Artificial Intelligence and Expert Systems in Mass Spectrometry

Ronald C. Beavis, Steven M. Colby, Royston Goodacre, Peter de B. Harrington, James P. Reilly, Stephen Sokolow, and Charles W. Wilkerson

in
Encyclopedia of Analytical Chemistry
R.A. Meyers (Ed.)
pp. 11558–11597
© John Wiley & Sons Ltd, Chichester, 2000

Artificial Intelligence and Expert Systems in Mass Spectrometry

Ronald C. Beavis
Proteometrics LLC, New York, NY, USA

Steven M. Colby
Scientific Instrument Services, Inc., Ringoes, NJ, USA

Royston Goodacre
University of Wales, Aberystwyth, UK

Peter de B. Harrington
Ohio University, Athens, OH, USA

James P. Reilly
Indiana University, Bloomington, IN, USA

Stephen Sokolow
Bear Instruments, Santa Clara, CA, USA

Charles W. Wilkerson
Los Alamos National Laboratory, Los Alamos, NM, USA

This article provides a brief introduction to aspects of mass spectrometry (MS) that employ artificial intelligence

Encyclopedia of Analytical Chemistry
R.A. Meyers (Ed.) Copyright © John Wiley & Sons Ltd

(AI) and expert system (ES) technology. These areas have grown rapidly with the development of computer software and hardware capabilities. In many cases, they have become fundamental parts of modern mass spectrometers.

Specific attention is paid to applications that demonstrate how important features of MS are now dependent on AI and ESs. The following topics are specifically covered: history, MS data systems, biological applications, artificial neural networks (ANNs), and optimization techniques.

1 INTRODUCTION

1.1 Definitions of Artificial Intelligence and Expert Systems

This article covers the application of AI and ESs as applied to the techniques of MS. ESs are methods or programs by which a fixed set of rules or data is used to control a system, analyze data, or generate a result. In contrast, AI is associated with the higher intellectual processes, such as the ability to reason, discover meanings, generalize, or learn. In relation to MS, AI is generally limited to cases wherein ANNs are employed to learn or discover new patterns or relationships between data. Reviews of AI and ESs are available.[1,2]

1.2 Growth in Technology

The growth in MS has been spurred by improvements in software sophistication and computer capabilities. The ability of computing systems to both collect and analyze data has grown very rapidly since the 1970s. The most important improvements have been in calculation speed of the machines, their ability to store large amounts of data very quickly, and their size. These improvements have allowed processes such as multitasking during data acquisition, where the computer both collects data and controls the instrument operation, and automated spectral matching, where large volumes of data are quickly analyzed.

The improvements in computer technology have resulted in an increase in the performance and types of mass spectrometers available. For example, instruments with even the simplest types of mass analyzers are now computer controlled. This has dramatically increased the stability, reproducibility, and capabilities of these devices. It is now possible to perform a tandem mass spectrometry (MS/MS) experiment where the data collection parameters are changed on the millisecond timescale in response to the data collected.[3] This allows a library search to be performed, possible match candidates to be experimentally tested, and a positive identification to be made, all during the elution of a chromatography peak.

The development of computers has also allowed the use of new types of MS. For example, the data generated by Fourier transform mass spectrometry (FTMS), pyrolysis, and electrospray MS would be very difficult if not impossible to collect and analyze without high speed computers.

1.3 Article Summary

This article includes sections on the history of computers in MS, MS data systems, biological applications, MS applications of principal component analysis (PCA) and factor analysis (FA), ANNs, and optimization techniques in MS. This article does not include a discussion of the use and development of libraries of electron impact ionization data or of peak deconvolution and component identification based on these libraries. Reviews of these topics are available.[4,5]

2 BRIEF HISTORY OF COMPUTERS IN MASS SPECTROMETRY

2.1 Introduction

Digital computers are now an indispensable part of most analytical instruments. There are many reasons for this pervasive presence. Perhaps most important is the ability of computers to perform repetitive tasks without variation (in the absence of hardware failure), which is critical to reproducible and defensible experimental results. Further, properly designed and implemented computer control/data systems maximize instrument and laboratory efficiency, resulting in higher sample throughput, faster results to the end-user, and increased profitability (either in terms of publications or currency) for the laboratory. As a technique that arguably provides more chemical and structural information per unit sample than any other, MS has been employed in a variety of environments over its long history. The evolution of the mass spectrometer from a fundamental research tool for the elucidation of atomic and molecular properties to a benchtop turnkey instrument, in large measure parallels the evolution of both discrete and integrated electronic devices and computational hardware and software.

In this article no attempt is made to tabulate an exhaustive list of historical references to the application of computers in MS, but rather selected citations are presented to provide a flavor of the development in the field. There is one monograph dedicated to computers in MS,[6] and the topic is given treatments ranging from cursory to complete in a variety of books on mass spectrometric techniques.

2.2 Early Devices

Early mass analyzers were spatially dispersive instruments, or mass spectrographs,[7] utilizing static magnetic or DC (direct current) electric fields to perturb the

trajectories of accelerated ions. At the time of their development (ca. 1910–1920) photographic plates were placed in the focal plane of the spectrograph, and after exposure to the ion beam an image was developed and the resulting data analyzed. Quantitative analyses were effected by comparing the amount of exposure on a film produced when the unknown sample was determined to calibrated plates developed after measuring known amounts of reference materials. This technique is still in use today in certain specialized (and somewhat archaic) applications. In the 1930s and 1940s detectors based on direct measurement of ion beam flux (such as the Faraday cup and electron multiplier), were introduced. Such detectors are single-channel transducers, and require that slits be positioned (in a dispersive instrument) to limit the exposure of the detector to a single mass at any given time. The signal is then amplified and recorded as a function of some independent variable (such as magnetic field strength, or the ion accelerating voltage) that is proportional to the mass-to-charge ratio (m/z) of the ions in the sample.

With the introduction of electronic detectors, it became practical to couple detector output to a digital computer via some type of interface. For low-intensity signals, such as measurement of discrete ions, pulse-counting techniques are employed. As this is inherently a digital process, transmission of data to a computer is relatively straightforward. Larger signals, characterized by significant and measurable detector currents, employ analog-to-digital converters (ADCs) prior to storage and manipulation on the computer.

2.3 Instrument Design

2.3.1 Time-of-flight

Time-of-flight (TOF) mass spectrometers were first developed in 1932,[8] but the most familiar design, which forms the basis of current instruments, was described by Wiley and McLaren in 1955.[9] Accurate measurement of ion time of arrival at the detector is key to achieving optimum resolving power and mass accuracy with this instrument. Prior to the introduction of computer data acquisition, oscillographic recording was required, with manual postprocessing.

2.3.2 Quadrupole

The quadrupole mass filter was first described in 1958.[10] The advantages of this instrument include small size, low ion energy (volts rather than kilovolts for dispersive and TOF instruments), modest production costs, and the ability to quickly scan through a wide range of m/z values. As a result, this design has become by far the most popular variety of mass spectrometer. A related mass analyzer, the quadrupole ion trap, was not widely developed until the 1970s.[11] Like the linear quadrupole mass filter, the ion trap is small, inexpensive, and robust. Both of these devices rely on the application of concerted radiofrequency (RF) and DC fields in order to define conditions under which ions have stable trajectories in the instrument.

2.3.3 Ion Cyclotron Resonance

The ion cyclotron resonance (ICR) mass spectrometer, first reported in 1968,[12] relies on the absorption of RF energy and the natural precession of charged particles in the presence of a magnetic field for mass separation. Nominal resolving power is obtained in this instrument when operated in a continuous scanning mode, where the RF frequency is slowly swept and energy is absorbed when ions in the cell are resonant with the excitation. The most common incarnation of ICR is often referred to as FTMS,[13] and spectral information is extracted from the digitally recorded decay and dephasing of ion orbits after a pulsed application of RF energy. This approach allows for significantly improved resolving power (1000-fold improvement) over the scanning experiment.

2.4 Computerization

As a result of the widespread availability of minicomputers in the late 1960s, and microcomputers in the 1970s and 1980s, automation of mass spectrometer control, tuning, data acquisition, and data processing became practical. The reduction in both size and cost of computational engines, with a concomitant increase in processing power, cannot be overemphasized in the development of automated data systems for mass spectrometers (and other analytical instrumentation). Certainly, the widespread implementation of gas chromatography/mass spectrometry (GC/MS) would have been significantly delayed without the availability of reasonably priced quadrupole mass spectrometers and minicomputer-based data acquisition and processing equipment. The operation of FTMS would be nearly impossible without the involvement of computers.

2.5 Brief Introduction to Artificial Intelligence and Expert Systems

Almost from the beginning of the digital computing era, both hardware and software engineers have been interested in developing computing tools that can monitor their environments, and subsequently make decisions and/or carry out actions based on rules either known a priori – from programming – or deduced – as a result of iterative observation/decision/feedback experiences. Such computational devices may be called ‘expert systems’, or may be said to operate based on ‘artificial

intelligence'. It is certainly beyond the scope of this article to provide a complete history of AI and ES, but there are a multitude of both books and research articles related to this topic.[14,15] Today, many parts of our world are monitored, and in some cases controlled, by automated, computerized equipment. In an effort to make these devices more responsive and efficient, many of them employ embedded ESs of varying degrees of sophistication. Programming languages such as LISP and PROLOG have been developed specifically to facilitate the development of software to implement AI and ES. The combination of powerful hardware, innovative algorithms, and capture of years of expert knowledge has allowed instruments to become increasingly independent of operator interaction, reducing the possibility for error and allowing the scientist to concentrate on the interpretation of the processed data and the formulation of new experiments.

