges and solutions. Comput Proteomics Suppl 32:S34–S41.

Gay S, Binz PA, Hochstrasser DF, Appel RD. 1999. Modelling peptide mass fingerprinting data using the atomic composition of peptides. Electrophoresis 20:3527–3534.

Gentzel M, Köcher T, Ponnusamy S, Wilm M. 2003. Preprocessing of tandem mass spectrometric data to support automatic protein identification. Proteomics 3:1597–1610.

Gobom J, Mueller M, Egelhofer V, Theiss D, Lehrach H, Nordhof E. 2002. A calibration method that simplifies and improves accurate determination of peptide molecular masses by MALDI-TOF MS. Anal Chem 74:3915–3923.

Graber A, Juhasz PS, Khainovski N, Parker KC, Patterson DH, Martin SA. 2004. Result-driven strategies for protein identification and quantitation—A way to optimize experimental design and derive reliable results. Proteomics 4:474–489.

Gras R, Muller M, Gasteiger E, Gay S, Binz P, Bienvenut W, Hoogland C, Sanchez J, Bairoch A, Hochstrasser D, Appel R. 1999. Improving protein identification from peptide mass fingerprinting through a parameterized multi-level scoring algorithm and an optimized peak detection. Electrophoresis 20:3535–3550.

Grizzle WE, Semmes OJ, Basler J, Izbicka E, Feng Z, Kagan J, Adam BL, Troyer D, Srivastava S, Thornquist M, Zhang Z, Thompson IM. 2004. The Early Detection Research Network Surface-Enhanced Laser Desorption and Ionization Prostate Cancer Detection Study: A study in biomarker validation in genitourinary oncology. Urol Oncol 22:337–343.

Grushka E, Israeli D. 1990. Characterization of overlapped chromatographic peaks by their second derivative. The limit of the method. Anal Chem 62 :717–721.

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

Guyon I, Bitter HM, Ahmed Z, Brown M, Heller J. 2003. Multivariate non-linear feature selection with kernel multiplicative updates and Gram-Schmidt Relief. Proceedings of BISC FLINT CIBI 2003 Workshop. Berkeley.

Hall M, Holmes G. 2003. Benchmarking attribute selection techniques for discrete data class data mining. IEEE Transac Knowledge Data Eng 15:1437–1447.

Hastie T, Tibshirani R, Friedman J. 2001. The elements of statistical learning: Data mining, inference and prediction. New York: Springer.

Hastings CA, Norton SM, Roy S. 2002. New algorithms for processing and peak detection in liquid chromatography/mass spectrometry data. Rapid Commun Mass Spectrom 16:462–467.

Hedenfalk I, Duggan D, Chen Y, et al. MR. 2001. Gene expression profiles in hereditary breast cancer. N Engl J Med 344:539–548.

Hilario M, Kalousis A, Muller M, Pellegrini C. 2003. Machine learning approaches to lung cancer prediction from mass spectra. Proteomics 3:1716–1719.

Huang X, Pan W. 2002. Comparing three methods for variance estimation with duplicated high density oligonucleotide arrays. Funct Integr Genomics 2:126–133.

Huber W, vonHeydebreck A, Sultmann H, Poustka A, Vingron M. 2002. Variance stabilization applied to microarray data calibration and to the quantification of differential expression. Bioinformatics 18:S96–S104.

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.

Jain N, Thatte J, Braciale T, Ley K, O'Connell M, Lee JK. 2003. Local-pooled-error test for identifying differentially expressed genes with a small number of replicated microarrays. Bioinformatics 19:1945–1951.

Jarman KH, Daly DS, Anderson KK, Wahl KL. 2003. A new approach to automated peak detection. Chemom Intell Lab Sys 69:61–76.

Johnson KJ, Wright BW, Jarman KH, Synovec RE. 2003. High-speed peak matching algorithm for retention time alignment of gas chromatographic data for chemometric analysis. J Chromatogr A 996:141–155.