2.6 Brief Overview of Artificial Intelligence and Expert Systems in Mass Spectrometry

In the world of MS, AI and ES tools are used in three primary areas: optimization and control of the performance of the mass spectrometer itself, collection of the detector signal as a function of m/z, and analysis of the data.

2.6.1 Spectrometer Control

There are many instrumental parameters that need to be adjusted and held at an optimum value for best spectrometer performance. Initially, the instrument must be tuned, i.e. brought to a state in which peak intensity, peak shape, and mass calibration are all within acceptable limits. This is accomplished by introducing a known compound, such as PFTBA (perfluorotributylamine), into the spectrometer that produces a variety of well-characterized fragments over the mass range of interest, and adjusting (in an optimized fashion) the various instrument parameters to achieve the desired level of performance. Computers are almost invariably used to perform this task, because the adjustable parameters are often highly interrelated (repeller voltage, ion focusing lens potentials, electron multiplier voltage, mass scan rate, ion storage time, chemical ionization reagent gas pressure, time delay for ion extraction, etc.). Techniques such as simplex optimization are used to efficiently locate in parameter space the best-tune conditions. After tuning is complete, the computer can then monitor all of the vital signs of the instrument during operation, and alert the spectrometrist of marginal performance conditions, and even recommend appropriate interventions, before data quality is affected.

2.6.2 Data Collection

In almost all data systems, the operator uses the computer to define the scope of the measurements to be made. Subsequently, the computer sets instrument parameters to control, for example, the speed of data collection, the mass range to be recorded, and other instrument type-dependent variables. As the experiment is performed, the computer records the detector signal via either a direct digital interface (for counting experiments) or an ADC. Correlation of the detector signal with the corresponding m/z condition is accomplished through a mass-axis calibration routine. Depending on the mass spectrometer type, this may be a DC, RF, or time reference.

2.6.3 Data Analysis

After the data have been collected, their chemical information must be extracted and interpreted. There has been a significant amount of development in the area of data analysis software since the first report of such use in 1959.[16] In this early work, a system of simultaneous linear equations were used to convert raw peak areas to normalized analyte mole fractions. A 17-component sample required 0.5–3 min of computing time for processing. Today, mixtures with nearly an order of magnitude more analytes can be reduced in less time, providing significantly more information than simply peak quantitation. In addition to quantifying analytes, mass spectrometer data systems routinely provide identification of species from their mass spectral fingerprints. One of the earliest examples of the application of AI to mass spectral interpretation was the work of Djerassi et al.[17] A LISP (a list processing language)-based code, DENDRAL, was developed and subsequently applied to a variety of analyte classes. Most mass spectrometrists are familiar with spectral libraries, ranging from the print version of the so-called eight-peak index[18] to the most modern computerized systems. The latter use intelligent peak-searching and pattern-matching algorithms to provide the operator with the most likely identities of species in a spectrum.

3 MASS SPECTROMETRY DATA SYSTEMS

3.1 Introduction

Since the mid-1970s the programming of mass spectral data systems has changed enormously. Although the basic tasks of an MS data system are fundamentally the same now as they were in the 1970s, many of the numbers involved have become substantially larger. In addition, developing mass spectral technologies such as FTMS have placed very heavy demands on the acquisition process.

Spectrum libraries have become larger. Analyses of large complex molecules (i.e. peptides) may consume a great deal of computer resources. Fortunately, the changes in computer and operating system technologies since the 1970s have been even more staggering than the changes in MS.

Section 3.2 defines the basic tasks of a MS data system. Section 3.3 describes the requirements imposed on the computers and operating systems that aspire to perform these tasks. Section 3.4 examines some of the specifics of how changes in computer technology have affected mass spectral data systems. Section 3.5 treats the subject of programmability. As the number of MS algorithms proliferate, the need for a data system to be customizable (i.e. programmable) has become ever more important – if users cannot define their own ways of collecting and analyzing data, unlimited computer power may be useless. Practical examples from actual data systems are presented, to show that the concerns of a programmer are often quite different from those of a chemist.

3.2 Fundamental Tasks of a Data System

The tasks of an MS data system are often neatly divided into instrument control, acquisition of data to a storage medium, and analysis of the data. The division is, of course, not really so simple. The collection of data depends significantly on simultaneous instrument control and the analysis of the collected data may be fed back into the instrument control. For example, in the process of tuning an instrument, the software may vary a variety of different parameters, each time collecting and assessing some data before trying a new set of conditions. In this case there is a feedback loop that involves control, acquisition, and analysis. The feedback must be very tightly orchestrated to be useful.

3.2.1 Instrument Control

The task of instrument control has several aspects – routine operation, instrument protection, tuning, and diagnostic programs. During routine operation many voltages must be set or scanned, and as much instrument status as possible must be read from the instrument. This status information may be stored with the data. It may be used to keep temperatures stable within the instrument by running PID (Proportional–Integral–Differential) loops on heaters. Or, it may be used to protect the instrument. For example, a sudden rise in pressure may indicate a leak and some voltages should be turned off. If mass peaks are saturated, perhaps the detector voltage should be decreased, or a warning message should be shown on the computer screen. The process of tuning and diagnostic programs, each in their own way, are microcosms of the entire MS data system. Those experienced in designing MS data systems have learned that it is advantageous to first write the diagnostic programs, basing them on very small and easily understood modules. These will, after all, be needed for the first evaluation of the instrument. It is then possible to base the ordinary operation of the instrument on these same modules. Doing so tends to provide the entire system with a relatively good structure. This bottom-up modular structure also makes it easy to add elementary operations (e.g. when adding new hardware) and higher-level operations can almost always be defined as combinations of the elementary processes.

3.2.2 Data Collection

The task of data collection is fundamentally important. Today's computer operating systems are multitasking and therefore capable of running several processes at once. If the mass spectrometer is connected to a chromatograph or other time-dependent sample-introduction device, then the data collection must have priority over all other operations. A disaster can result if some data are missed. To guard against this, an MS data system may use more than one processor, dedicating at least one processor to data collection.

3.2.3 Data Analysis

Analysis of the collected data includes the following items:

- Conversion of raw (e.g. profile or Fourier-transform) data to mass peaks.
- Data display for the chemist.
- Enhancement of the data by background subtraction or other means.
- Use of the area under chromatogram peaks or other MS data to compare unknowns with standards and to achieve quantitative results.
- Library searching.
- Report generation.

A modern data analysis program includes other more advanced topics, which are covered elsewhere in this article; even the elementary operations listed above have many variations. Data systems must be flexible enough to allow the user to perform the operations in exactly the way and order required, hence the importance of programmability. The control, collection and analysis are all achieved through a user interface. This element of the data system determines the ways in which the user is able to enter information and communicate with the system. Section 3.4 looks at how changes in operating systems have affected the user interface and hence the ease of using mass spectral data systems.

It should be noted that, from the programmer's point of view, the design of an easy-to-use user interface is generally a much harder and time-consuming part of the programmer's task than implementing all of the chemical algorithms. The user interface includes the display of data and instrument status, as well as input devices such as menus and buttons that allow the user to control the system.

The display must respond to real changes in instrument status in a timely manner. For example, suppose that in the process of tuning an instrument the user is manually increasing a voltage setting by clicking a button on the screen. If nothing happens to the status display for more than a second, the user is very likely to click on the button again to accelerate the change in the system. This is simply because faster computer response time has naturally led to greater user impatience. However, overclicking can result in overshooting an optimum setting and this makes instrument adjustment almost impossible. Therefore, a crucial task of the data system to reflect the real-time status of the instrument.

3.3 Requirements for Operating Systems

As noted above, data collection must never fail. As the operating system used by a chemist is almost certainly a multitasking system, it is necessary to ensure that the highest possible priority is given to the data collection task. It must not be possible for other tasks to usurp the precious time required by the data collection procedures. This is the overriding concern in the selection of an operating system. For this reason Windows NT is a much more appropriate choice than Windows 95, for MS data systems. Several other operations also require high priority because they cannot be interrupted, such as those that involve delicate timing or real-time feedback.

If multiple processors are used, other requirements must be considered. For example, if an embedded processor in the instrument communicates with the data system over a serial or parallel line, it is important that the instrument be plug-and-play; that is, both sides should disconnect cleanly when the cable is disconnected and reconnect automatically when the cable is reconnected. If the embedded processor is depending on the data system for control and the connection is broken, the embedded processor should go into a standby state for safety purposes.

Most instrument manufacturers have chosen to base their data systems on PCs running Microsoft operating systems. A brief survey of 22 instrument manufacturers found that 18 of them were using a version of Microsoft Windows. Others used OS/2, and operating systems from Hewlett-Packard, Sun, and Apple.