Jong K, Marchiori E, van derVaart A. 2004b. Analysis of proteomic pattern data for feature selection. Applications of Evolutionary Computing. EvoBIO: Evolutionary Computation and Bioinformatics. New York: Springer.

Jong K, Marchiori E, Sebag M, van derVaart A. 2004a. Feature selection in proteomic pattern data with support vector machines. IEEE Symp Comput Intell Bioinformatics Comput Biol, 41–48.

Karas M, Hillenkamp F. 1988. Laser desorption ionization of proteins with molecular masses exceeding 10,000 daltons. Anal Chem 60:2299–2301.

Kiers HAL, tenBerge JMF, Bro R. 1999. PARAFAC2 Part i. A direct fitting algorithm for the PARAFAC2 model. J Chemometrics 13:275–294.

King R, Bonfiglio R, Fernandez-Metzler C, Miller-Stein C, Olah T. 2000. Mechanistic investigation of ionization suppression in electrospray ionization. J Am Soc Mass Spectrom 11:942–950.

Kira K, Rendell L. 1992. The feature selection problem: Traditional methods and a new algorithm. Proc Natl Conf Artif Intell (AAAI-92), 129–134.

Kohavi R. 1995. A study of cross-validation and bootstrap for accuracy estimation and model selection. Proceedings of the 14th International Joint Conference on AI. Morgan Kaufman.

Kohavi R, John GH. 1997. Wrappers for feature subset selection. Artif Intell 97:273–324.

Kohonen T. 1995. Self-organizing maps. New York: Springer-Verlag.

Kononenko I. 2004. Estimating attributes: Analysis and extensions of RELIEF. Proceedings of European Conference on Machine Learning.

Koopmann J, Zhang Z, White N, Rosenzweig J, Fedarko N, Jagannath S, Canto MI, Yeo CJ, Chan DW, Goggins M. 2004. Serum diagnosis of pancreatic adenocarcinoma using suface-enhanced laser desorption and ionization mass spectrometry. Clin Cancer Res 10:860–868.

Kozak K, Amneus M, Pusey S, Su F, Luong M, Luong S, Reddy S, Farias-Eisner R. 2003. Identification of biomarkers for ovarian cancer using strong anion-exchange proteinchips: Potential use in diagnosis and prognosis. Proc Natl Acad Sci USA 100:12343–12348.

Kratzer R, Eckerskorn C, Karas M, Lottspeich F. 1998. Suppression effects in enzymatic peptide ladder sequencing using ultraviolet—Matrix assisted laser desorption/ionization—Mass spectormetry. Electrophoresis 19:1910–1919.

Kreil DP, Karp NA, Lilley KS. 2004. DNA microarray normalization methods can remove bias from differential protein expression analysis of 2d difference gel electrophoresis results. Bioinformatics 20:2026–2034.

Krutchinsky AN, Chait BT. 2002. On the nature of the chemical noise in MALDI mass spectra. J Am Soc Mass Spectrom 13:129–134.

Langley P. 1996. Elements of machine learning. San Mateo, CA: Morgan-Kaufmann.

Lee TA, Headley LM, Hardy JK. 1991. Noise reduction of gas chromatography/mass spectrometry data using principal component analysis. Anal Chem 63:357–360.

Lee KR, Lin X, Park DC, Eslava S. 2003. Megavariate data analysis of mass spectrometric proteomics data using latent variable projection method. Proteomics 3:1680–1686.

Li J. 1997. Development and evaluation of flexible empirical peak functions for processing chromatographic peaks. Anal Chem 69:4452–4462.

Li J, Zhang Z, Rosenzweig J, Wang YY, Chan DW. 2002. Proteomics and bioinformatics approaches for identification of serum biomarkers to detect breast cancer. Clin Chem 48:1296–1304.