3.4 Impact of Continuing Advances in Computers on Mass Spectrometry Data Systems

The most obvious improvements in computers have been the dramatic increases in speed and in the size of computer memories and storage. The forefathers of today's data systems were developed on home-built computers using Intel chipsets or on systems produced by Data General, Digital Equipment, Commodore, or Apple (section 2). These systems typically had 16–64 kB of ram and sometimes included a 5 or 10 MB disk. Since the 1970s the availability of memory and storage has increased by over three orders of magnitude. Execution times have also increased, albeit to a lesser extent. For example, library searches are now four to eight times faster.

Operations that require large arrays of data and massive amounts of arithmetic have benefited most from the improvements in hardware design. These improvements have also made it much easier to implement algorithms. Previously, developers had to implement programming tricks to handle very large arrays of data. Activities such as library searches required extensive coding in order for their execution to be completed in a reasonable amount of time. Today even more advanced and thorough searches can be implemented with a few lines of C code. These advantages also apply to algorithms written by the user of the data system (if a programming language is available – see section 3.5).

Networks are beginning to have a major impact on data systems. Local networking provides a great advantage by giving the user a wide variety of high-capacity storage options. The internet allows easier transfer of data and results, but has found only limited use in instrument control. In both cases security issues are a major concern. Although most laboratory management systems provide security features, such as passwords, etc. the proper set-up and administration of these controls is required. This may be beyond the resources of some laboratories and is clearly an added cost.

The current operating systems have had a significant impact on the standardization of user interfaces. In the first mass spectral data systems, each had different ways to enter commands or click with a mouse. It was therefore a major challenge to instruct users on how to use a data system. In some cases the operator had to develop significant programming skills to use the system. In current user interfaces many operations, such as cut and paste, are standardized on the computer. As these are performed in the same way in most computer programs, everyone who has worked with a computer is well-versed in the art of using menus and mouse clicks to interact with a computer program. The fact that a large majority of data systems are based on Windows makes this even more true. Chemists now have a much easier time learning

to use new data systems because they already have a good idea of how the user interface will work. This standardization has produced the one drawback, in that many programs now look the same and it is becoming a challenge for programmers to make their systems unique and original.

3.5 Programmability

As discussed above, many aspects of modern mass spectral data systems require that they be programmable (or customizable). Every system is limited to have a finite number of built-in operating modes and algorithms. The chemist, therefore, needs to have the ability to mix modes and tailor algorithms to suit experimental objectives. The programmer who writes the data system is not able to anticipate which aspects of an algorithm the user may wish to vary, so ultimately the user needs to be able to program functions into the system. This section describes the elements that a system must include, to be truly programmable.

First the user needs a language to write algorithms in. The language needs to incorporate basic arithmetic and common math functions. It also needs to have program flow control elements such as loop and logic structures ('if', 'while', and 'repeat'). The user needs to be able to use predefined variables such as 'first_mass', 'last_mass', 'detector_voltage'. They also need to control MS operations with built-in commands such as 'Do_one_scan', 'filament_on', 'filament_off'. The language must have built-in feedback so that decisions can be based on the state of the instrument or the nature of the data. Functions such as 'Source_temperature' or 'Manifold_pressure' can serve this purpose. The most advanced systems include functions such as 'Intensity of mass 278 in the last data-set' or 'Mass of the biggest peak in the last data-set' that return facts about the data.

The language should to be able to perform all control, collection, and analysis steps. It ought to be possible to run more than one process at once, so that the system can collect one set of data while analyzing another, and perhaps reporting on a third. For good laboratory practice, it is important to have functions to write any sort of information into a file. This will ensure that every dataset has enough information stored within it to show exactly how it was acquired. It also allows diagnostic programs to keep track of instrument performance over any period of time.

The feedback functions in the language can be used for a wide variety of algorithms. For example, in the arena of safety, the chemist can specify the actions to be taken if a temperature or pressure gets too high. Alternatively, the chemist could write a tuning loop that sets a voltage, collects a scan of data, and reads back information about a peak.

Section 3.5.1 includes a number of illustrative examples. The procedures are written in a pseudocode quite similar to an actual programming language. The first example shows the optimization of data collection by timing acquisition. The second is part of an autotune algorithm. The third is a higher-level procedure for automatic quantitation, meant to run continuously in the background.

3.5.1 Example 1: Timed Acquisition

One can increase the amount of analytically relevant information by only collecting data that is appropriate for the retention time. The following routine is for an MS/MS instrument that does single reaction monitoring of several different reactions, 219–69 for the first two minutes, 512–69 for the next two minutes and 131–69 thereafter:

```
start_collection
srm(219,69)
while retention_time < 2 : scan : end
srm(512,69)
while retention_time < 4 : scan : end
srm(131,69)
while retention_time < 10 : scan : end
end_collection
```

The functions referred to have the following meaning:

- srm(m1,m2) means set the instrument to monitor the reaction m1–m2.
- scan means collect one scan of data.

3.5.2 Example 2: Tuning

This is an example of a tuning algorithm called 'optimize_lens'; it's one argument specifies which lens to tune. While tuning, the system collect raw data. For these data, 'height' refers to the height of the biggest peak in the dataset. As before, 'scan' means collect one scan of data. The items 'biggest_area' and 'best_lens' are temporary variables. The goal of the procedure is to find an optimum value of a lens.

```
optimize_lens(n)
  biggest_height = 0
  for lens(n) = −100 to 0 in steps of 1
    scan
    if height >= biggest_height
      biggest_height = height
      best_lens = lens(1)
    end
  end
  lens(n) = best_lens
```

When this is done, lens *n* will have been optimized.
Such a routine may be built into a higher level-routine:

```
optimize_all_lenses
  optimize_lens(1)
  optimize_lens(2)
  optimize_lens(3)
  etc . . .
```

This process may be abstracted to as high a level as required.

3.5.3 Example 3: Automatic Quantitation

If the data system is designed properly, rules can be defined to run continuously in the background. Here is an example of a high-level algorithm that provides automatic updating of a quantitation list when the chemist changes the calculations for one of the compounds in the list. For example, suppose the user has collected several data files, including analytes and internal and external standards. They have quantitated a set of compounds in these data files, using mass chromatograms to obtain an area for each unknown or standard. The areas and concentrations of the standards are used to create a response curve. The areas of unknowns are used, in conjunction with the response curve, to calculate the unknown concentrations. One now has a list of areas and quantities for each compound, along with the information on how they were computed. If the user were to change the area of one of the standard compounds by changing the parameters that went into its calculation, we would like to see the amounts of all related peaks change correspondingly. Here is an example of a procedure that performs this operation.

```
Repeat-forever
  if some_compound_changed
    for r = 1 to number_of_response_points
      c = external_standard(r)
      c1 = internal_standard(r)
      response_x(r) = compound_area(c)/
        compound_area(c1)
      response_y(r) = compound_amount(c)/
        compound_amount(c1)
    end
    for c = 1 to number_of_compounds
      compute_amount(c)
    end
  end
  Sleep_one_second
end
```

The functions referred to have the following meanings:

- some_compound_changed — set to 'True' if any one of the compounds in the list changed area or amount, which means that 'compound_area' or 'compound_amount' changed for that compound.
- number_of_response_points — the number of points in the response list.
- external_standard(r) — the compound number of the external standard at position r in the response list.
- internal_standard(r) — the compound number of the internal standard at position r in the response list.
- compound_area(c) — the area under the chromatogram for compound c.
- compound_amount(c) — the calculated or given amount of compound c.
- number_of_compounds — the number of compounds in the list.
- compute_amount(c) — computes the amount of compound c from its area and the response list.
- sleep_one_second — prevents the procedure from hogging the system – there is no need to check more than once a second that the user has changed the data.

This procedure checks whether some compound has changed area or amount (changed by the user). If so, it recalculates the response curve by filling in each point on the response curve from the areas and the amounts of the appropriate compounds. Then, for each compound, it computes the amount of that compound ('compute_amount' uses the response curve). If the display of data is responsive to changes in the data, the user will see all areas and amounts change as soon as one value is changed. In section 3.2 an example was given of the necessity of a close link between data and display; this procedure is another example.

To keep the code simple, this example assumes that there is only one response list involved. However, it is easy to extend the code to a system that includes several response lists.

These examples give an indication of how programmable a data system can be. The challenge for the

designers of data systems is to balance flexibility with simplicity for the sake of the chemist who is content with the basic operation of the system. MS is not a trivial task and operating a mass spectral data system is likely to remain a challenging task as the functionality of MS Data systems continues to expand. Hopefully, the user interface, which is what makes it possible to use all this functionality, will keep pace.

4 BIOLOGICAL APPLICATIONS

4.1 Protein Sequence Determination

MS has long had as a goal the ability to determine the sequence within polymeric biologically important molecules, such as DNA and proteins. There have been notable advances in this area in the period 1990–1999.[19–24] However, the goal of developing a simple yet general method for rapidly sequencing these molecules by MS has remained elusive.

Fortunately, alternative approaches have been introduced that take advantage of the large amount of DNA and RNA sequence information that has been generated by genome sequencing projects and which is currently stored in databases. Using this nucleic acid sequence information, it is possible to determine whether the results of a mass spectrometric experiment correspond to a sequence in a database. If such a correspondence exists, it is no longer necessary to sequence the protein (or corresponding RNA) by MS or other means – the answer can be simply read from the database. If the database information is incomplete, it can serve as a starting point for other studies, greatly reducing the experimental work required for the determination of the full sequence.