Li J, Liu H, Ng SK, Wong L. 2003. Discovery of significant rules for classifying cancer diagnosis data. Bioinformatics 19:ii93–ii102.

Li L, Umbach DM, Terry P, Taylor JA. 2004. Application of the GA/KNN method to SELDI proteomics data. Bioinformatics 20:1638–1640.

Lilien RH, Farid H, Donald BR. 2003. Probabilistic disease classification of expression-dependent proteomic data from mass spectrometry of human serum. J Comput Biol 10:925–946.

Liu H, Li J, Wong L. 2002. A comparative study on feature selection and classification methods using gene expression profiles and proteomic patterns. Genome Inform 13:51–60.

Malmquist G. 1994. Multivariate evaluation of peptide mapping using the entire chromatographic profile. J Chromatogr A 687:89–100.

Mann M, Meng CK, Fenn JB. 1989. Interpreting mass spectra of multiply charged ions. Anal Chem 61:1702–1708.

Marchetti N, Felinger A, Pasti L, Pietrogrande MC, Dondi F. 2004. Decoding two-dimensional complex multicomponent separations by autocovariance function. Anal Chem 76:3055–3068.

Model F, Konig T, Piepenbrock C, Adorjan P. 2002. Statistical process control for large scale microarray experiments. Bioinformatics 18:S155–S163.

Mohammad-Djafari A, Giovannelli J, Demoment G, Idier J. 2002. Regularization, maximum entropy and probabilistic methods in mass spectrometry data processing problems. Int J Mass Spectrom 215:175–193.

Muddiman DC, Rockwood AL, Gao Q, Severs JC, Udseth HR, Smith RD. 1995. Application of sequential paired covariance to capillary electrophoresis electrospray ionization time-of-flight mass spectrometry: Unraveling the signal from the noise in the electropherogram. Anal Chem 67:4371–4375.

Muller M. 2003. Molecular scanner data analysis. Geneva: Doctoral dissertation University of Geneva.

Muller M, Gras R, Appel RD, Bienvenut WV, Hochstrasser DF. 2002a. Visualization and analysis of molecular scanner peptide mass spectra. J Am Soc Mass Spectrom 13:221–231.

Muller M, Gras R, Binz PA, Hochstrasser DF, Appel RD. 2002b. Improving protein identification for a molecular scanner experiment with human plasma by using correlations between spectra. Proteomics 2:1413–1425.

Neville P, Tan PY, Mann G, Wolfinger R. 2003. Generalizable mass spectrometry mining used to identify disease state biomarkers from blood serum. Proteomics 3:1710–1715.

Nielsen NPV, Carstensen JM, Smedsgaard J. 1998. Aligning of single and multiple wavelength chromatographic profiles for chemometric data analysis using correlation optimised warping. J Chromatogr A 805: 17–35.

Palmblad M, Buijs J, Hakansson P. 2001. Automatic analysis of hydrogen/deuterium exchange mass spectra of peptides and proteins using calculations of isotopic distributions. J Am Soc Mass Spectrom 12: 1153–1162.

Papadopoulos M, Abel PM, Agranoff D, Stich A, Tarelli E, Bell BA, Planche T, Loosemore A, Saadoun S, Wilkins P, Krishna S. 2004. A novel and accurate diagnostic test for human african trypanosomiasis. Lancet 363:1358–1363.

Peng WP, Cai Y, Chang HC. 2004. Optical detection methods for mass spectrometry of macroions. Mass Spectrom Rev 23:443–465.

Pepe MS. 1995. Receiver operating characteristic methodology. J Am Stat Assoc 95:308–311.

Perkins DN, Pappin DJC, Creasy DM, Cottrell JS. 1999. Probability-based protein identification by searching sequence databases using mass spectrometry data. Electrophoresis 20:3551–3567.

Petricoin E, Liotta LA. 2003. The vision of a new diagnostic paradigm. Clin Chem 49:1276–1278.