4.1.1 Peptide Cleavage and Extraction

All protein sequence identification experiments begin with the creation of a set of smaller oligopeptide molecules from the intact protein. The patterns generated from these oligopeptides are then used to search nucleotide sequence databases. These oligopeptides (frequently referred to simply as 'peptides') are produced by the action of a reagent that cleaves the protein's peptide bond backbone at sequence-specific sites, such as peptide bonds that are adjacent to a limited set of amino acids. Peptide digesting enzymes, such as trypsin or endopeptidase Lys-C, are commonly used for this purpose. Reactive amino acids, particularly cysteine residues, are protected with chemical reagents that prevent them from modification during the process.

4.1.2 Dataset Generation – Mass Spectrometry, Matrix-assisted Laser Desorption/Ionization and Electrospray Ionization

Once the oligopeptides have been generated, the masses of all of the peptides generated from a protein can be measured at once, using matrix-assisted laser desorption/ionization (MALDI) or electrospray ionization (ESI) ion sources, mounted on a variety of different types of mass analyzers. Analysis using a MALDI ion source is currently the most common method, but the use of sophisticated deconvolution will make ESI a viable option. Proteins produce patterns containing 10–1000 isotopic peak clusters, depending on the sequence of a particular protein. Each peak cluster represents a particular peptide sequence.

Alternatively, the ions corresponding to an individual peptide from a protein digestion can be isolated, either using chromatography or MS/MS techniques. The resulting ions can then be fragmented in a gas phase collision cell producing a pattern of masses characteristic of the sequence of the original peptide (MS/MS or MS/MS/MS, i.e. MS^n). This pattern can be used to search databases, using the accumulated knowledge of the preferred gas-phase peptide bond cleavage rate constants. The resulting pattern is strongly affected by the time elapsed between collision and measurement of the product ion distribution, so different rules must be applied for different types of MS/MS analyzers.

4.2 Database Search Strategies

The data sets generated by mass spectrometric experiments can be compared to the nucleotide sequence information present in databases in several ways. All of these methods share some common features. In order to compare sequences, the chemical reactions involved in producing the cleaved peptides are simulated, producing a theoretical set of peptides for each known protein sequence in the database. This simulation can either be done during the search process or a specialized database consisting of the peptides resulting from a particular cleavage and protection chemistry can be prepared in advance. The simulations are then compared to the experimental data, either using specialized correlation functions or using multiple-step discrete pattern matching. This comparison is done by assuming that sequences that correspond to the experimental data set will contain a set of peptides with masses that agree with the experimental data, within some experimental error.

4.3 Nucleotide Databases

Databases of complete gene sequences can be searched as though they were protein sequence databases. The

existence of known start codons and intron/exon assignments allows the use of, either MS or MS^n patterns. Nucleotide databases that contain incomplete sequence information, such as the database of expressed sequence tags (dBEST),[25] present special challenges. In this type of database, there are six possible peptide sequences for each nucleotide sequence and each must be searched independently. The short length of the sequences makes the use of MS-only data impractical; these databases require the use of MS^n fragmentation patterns.

4.3.1 Annotated Protein Databases

Dedicated protein sequence databases that store annotated oligopeptide translations of nucleic acid sequences are the best databases for any MS-related search strategy. The annotations in the database indicate what is known about post-translational modification of the protein, allowing the chemical cleavage simulation to be performed more accurately than is possible using nucleotide information alone. The number of protein sequences in this type of database is still very limited – annotation is time-consuming and only possible when detailed experimental results are available for a particular sequence.

4.3.2 Confirmation and Scoring Results

The results of comparing a set of experimental masses to a sequence database usually results in the identification of a number of candidate sequences that match to some extent with the experimental data. The task of distinguishing random matches from the 'correct' match has been approached in a number of ways. The simplest scoring system involves counting the number of masses that agree within a given error and reporting the sequence with the most matches as being the best candidate sequence. This approach is very simplistic and frequently deceptive. More sophisticated scoring schemes involve appraising pattern matches on the following criteria:

- sequence coverage – the fraction of the candidate protein represented by the experimental masses;
- sequence distribution – the pattern of matched peptides in the candidate protein;
- mass deviation – the pattern of experimental mass deviations from the simulation values;
- statistical significance – the likelihood that the match could have occurred at random.

Research into the appropriate scoring scheme for MS and MS^n match scoring is still ongoing. The most successful of scoring systems will be the basis for the next generation of fully automated protein identification instruments.

Currently, none of the protein identification algorithms make use of AI or algorithm training methods. The Profound algorithm is currently the closest to using AI – it uses a Bayesian statistical approach to evaluating data sets, allowing for the unbiased evaluation of search results and for the detection of multiple sequences in a single MS data set.[26]

4.4 Protein Modification Analysis

MS may have limitations in the determination of protein sequences de novo, but it is very well suited to the detection of chemical modifications of a known sequence. The detection of these modifications is very dependent on good software as there is too much information for manual data reduction. The general strategy is very similar to that used to identify proteins, a process that grew out of the standard practice for finding modifications. The general strategy is as follows: determination of the intact protein molecular mass; cleavage to peptides; generation of mass spectra; and automated, detailed comparison of the MS data set with a known sequence.

4.4.1 Peptide Cleavage and Extraction

The cleavage and protection chemistry available for detection of modifications is much broader than that used in protein identification experiments. Any proteolytic enzyme, chemical cleavage or protection method can be used, depending on the type of modification sought. Popular endoproteinase enzymes are trypsin, endoproteinases Lys-C and Asp-N, and *Staph*. V8 proteinase.[27] Exopeptidases, such as carboxypeptidase A, B, and P can also be useful for generating C-terminal sequencing ladders for smaller peptides.[28] Unlike the protein identification procedure, it is very useful to follow a time course of protein cleavage, as the dynamics of proteolysis can provide valuable clues to the identity and location of modifications. Chemical cleavage reagents, such as cyanogen bromide, iodosobenzoic acid and hydroxylamine, can be used in place of enzymes. These reagents are less popular than enzymes, because of their propensity for producing complicating modifications in the sequence through side-reactions.

4.4.2 Generation of Mass Spectroscopy Datasets

Mass spectroscopy datasets are collected in the same way as for protein identification experiments. Typically, a number of experiments are run, using different cleavage reagents with different and complementary specificity. For example, both a trypsin and endoproteinase Asp-N digest would be both run, taking several time points during the reaction to reconstruct its time course. All of the data collected is stored for later analysis.

Datasets for MS^n can be prepared that greatly assist analysis in the detection of common modifications, such as phosphorylation or disulfide cross-linking. These modifications produce characteristic fragmentation signals following gas-phase collisions. The most popular method for collecting this type of specialized data is directly coupling the output from high-performance liquid chromatography (HPLC) to an MS/MS device (such as a triple quadrupole or a ion trap analyzer) and flagging spectra that contain these characteristic signals.

4.4.3 Comparison with Sequence

Once a dataset has been assembled, it must be compared with the results that would be expected from the predicted amino acid sequence. For a simple enzymatic cleavage experiment on a protein that has 30 theoretical cleavage sites (N) and no cystines, there are approximately 450 possible product peptides. The complexity of the task of examining a dataset for each of the possible products and locating the peaks that do not agree is clearly too time-consuming and error prone to be performed manually.

The majority of data is analyzed using automated systems to assist the investigator – no system that performs a complete and reliable analysis is currently available. Modern analysis is performed by first determining the mass of a peak in the MS dataset and searching a sequence for a peptide with a mass that is within a user-defined error of the experimental value. The dataset can be a single mass spectrum containing all of the cleaved peptides or an HPLC/MS dataset that contains thousands of individual spectra, each of which will contain zero or more of the peptides, depending on the chromatographic conditions.

The best analysis systems use a multifactorial fuzzy-logic-based approach to analyzing the data. The entire dataset is interrogated and individual matches rated with respect to all of the other assignments. Peptides with the same mass (within the allowed error) are assigned based on the kinetics of the cleavage reaction, as inferred by the fuzzy logic rules. Peaks that can be assigned by mass, but which are unlikely based on the entire data set, are flagged for further examination and confirmation. These flagged peaks, as well as those that could not be assigned are then iterated through a selection of known modifications and the complete sequence assignment process repeated. The fuzzy logic assignments depend on the entire data set so the change of value in the simulated experiment requires a complete reexamination for the assignments.

Once this iterative process is finished, the results can be projected back onto the theoretical sequence, with each assignment flagged and color coded so that interesting portions of the sequence are displayed. This process is particularly effective if the three-dimensional structure of the protein is known, where the peptides can be located in a structure diagram shown in a stereoscopic display.

4.5 Use with Differential Displays

Differential displays are a particularly useful tool in current cell biology. They consist of some type of high-resolution protein separation system, such as two-dimensional gel electrophoresis, and a signal detection process such as affinity or silver staining. A cell challenged in various ways will produce displays that differ as the protein complement being expressed in the cell changes. By overlaying displays, spots that change are apparent. The challenge is then to determine what protein has appeared (or disappeared or changed positions).