Petricoin EF, Ardekani AM, Hitt BA, Levine PJ, Fusaro VA, Steinberg SM, Mills GB, Simone C, Fishman DA, Kohn EC, Liotta LA. 2002. Use of proteomic patterns in serum of identify ovarian cancer. Lancet 359:572–577.

Poon T, Yip TT, Chan A, Yip C, Yip V, Mok T, Lee C, Leung T, Ho S, Johnson PJ. 2003. Comprehensive proteomic profiling identifies serum proteomic signatures for detection of hepatocellular carcinoma and its subtypes. Clin Chem 49:752–760.

Powell DA, Anderson LM, Cheng RYS, Alvord WG. 2002. Robustness of the Chen–Dougherty–Bittner procedure against non-normality and heterogeneity in the coefficient of variation. J Biomedical Optics 7:650–660.

Prados J, Kalousis A, Sanchez JC, Allard L, Carrette O, Hilario M. 2004. Mining mass spectra for diagnosis and biomarker discovery of cerebral accidents. Proteomics 4:2320–2332.

Press WH, Teukolsky S, Vetterling WT, Flannery BP. 1995. Numerical recipies in C. Cambridge UK: Cambridge University Press.

Provost F, Fawcett T. 2001. Robust classification for imprecise environments. Machine Learn 42:203–231.

Purohit PV, Rocke DM. 2003. Discriminant models for high-throughput proteomics mass spectrometer data. Proteomics 3:1699–1703.

Qu Y, Adam BL, Yasui Y, Ward MD, Cazares LH, Schellhammer PF, Fend Z, Semmes OJ, Wright GL Jr. 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.

Qu Y, Adam Bl, Thornquist M, Potter JD, Thompson ML, Yasui Y, Davis J, Schellhammer PF, Cazares L, Clements M, Wright GL, Feng Z. 2003. Data reduction using a discrete wavelet transform in discriminant analysis of very high dimensional data. Biometrics 59:143–151.

Quinlan JR. 1993. C4.5: Programs for machine learning. San Mateo, CA: Morgan Kaufmann.

Quinlan JR. 1994. Comparing connectionist and symbolic learning methods. In: Hanson SJ, Drastal GA, Rivest RL, editors. Computational learning theory and natural learning systems. Vol. I. chapter 15. Cambridge, MA: MIT Press. p 446–456.

Rai AJ, Zhang Z, Rosenzweig J, Shih JM, Pham T, Fung ET, Sokoll LJ, Chan DW. 2002. Proteomic approaches to tumor marker discovery. Arch Pathol Lab Med 126:1518–1526.

Reinhold BB, Reinhold VN. 1992. Electrospray ionisation mass spectrometry: Deconvolution by an entropy-based algorithm. J Am Soc Mass Spectrom 3:207–215.

Rockwood AL, Orden SLV, Smith RD. 1995. Rapid calculation of isotope distributions. Anal Chem 67:2699–2704.

Rockwood AL, Orden SLV. 1996. Ultrahigh-speed calculation of isotope distributions. Anal Chem 68:2027–2030.

Rogers MA, Clarke P, Noble J, Munro NP, Paul A, Selby PJ, Banks RF. 2003. Proteomic profiling of urinary proteins in renal cancer by surface enhanced laser desorption ionization and neural-network analysis: Identification of key issues affecting potential clinical utility. Cancer Res 63:6971–6983.

Sadygov RG, Eng J, Durr E, Saraf A, McDonald H, MacCoss MJ, Yates JR III. 2002. Code developments to improve the efficiency of automated MS/MS spectra interpretation. J Proteome Res 1:211–215.

Samuelsson J, Dalevi D, Levander F, Rognvaldsson T. 2004. Modular, scriptable, and automated analysis tools for high-throughput peptide mass fingerprinting. Bioinformatics 20:3628–3635.