The techniques described in sections 4.1–4.3 can be applied to these displays. By excising interesting areas of the separation bed and extracting the protein components in various ways, the protein sequence can be rapidly identified. A new generation of automated differential display devices utilizing MS as a protein identification system is currently being designed. These instruments will replace the current practice of manual sample preparation and mass analysis, although the protein identification algorithms will remain the same. The fully automated instruments will probably perform best on data derived from species with known genomes.

4.6 Alternate Splicing

When a eukaryotic organism translates its DNA into RNA in the nucleus (the primary transcript), the transfer RNA is usually edited before it is exported out of the nucleus as transfer RNA for transcription into a peptide chain. This editing process, generally referred to as RNA splicing, involves the removal of portions of the RNA that do not code for peptide sequence (exons), leaving the portions that do code for sequence and transcription regulatory functions (introns). In multicellular organisms with differentiated cell and tissue types – which includes all animals and plants – different exons can be spliced into the transfer-RNA in different cell types, resulting in different protein sequences that originate from the same gene. These different proteins that originate from the same gene are called ‘alternate splices’. The regions of genomic DNA that will be deleted or included can be predicted with some accuracy for the most likely transfer-RNA product; however, the alternate forms cannot be predicted in advance and they must be discovered experimentally.

Protein identification-type experiments are ideally suited to the rapid identification of alternately spliced proteins. In order to distinguish alternate splicing from proteolytic processing, the existing generation of protein recognition algorithms will need to include a method for searching and scoring multiple gaps using the genomic sequence as a starting point. By using predicted exon/intron divisions, it should be possible to search the possible DNA-to-RNA translation sequences to determine whether an alternate splice form is present in a particular differential display. Such a search is beyond the capabilities of the current generation of software: they all require an accurate RNA translation. However, with the introduction of AI-type training capabilities, it should be possible to apply the most sophisticated of the current algorithms to this problem.

5 MASS SPECTROMETRY APPLICATIONS OF PRINCIPAL COMPONENT AND FACTOR ANALYSES

5.1 Introduction

PCA calculates an orthogonal basis (i.e. coordinate system) for sets of mass spectra for which each axis maximizes the variation of the spectral dataset. Each axis is represented as a vector that relates the linear dependence of the mass spectral features (i.e. m/z variables). Typically, the new coordinate system has a reduced dimensionality. The PCA procedure allows the scientist to compress the data, remove noise, and discover the underlying or latent linear structure of the data.

FA rotates the principal components away from directions that maximize variance towards new chemically relevant directions; it allows scientists to resolve underlying pure components in mixtures, build classification models, and determine mass spectral features that relate to specific properties such as concentration or class.

5.2 Selected History

When computers were interfaced with mass spectrometers, numerical calculations could be used to simplify the data. A brief and somewhat selective history follows. The PCA technique was developed for the study of psychological measurements that are inherently complicated by many difficult-to-control factors.[29] These factors can be attributed to the different environmental, behavioral, or genetic influence on the human subjects who are evaluated. Some method was needed that would determine which factors were important and which factors were correlated.

The earliest applications of PCA in analytical chemistry determined the number of underlying components in mixtures. Specifically, for optical measurements, a mixture could be effectively modeled by a linear combination of the spectra of the pure components. The number of pure components of the mixture would correspond to the rank of the data matrix. The rank of a matrix of optical spectra of mixtures was computed using Gaussian elimination.[30,31] The application of FA to solving problems in chemical analysis was pioneered by Malinowski et al.[32,33]

The applications of PCA and FA to gas chromatography (GC) and MS first occurred in the 1970s. Initially, FA was employed to study the relationships between chemical structure and GC retention indices.[34–37] Then PCA was demonstrated as a tool for deconvolving overlapping GC peaks.[38] Next, FA was applied to 22 isomers of alkyl benzenes to assist the interpretation of fragmentation pathways and as a method for compressing the mass spectra to lower dimensionality.[39,40] The FA method was used for classifying mass spectra.[41]

The coupling of multichannel detection, specifically MS to GC, allowed PCA and FA to resolve overlapping components of GC/MS peaks.[42,43] The target transform FA method was automated for GC/MS analysis.[44]

FA was initially applied to solving problems of overlapping peaks in GC/MS. Soon it was realized that FA was a useful tool for the analysis of complex mixtures such as biological (bacteria, proteins, and hair) and geological (coal, atmospheric particles, and kerogen) samples. These complex samples were all amenable to pyrolysis mass spectrometry (PyMS).[45] The discriminant and FA were applied to various biological samples.[46] An unsupervised graphical rotation method was developed and applied to geological samples.[47] Canonical variates analysis (CVA)[48] was used to take advantage of measurement errors furnished by replicate spectra and was combined with rotation for mixtures of glycogen, dextran, and bovine serum albumin,[49] and has become one of the methods of choice for the analysis of MS fingerprints from bacteria.[50] The FA method was demonstrated as an effective tool for analysis of smoke particles by PyMS.[49] A related method that exploits PCA for classification is soft independent modeling for class analogies (SIMCA).[51]

Other techniques that benefited from FA and PCA are laser ionization mass spectrometry (LI/MS),[52] fast atom bombardment mass spectrometry (FAB/MS),[53] electrospray MS,[54] and secondary ion mass spectrometry (SIMS).[55] In the SIMS work, cluster analysis was used to help align high-resolution mass measurements into optimized columns of the data matrix, which was evaluated using PCA.

Table 1 The number of hydrocarbon spectra in the data set with respect to class and carbon number

Hydrocarbon class	Carbon number					Total
	4	5	6	7	10	
Diene	16	40	52	56	33	197
Alkene	12	17	60	61	37	187
Alkane	8	14	28	31	62	143
Total	36	71	140	148	132	527

5.3 Introductory Example

A brief demonstration of PCA and FA is presented with accompanying graphs. A data set of mass spectra was obtained from the Wiley Registry of Mass Spectra, 5th edition,[245] that comprised spectra of hydrocarbons that were alkane, alkene, or diene. This data matrix is exemplary because the MS fragmentation patterns are easy to predict. These data were part of a larger project that built classification models for identifying spectra of plastic recycling products.[56] The data matrix was composed of 527 spectra and 95 columns that correspond to m/z values. The m/z values ranged from 50 to 144. Typically, if all the spectra have no mass peaks at a specified m/z, this column is excluded from the data matrix. Table 1 gives the design of the hydrocarbon data set.

The principal components were calculated by singular value decomposition (SVD)[57] in a Win32 program that was written in C++. The analysis of these data required less than 5 s on a 300 MHz PC computer with 128 MB of random access memory and operating under Windows 98 in console mode.

The spectra were preprocessed by normalizing to unit vector length and centering the spectra about their mean spectrum before the PCA. Figure 1 gives the eigenvalues with respect to the component number. The eigenvalues measure the variance spanned by each eigenvector. For intricate data sets, the eigenvalues typically asymptotically approach zero. The relative variance of each eigenvalue is calculated by dividing the eigenvalue by the total variance of the data matrix. The total variance is easily obtained as the sum of the eigenvalues. From this calculation, the first two principal components account for approximately half the variance in this data set.

Examination of the mass spectral scores on the first two components in Figure 2 shows that the spectra tend to cluster by class (i.e. degree of unsaturation). The first component has the largest range of values and is responsible for separating the spectra in order of diene, alkene, and alkane. This component can be investigated further using the variable loadings in Figure 3. This graph shows the principal component plotted with respect to m/z, so that key spectral features may be investigated.

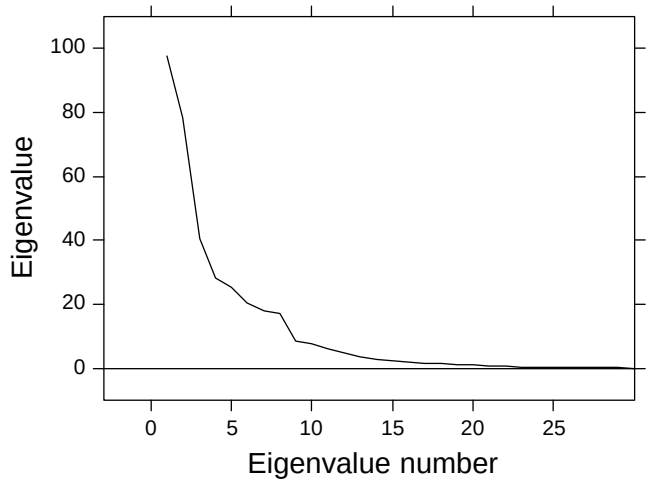

Figure 1 Eigenvalues plotted as function of the number of components for a set of 527 mass spectra with 95 variables.

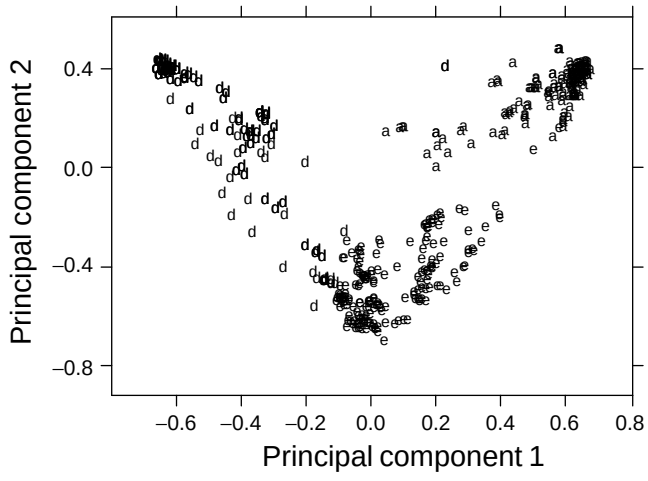

Figure 2 Observation scores of hydrocarbon mass spectra on the first two principal components, 47% of the cumulative variance: a, alkanes; d, dienes; e, alkenes.