Satten GA, Datta S, Moura H, Woolfitt AR, Carvalho MG, Carlone GM, De BK, Pavlopoulos A, Barr JR. 2004. Standardization and denoising algorithms for mass spectra to classify whole-organism bacterial specimens. Bioinformatics 20:3128–3136.

Savitzky A, Golay M. 1964. Smoothing and differentiation of data by simplified least squares procedure. Anal Chem 36:1627–1639.

Schmidt F, Schmid M, Jungblut PR, Mattow J, Facius A, Pleissner KP. 2003. Iterative data analysis is the key for exhaustive analysis of peptide mass fingerprints from proteins separated by two-dimensional electrophoresis. J Am Soc Mass Spectrom 14:943–956.

Schölfkopf B, Tsuda K, Vert J. 2004. Kernel methods in computational biology. Cambridge, MA: MIT Press.

Schölkopf B, Guyon I, Weston J. 2003. Statistical learnining and kernel methods in bioinformatics. In: Frasconi P, Shamir R, editors. Artificial intelligence and heuristic methods in bioinformatics. Amsterdam: IOS Press. p 1–21.

Senko MW, Beu SC, McLafferty FW. 1995. Automated assignement of charge states from resolved isotopic peaks for multiply charged ions. J Am Soc Mass Spectrom 6:52–56.

Shackmana JG, Watson CJ, Kennedy RT. 2004. High-throughput automated post-processing of separation data. J Chromatogr A 1040:273–282.

Shao XG, Leung AK, Chau FT. 2003. Wavelet: A new trend in chemistry. Acc Chem Res 36:276–283.

Shao XG, Cai W, Sun P, Zhang M, Zhao G. 1997. Quantitative determination of the components in overlapping chromatographic peaks using wavelet transform. Anal Chem 69:1722–1725.

Shi SDH, Hendrickson CL, Marshall AG. 1998. Counting individual sulfur atoms in a protein by ultrahigh-resolution fourier transform ion cyclotron resonance mass spectrometry: Experimental resolution of isotopic fine structure of proteins. Proc Natl Acad Sci USA 95:11532–11537.

Simon R, Radmacher M, Dobbin K, McShane LM. 2003. Pitfalls in the use of DNA microarray data for diagnostic and prognostic classification. J Natl Cancer Inst 95:14–18.

Soille P. 2003. Morphological image analysis. Berlin: Springer Verlag.

Somorjai RL, Dolenko B, Baumgartner R. 2003. Class prediction and discovery using gene microarray and proteomics mass spectroscopy data: Curses, caveats, cautions. Bioinformatics 19:1484–1491.

Sorace JM, Zhan M. 2003. A data review and re-assessment of ovarian cancer serum proteomic profiling. BMC Bioinformatics 4:24.

Tammen H, Mohring T, Kellmann M, Pich A, Kreipe HH, Hess R. 2004. Mass spectrometric phenotyping of Val34Leu polymorphism of blood coagulation factor XIII by differential peptide display. Clin Chem 50:545–551.

Tang K, Page JS, Smith RD. 2004. Charge competition and the linear dynamic range of detection in electrospray ionization mass spectrometry. J Am Soc Mass Spectrom 15:1416–1423.

Tibshirani R, Hastie T, Narasimhan B, Chu G. 2002. Diagnosis of multiple cancer types by shrunken centroids of gene expression. Proc Nat Acad Sci USA 99:6567–6572.

Tibshirani R, Hastie T, Narasimhan B, Soltys S, Shi G, Koong A, Le QT. 2004. Sample classification from protein mass spectrometry by "peak probability contrasts". Bioinformatics 20:3034–3044.

Tukey JW. 1992. Tightening the clinical trial. Control Clin Trials 14:266–285.

Turney P. 1995. Technical note: Bias and the quantification of stability. Machine Learn 20:23–23.

Tusher VG, Tibshirani R, Chu G. 2001. Significance analysis of microarrays applied to the ionizing radiation response. Proc Natl Acad Sci USA 98:5116–5121.