The principal components point in mathematically, but not necessarily chemically, relevant directions. Target transform FA was used to rotate 13 principal components that spanned 95% of the variance in directions that correlate with the specific structural classes of the spectra. Figures 4–6 give the rotated factors for the diene, alkene, and alkane classes. Notice that the periodicity of the fragmentation pattern is precisely as one would expect for these sets of data. The alkenes follow a pattern of carbon number times 14, the dienes follow the same pattern except shifted to two less, and the alkanes shifted by two more. The shifts account for the change in mass of the molecule by the loss of two hydrogen atoms for each degree of unsaturation.

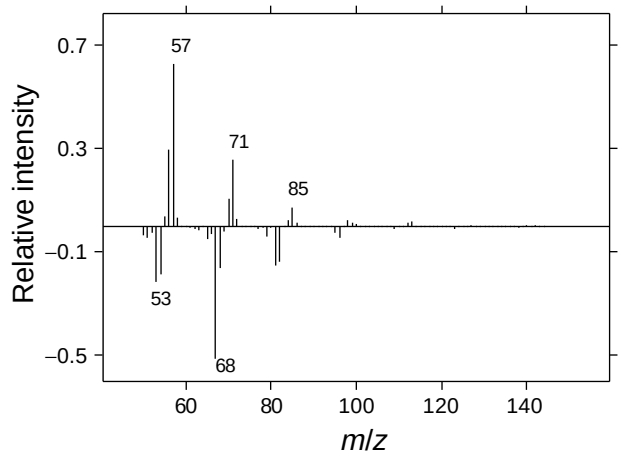

Figure 3 Variable loadings for the first principal component of the mass spectra dataset, 21% of the cumulative variance.

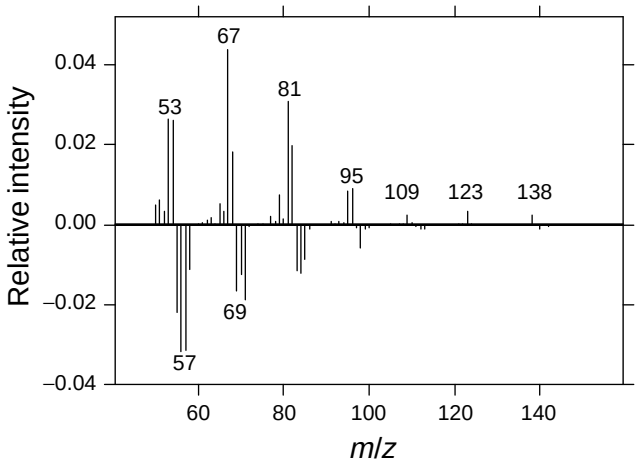

Figure 4 The target-transformed factor for dienes obtained from a set of 13 principal components that spanned 95% of the cumulative variance.

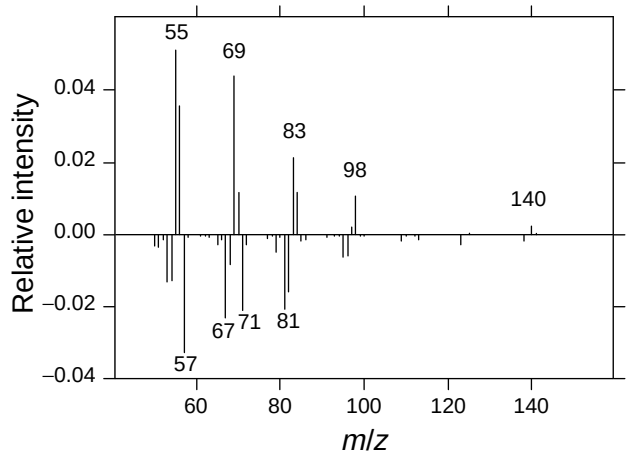

Figure 5 The target-transformed factor for alkenes obtained from a set of 13 principal components that spanned 95% of the cumulative variance.

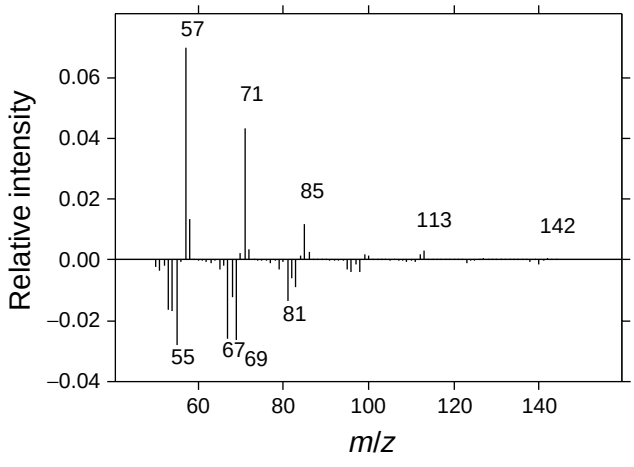

Figure 6 The target-transformed factor for alkanes obtained from a set of 13 principal components that spanned 95% of the cumulative variance.

5.4 Theoretical Basis

5.4.1 Principal Component Analysis

Typically, data are arranged into a matrix format so that each row corresponds to a mass spectrum and each column to a measurement at a specific m/z value. This matrix is designated as **D**. The PCA method is mathematically based on eigenvectors or eigenanalysis.

The method decomposes a set of data into two sets of matrices. The matrices are special in that the columns point in directions of major sources of variation of the data matrix. These vectors are eigenvectors. (*Eigen* is the German word for characteristic.) Because these vectors already point in a direction inherent to the data matrix, they will not change direction when multiplied by the data matrix. This property is referred to as the eigenvector relationship and is defined as Equations (1) and (2):

$$\mathbf{D}^{\mathrm{T}}\mathbf{D}\mathbf{v}_i = \lambda_i \mathbf{v}_i \tag{1}$$

$$\mathbf{D}\mathbf{D}^{\mathrm{T}}\mathbf{u}_i = \lambda_i \mathbf{u}_i \tag{2}$$

where $\mathbf{D}^{\mathrm{T}}\mathbf{D}$ is a square symmetric matrix that characterizes the covariance of the columns of the data set **D**, $\mathbf{v}_i$ is eigenvector i that is in the row-space of **D**, and λ_i is eigenvalue i. In Equation (2), $\mathbf{D}\mathbf{D}^{\mathrm{T}}$ is a square symmetric matrix that characterizes the covariance of the rows of the data set and $\mathbf{u}_i$ is in the column-space of the **D**.

Besides maximizing the variance, the sets of eigenvectors form orthogonal bases. This property can be expressed by Equations (3) and (4),

$$\mathbf{V}^{\mathrm{T}}\mathbf{V} = \mathbf{I} \tag{3}$$

$$\mathbf{U}^{\mathrm{T}}\mathbf{U} = \mathbf{I} \tag{4}$$

for which **V** is a matrix comprising row-space eigenvectors ($\mathbf{v}_i$) and **U** is a matrix comprising column-space eigenvectors ($\mathbf{u}_i$). The identity matrix **I**, comprises values of unity along the diagonal and values of zero for all other matrix elements. The relationship given in Equations (3) and (4) is important because it shows that the transpose of an orthogonal matrix is equal to its inverse.

For large sets of data, computing the covariance matrix is time-consuming. A method that is precise and fast for computing both sets of eigenvectors is SVD:[58]

$$\mathbf{D} = \mathbf{USV}^{\mathrm{T}} \tag{5}$$

From Equation (5) **D** can be decomposed into the two matrices of eigenvectors and a diagonal matrix **S** of singular values (Equation 6):

$$\lambda_i = s_i^2 \tag{6}$$

The singular values w_i are equal to the square root of the eigenvalues, which leads to another important property that is given by Equation (7):

$$\mathbf{D}^n = \mathbf{US}^n\mathbf{V}^{\mathrm{T}} \tag{7}$$

This relationship is important because any power of **D** can be calculated by decomposing the matrix, raising the diagonal matrix of singular values to the *n*th power, and reconstructing the matrix. A useful power is negative unity, because $\mathbf{D}^{-1}$ can be used for calculating calibration models. Furthermore, pseudoinverses can be calculated from singular or ill-conditioned data matrices by reconstructing using only the components (i.e. vectors) that correspond to singular values above a threshold.

The other important element of PCA is the concept of a principal component. Because the row-space and column-space eigenvectors are orthogonal and hence independent, the number of eigenvectors will always be less than the dimensionality (i.e. minimum of the number of rows or columns) of **D**. The number of nonzero eigen or singular values gives the mathematical rank *r* of **D**. The rank gives the number of underlying linear components of a matrix. However, besides mathematical rank, there are physical and chemical ranks. The physical rank gives all the sources of variances that are associated with the physics of obtaining the measurement including noise. These variances may correspond to background or instrumental components. The chemical rank corresponds to variances related to the chemical components of interest. Therefore, the mathematical rank is the number of components with eigenvalues greater than zero. The physical rank corresponds to eigenvalues greater than a threshold that characterizes the indeterminate error of making the measurement. The chemical rank is typically the smallest and corresponds to the number of chemical components, when the variances of the data follow a linear model.

Typically, the components that are less than either the physical or chemical ranks are referred to as principal components. The components that correspond to the smaller eigenvalues are referred to as secondary components. Secondary components usually characterize noise or undesirable variances in the data. The determination of the correct number of principal components *r* is important. If the number of principal components is too small then characteristic variances will be removed from the data. If the number of principal components is too large then noise will be embedded in the components as well as signal. There are several methods to evaluate the calculation of the correct number of principal components.

One of the simplest methods is to reconstruct the data **D** using subsets of the eigenvectors. When the reconstructed data resemble the original data within the precision of the measurement, then the proper number of principal components has been obtained. An empirical approach determines the minimum of the indicator function (IND), which is not well understood, but furnishes reliable estimates of the chemical rank.[59]

There are three key pieces of information furnished by PCA. The first is the relationship of the variance that is spanned by each component. Plotting the eigenvalues as a function of components, gives information regarding the distribution of information in the data. The eigenvalues also convey information regarding the condition number of the data matrix. The condition number is obtained by dividing the largest eigenvalue by the smallest. This condition number can be used to assess the error bounds on a regression model[60] and as a means to evaluate selectivity.[61]

This approach was what made PCA useful for assessing the number of analytical components contained in a GC peak. This methodology is still used; however, it is referred to as window or evolving factor analysis (EFA). Instead of processing the spectra contained in a chromatographic peak, a window (i.e. a user-defined dataset) can be moved along the chromatogram. The chemical rank is evaluated and gives the number of chemical components in the window.

The second piece of information is furnished by the observation scores. Score plots display the distribution of spectra or rows of **D** in a lower dimension graph. The scores of the first two components provide a two-dimensional window that maximizes information content of the spectra. If the rows are ordered with respect to time, the observation scores give trajectories of the changes that occur in the data over time (Equation 8):

$$\mathbf{o}_i = \mathbf{d}_i\mathbf{V} = \mathbf{u}_i\mathbf{S} \tag{8}$$

for which $\mathbf{o}_i$ is a row vector of the *i*th observation score of spectrum *i* ($\mathbf{d}_i$). This may be calculated by multiplying

a spectrum or the *i*th row of **D** by the matrix of principal components. The observation scores can be calculated directly for the results of SVD by multiplying the matrix of singular values **W** by the *i*th row of the column-space eigenvectors **U**. Plots of the observation scores are also referred to as the Karhunen–Loève plots. These plots allow clustering of the data to be visualized.

The final piece of information is yielded by the variable loadings, which indicate the direction that the row-space eigenvectors point. The variable loadings show the importance of each variable for a given principal component. Plots of variable loadings can be examined for characteristic spectral features. They also are used together with the observation score plots to see which spectral features are responsible for separating objects in the score plots.

In some instances, the data matrix **D** can be modified so that the principal components point in directions that are more chemically relevant. These modifications to **D** are referred to collectively as preprocessing. Typically, the spectra are mean-centered, which refers to subtracting the average spectrum from each spectrum in the dataset. This centers the distribution of spectra about the origin. If the data are not mean-centered, the first principal component will span the variance characterized by the overall distance of the data set from the origin.

In some cases, the spectra are normalized so as to remove any variations related to concentrations. Normalization scales the rows of **D**, so that each row is weighted equally. Mathematically, normalizing the spectra to unit vector length will achieve this equalized weighting. For spectra that vary linearly with concentration, the concentration information is manifested in the vector length of the spectrum. Other methods of normalization include normalizing to a constant base peak intensity (i.e. maximum peak of unity) or to a constant integrated intensity (i.e. sum of peaks of unity).

The data may be scaled so that the variables or columns of **D** are weighted equally. Scaling is important for mass spectra, because peaks of higher mass that tend to convey more information, have smaller intensities, and tend to be less influential in the calculation of principal components.

Autoscaling gives each variable or column of data equal weight. This method of scaling is useful when the noise or the signals are not uniformly distributed across the mass range. For this method of preprocessing, each column of **D** is divided by its standard deviation. The problem with autoscaling is that variables that convey noise only are given equal weight with those that convey signal. A better approach is to scale the data by the experimental errors for each variable. Experimental error can be measured as the standard deviation of replicate spectra. The variances of these standard deviations can be added for different samples to calculate an estimate of the experimental error. The experimental error avoids the diminution of the signals during scaling.

An alternative to scaling is transformation. In some cases the data may be converted to the square root or logarithm to give greater weight to smaller features. A useful method for preprocessing mass spectra is through modulo compression.[62]

5.4.2 *Canonical Variates Analysis*

For supervised classification, a useful method related to PCA is CVA,[63] which is also applied with discriminant (function) analysis.[64] The CVA method is not usually performed on the original feature space (mass spectra) because the mass spectra have colinear variables or too many variables for CVA. This problem may be resolved by compressing the data, such as using principal component scores[52] or by calculating the pseudo-inverse of the covariance matrix.[65] The canonical variates (CVs) are principal components that are calculated from a matrix that is related to Fisher variance and analysis of variance. In the traditional method, two covariance matrices are calculated. The first matrix characterizes the covariance of the class means about the grand mean of the data. The second matrix characterizes the variation of the spectra about their class means.

The CVA approach uses PCA twice in the calculation. First, SVD is used to compute the pseudo-inverse of the within-groups sum of squares matrix ($\mathbf{SS}_{\mathrm{w}}^{+}$). The CVs are the variable loadings obtained from PCA applied to **R**, which is obtained by Equation (9):

$$\mathbf{R} = \mathbf{SS}_{\mathrm{b}}\,\mathbf{SS}_{\mathrm{w}}^{+} \tag{9}$$

for which $\mathbf{S}_{\mathrm{b}}$ is the between-class sum of squares matrix and $\mathbf{S}_{\mathrm{w}}^{+}$ is the pseudo-inverse of the within-class sum of squares matrix ($\mathbf{S}_{\mathrm{w}}$). These are calculated by Equations (10) and (11),

$$\mathbf{SS}_{\mathrm{b}} = \sum_{i=1}^{N_{\mathrm{c}}} N_i(\bar{\mathbf{x}}_i - \bar{\mathbf{x}})(\bar{\mathbf{x}}_i - \bar{\mathbf{x}})^{\mathrm{T}} \tag{10}$$

$$\mathbf{SS}_{\mathrm{w}} = \sum_{i=1}^{N_{\mathrm{c}}} \sum_{j=1}^{N_i} (\mathbf{x}_{ji} - \bar{\mathbf{x}}_i)(\mathbf{x}_{ji} - \bar{\mathbf{x}}_i)^{\mathrm{T}} \tag{11}$$

for which N_{c} is the number of classes, N_i is the number of spectra in the *i*th class, $\bar{\mathbf{x}}_i$ is the class mean, and the $\bar{\mathbf{x}}$ is the global mean. The rank of **R** will be equal to the number of classes less one (e.g. $N_{\mathrm{c}} - 1$), because a degree of freedom is lost by centering the spectra about the global mean and the product of two matrices can not exceed the minimum rank of the two multiplier matrices.[66] The CVs are a basis set of orthogonal vectors that maximize the separations of the classes (i.e. maximize the distance among the means

and minimize the distance of the spectra from their class means). Thus the principle of CVA is similar to PCA but, because the objective of CVA is to maximize the ratio of the between-group to within-group variance, a plot of the first two CVs displays the best two-dimensional representation of the class separation.

5.4.3 Factor Analysis

PCA yields variable loadings that are characteristic for the data matrix. The variable loadings are meaningful with respect to maximizing variances. For other applications it is useful to rotate these loadings in other directions that pertain to specific problems. Once the principal components are rotated, the technique is referred to as FA. Rotations are either oblique or orthogonal. The orthogonal rotations maintain the linear independence of the principal components and the basis. Oblique rotations remove the constraint of linear independence and therefore model more closely physical and chemical phenomena. These methods calculate a square matrix $\mathbf{T}$ of coefficients that rotate the components with a dimensionality of r. For which r is the number of principal components. Typically, the column-space components or observation scores are rotated in the forward direction and the row-space components or variable loadings are rotated in the reverse direction using $\mathbf{T}^{-1}$. The rotation matrices can be computed by numerical optimization of a variety of objective functions or they can be rotated graphically until they resemble a target. For orthogonal rotation, the most popular objective function is Varimax.[67] This rotation method seeks to increase the magnitude of the observation scores on a single component and reduce the scores magnitude on all other components.