Vapnik V. 1998. Statistical learning theory. New York: Wiley.

Venable JD, Dong MQ, Wohlschlegel J, Dillin A, Yates JR III. 2004. Automated approach for quantitative analysis of complex peptide mixtures from tandem mass spectra. Nature Methods 1:39–45.

Vestal M, Juhash P. 1998. Resolution and mass accuracy in matrix-assisted laser desorption ionization time-of-flight. J Am Soc Mass Spectrom 9:892–911.

Wagner M, Naik D, Pothen A. 2003. Protocols for disease classification from mass spectrometry data. Proteomics 3:1692–1698.

Wagner M, Naik DN, Pothen A, Kasukurti S, Devineni RR, Adam BL, Semmes OJ, Wright GL, Jr. 2004. Computational protein biomarker prediction: A case study for prostate cancer. BMC Bioinformatics 5.

Wallace WE, Kearsley AJ, Guttman CM. 2004. An operator-independent approach to mass spectral peak identification and integration. Anal Chem 76:2446–2452.

Wang CP, Isenhour TL. 1987. Time-warping algorithm applied to chromatographic peak matching gas chromatography/fourier transform infraredimass spectrometry. Anal Chem 59:649–654.

Wang M, Howard B, Campa M, Patz EJ, Fitzgerald M. 2003. Analysis of human serum proteins by liquid phase isoelectric focusing and matrix-assisted laser desorption/ionization-mass spectrometry. Proteomics 3:1661–1666.

Washburn MP, Wolters D, Yates JR III. 2001. Large-scale analysis of the yeast proteome by multidimensional protein identification technology. Nat Biotechnol 19:242–247.

Watkins B, Szaro R, Ball S, Knubovets T, et al. 2001. Detection of early-stage cancer by serium protein analysis. Am Lab 32:31–36.

Windig W, Phalp JM, Payne AW. 1996. A noise and background reduction method for component detection in liquid chromatography/mass spectrometry. Anal Chem 68:3602–3606.

Wolpert DH. 1992. Stacked generalization (Technical Report LA-UR-90-3460). Los Alamos, NM.

Won Y, Song HJ, Kang TW, Kim JJ, Han BD, Lee SW. 2003. Pattern analysis of serum proteome distinguishes renal cell carcinoma from other urologic diseases and healthy persons. Proteomics 3:2310–2316.

Wool A, Smilansky Z. 2002. Precalibration of matrix-assisted laser desorption/ionization-time of flight spectra for peptide mass fingerprinting. Proteomics 2:1365–1373.

Wu B, Abbott T, Fishman D, McMurray W, Mor G, Stone K, Ward D, Williams K, Zhao H. 2003. Comparison of statistical methods for classification of ovarian cancer using mass spectrometry data. Bioinformatics 19:1636–1643.

Wulfkuhle J, Liotta L, Petricoin EF. 2003. Proteomic applications for the early detection of cancer. Nature Rev 3:267–276.

Yanasigawa K, Shyr Y, Xu B, Massion P, Larsen PH, White BC, Roberts JR, Edgerton M, Gonzalez A, Nadaf S, Moore JH, Caprioli RM, Carbone DP. 2003. Proteomic patterns of tumour subsets in non-small-cell lung cancer. Lancet 362:433–439.

Yang YH, Dudoit S, Luu P, Lin DM, Peng V, Ngai J, Speed TP. 2002. Normalization for cDNA microarray data: A robust composite method addressing single and multiple slide systematic variation. Nucleic Acids Res 30.

Yasui Y, Pepe M, Thompson M, Adam B, G Wright J, Qu Y, Potter J, Winget M, Thornquist M, Feng Z. 2003. A data-analytic strategy for protein biomarker discovery: Profiling of high-dimensional proteomic data for cancer detection. Biostatistics 4:449–463.

Zhang Z, Guan S, Marshall AG. 1997. Enhancement of the effective resolution of mass spectra of high-mass biomolecules by maximum entropy-based deconvolution to eliminate the isotopic natural abundance distribution. J Am Soc Mass Spectrom 8:659–670.

Zhang Z, Marshall AG. 1998. A universal algorithm for fast and automated charge state deconvolution of electrospray mass-to-charge ratio spectra. J Am Soc Mass Spectrom 9:225–233.

Zhang Z, McElvain JS. 1999. Optimizing spectroscopic signal-to-noise ratio in analysis of data collected by a chromatographic/spectroscopic system. Anal Chem 71:39–45.

Zhu H, Yu CY, Zhang H. 2003a. Tree-based disease classification using protein data. Proteomics 3:1673–1677.

Zhu W, Wang X, Ma Y, Rao M, Glimm J, Kovach JS. 2003. Detection of cancer-specific markers amid massive mass-spectral data. Proc Natl Acad Sci USA 100:14666–14671.

Zien A, Aigner T, Zimmer R, Lengauer T. 2001. Centralization: A new method for the normalization of gene expression data. Bioinformatics 17:S323–S331.

Melanie Hilario holds her Ph.D. in Computer Science from the University of Paris VI and currently works at the University of Geneva's Artificial Intelligence Laboratory. She has initiated several European research projects on issues such as integrating neural networks and symbolic processing, meta-learning, and biological text mining. Her current research interests include mass-spectra based biomarker discovery, genomic/proteomic data mining, and automated information extraction from MEDLINE documents. In 2004, her publication on "Classifying Protein Fingerprints" won the Best Paper Award in the European Conference on Principles and Practice of Knowledge Discovery in Databases. In 2005, she was a Program Committee member for more than six international machine learning and data mining conferences and workshops. She is currently an Associate Editor of the International Journal on Artificial Intelligence Tools.

Alexandros Kalousis his B.Sc. in Computer Science and an M.Sc. in Advanced Information Systems from the University of Athens. In 1997, he started a Ph.D. thesis at the same university in the area of Machine Learning to successfully complete it 5 years later at the University of Geneva, where he continues as a senior researcher. He has published widely in the area of machine learning and knowledge discovery, in particular on meta-learning, data preprocessing, feature extraction, model selection, and model evaluation. He has also worked extensively on proteomics-related applications such as classification of mass-spectral data. Currently, his research interests include the quantification of model stability as well as machine learning in structured domains using distances and kernels.

Christian Pellegrini holds an M.Sc. in high energy physics and a Ph.D. in Computer Science. He was "IBM Post-Doctoral Fellow" at T. J. Watson Research Center, New York from 1977 to 1978. He was nominated research associate and lecturer in Computer Science at the University of Geneva (Switzerland) in 1978, associate professor of Computer Science in 1981, and full professor of Computer Science in 1982. He held the position of vice-dean of the Faculty of Sciences from 1992 to 1998. He is currently Chairman of the Computer

Science Department of the Faculty of Sciences of the University of Geneva since 2000. His research interests are mainly in artificial intelligence, machine learning, heuristic programming, simulation of human cognition, and neurobiological simulation.

Markus Müller holds a Ph.D. in Bioinformatics from the University of Geneva. He has been involved for several years in the development of algorithms and software for the analysis of proteomic MS data. He wrote programs for the detection of peptide signals in MALDI and ESI spectra. In many cases, such as the molecular scanner or LC-MS, mass spectra are not measured independently and their correlation was used to enhance the signal to noise ratio and to improve the identification of proteins. Further work included statistical interpretation of a PMF scoring schema and a study about protein expression patterns for biomarker detection. Currently, he has the position of group leader for bioinformatics at the Institute of Molecular Systems Biology in Zurich, Switzerland. His current research interests are MS/MS identification, LC-MS data processing, and the identification of protein complexes by means of mass spectrometry.