Target transformation calculates a transformation matrix that rotates the row-space and column-space eigenvectors or components in directions that agree with a target vector. Typically, the targets are a set of properties that may correlate with the objects, and the transformation matrix is calculated by regression. These transformation matrices may be calculated using the eigenvectors from SVD (Equations 12 and 13),

$$\mathbf{T} = \mathbf{U}^{\mathrm{T}}\mathbf{X} \tag{12}$$

$$\widehat{\mathbf{X}} = \mathbf{UT} \tag{13}$$

for which $\mathbf{X}$ is composed of columns of targets, $\mathbf{T}$ is the transformation matrix, and $\widehat{\mathbf{X}}$ is the estimated target matrix. The loadings can be rotated by regressing the matrix of variable loadings $\mathbf{V}$ onto the target matrix $\mathbf{T}$ that has r rows and the number of columns equals the number of target vectors (Equation 14):

$$\widehat{\mathbf{Y}} = \mathbf{VT}(\mathbf{T}^{\mathrm{T}}\mathbf{T})^{-1} \tag{14}$$

In some cases, it is advised to use the pseudo-inverse of $\mathbf{T}$, because the inner product of $\mathbf{T}$ may be ill-conditioned or singular. The factor variable loadings for the targets are estimated by $\widehat{\mathbf{Y}}$.

5.5 Related Methods and Future Applications

5.5.1 Calibration

There are various methods to exploit the properties of eigenvectors to accomplish calibration. Calibration furnishes models that predict properties from data such as mass spectra. The most common use for calibration is to construct models that estimate the concentration of components in complex samples by their mass spectra.

Principal component regression (PCR) uses the observation scores for computing the regression model. The advantage of this approach is that for MS data in many cases $\mathbf{D}$ is underdetermined (i.e. more m/z measurements than spectra). Because the observation scores will equal the chemical rank, the number of variables are reduced and regression by inverse least squares becomes possible.

A related method uses SVD to calculate the pseudoinverse $\mathbf{D}^{+}$. The SVD regression is computationally more efficient than PCR, but is mathematically equivalent. A very effective method for many problems is partial least squares (PLS). This calculates common column-space eigenvectors between the independent block (i.e. $\mathbf{D}$) and dependent block (i.e. $\mathbf{Y}$) of data. The PLS method was initially developed in the field of econometrics. Both PLS and PCA are described in a tutorial;[68] PLS has been enhanced to handle multiway or higher-order data.[69]

Quantitative analysis of complex binary and tertiary biochemical mixtures analyzed with PyMS[70] showed that, of the latent variable PCR and PLS methods, the best technique was PLS, a finding to be found generally by other studies.[71,72]

5.5.2 Multivariate Curve Resolution

The same FA methods that were initially applied to peaks of GC/MS data have evolved so that they can be applied to the entire chromatographic runs. These methods start with a set of principal components. The components are rotated by a method known as alternating least squares (ALS). The key is to apply mathematical constraints such as non-negativity (no negative peaks) and unimodality (a spectrum will appear in only one peak of a chromatogram).

Curve resolution provides a means to enhance the spatial or depth resolution of ion measurements of surfaces or could be exploited to examine changes in electrospray mass spectra as a function of changing solvent conditions. Curve resolution will continue to exploit PCA and FA

to detect impure chromatographic peaks and mathematically resolve the overlapping components.

EFA and window factor analysis (WFA) use the eigenvalues to model the change in concentrations of components in the data matrices. The eigenvalues can be combined to form initial concentration profiles that are regressed onto the data. The concentration profiles and extracted spectra are refined using ALS with constraints.

5.5.3 Multiway Analysis

The entire chromatographic mass spectral data matrix **D** is only the beginning. If several chromatographic runs are used to characterize a chemical process or if multidimensional MS matrices of data are collected, a tensor or cube of data would be obtained. Using methods based on the Tucker model,[73] the higher-order sets of data can be decomposed into vectors or planes of principal components. A method related to the Tucker model is PARAFAC.

5.6 Reviews and Tutorials

Malinowski's monograph is an excellent resource for PCA and FA.[74] Tutorials on FA and related methods can be found in the literature – the philosophical basis of PCA and FA,[75] EFA,[76] and target transform FA.[77] Multivariate curve resolution applied to chromatography with multichannel detection has been published as a tutorial[78] and reviewed specifically for GC/MS.[79] Tutorials of the multiway PCA method PARAFAC[80] and PLS[68] are also useful entry points into these methods. The text by Martens and Næs on multivariate calibration thoroughly describes PLS.[81]

5.7 Acknowledgments

Tricia Buxton, Guoxiang Chen, and Aaron Urbas are thanked for their help with preparing this section. Thomas Isenhour and Kent Voorhees are thanked for their help with searching the literature. The introductory example data set was initially prepared by Peter Tandler.

6 ARTIFICIAL NEURAL NETWORKS

6.1 Summary

The availability of powerful desktop computers in conjunction with the development of several user-friendly packages that can simulate ANNs has led to the increase in adoption of these 'intelligent' systems by the analytical scientist for pattern recognition. The nature, properties and exploitation of ANNs with particular reference to MS is reviewed.

6.2 Introduction to Multivariate Data

Multivariate data consist of the results of observations of many different characters (variables) for a number of individuals (objects).[82,83] Each variable may be regarded as constituting a different dimension, such that if there are n variables each object may be said to reside at a unique position in an abstract entity, referred to as n-dimensional hyperspace. In the case of MS, these variables are represented by the intensities of particular mass ions. This hyperspace is necessarily difficult to visualize, and the underlying theme of multivariate analysis (MVA) is thus simplification[84] or dimensionality reduction, which usually means that we want to summarize a large body of data by means of relatively few parameters, preferably the two or three that lend themselves to graphical display, with minimal loss of information.

6.3 Supervised Versus Unsupervised Learning

Conventionally the reduction of the multivariate data generated by MS[85–87] has normally been carried out using PCA;[84,88–90] the PCA technique is well-known for reducing the dimensionality of multivariate data while preserving most of the variance, and the principal component scores can easily be plotted and clusters in the data visualized.

Analyses of this type fall into the category of unsupervised learning (Figure 7a), in which the relevant multivariate algorithms seek clusters in the data.[90] Provided that the data set contains standards – of known origin and relevant to the analyses – it is evident that one can establish the closeness of any unknown samples to a standard, and thus effect the identification of the former. This technique is termed 'operational fingerprinting' by Meuzelaar et al.[91]

Such methods, although in some sense quantitative, are better seen as qualitative as their chief purpose is merely to distinguish objects or populations. More recently, a variety of related but much more powerful methods, which are most often referred to within the framework of chemometrics, have been applied to supervised analysis of multivariate data (Figure 7b). In these methods, one seeks to relate the multivariate MS inputs to the concentrations of target determinands, i.e. to generate a quantitative analysis, essentially via suitable types of multidimensional curve fitting or linear regression analysis.[83,92–96] Although nonlinear versions of these techniques are increasingly available,[97–103] the usual implementations of these methods are linear in scope. However, a related approach to chemometrics, which is inherently nonlinear, is the use of ANNs.

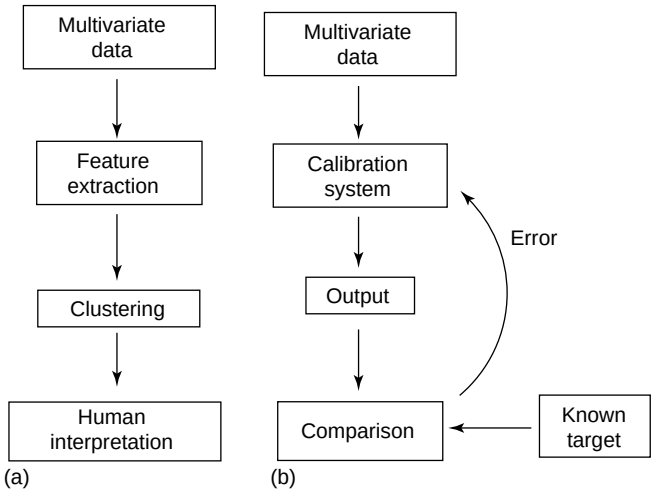

Figure 7 (a) Unsupervised learning – when learning is unsupervised, the system is shown a set of inputs (multivariate MS data) and then left to cluster them into groups. For MVA this optimization procedure is usually simplification or dimensionality reduction; this means that a large body of data (the inputs) are summarized by means of a few parameters with minimal loss of information. After clustering the results then have to be interpreted. (b) Supervised learning – when the desired responses (targets) associated with each of the inputs (multivariate data) are known then the system may be supervised. The goal of supervised learning is to find a model that will correctly associate the inputs with the targets; this is usually achieved by minimizing the error between the known target and the model's response (output).

6.4 Biological Inspiration

ANNs are biologically inspired; they are composed of processing units that act in a manner that is analogous to the basic function of the biological neuron (Figure 8). In essence, the functionality of the biological neuron consists of receiving signals, or stimuli, from other cells at their synapses, processing this information, and deciding (usually on a threshold basis) whether or not to produce a response, that is passed onto other cells. In ANNs these neurons are replaced with very simple computational units which can take a numerical input and transform it (usually via summation) into an output. These processing units are then organized in a way that models the organization of the biological neural network, the brain.

Despite the rather superficial resemblance between the ANN and biological neural network, ANNs do exhibit a surprising number of the brain's characteristics. For example, they learn from experience, generalize from previous examples to new ones, abstract essential characteristics from inputs containing irrelevant data, and make errors (although this is usually because of badly chosen training data.[83,104–109]) All these traits are considered more characteristic of human thought than of serial processing by computers. These systems offer the

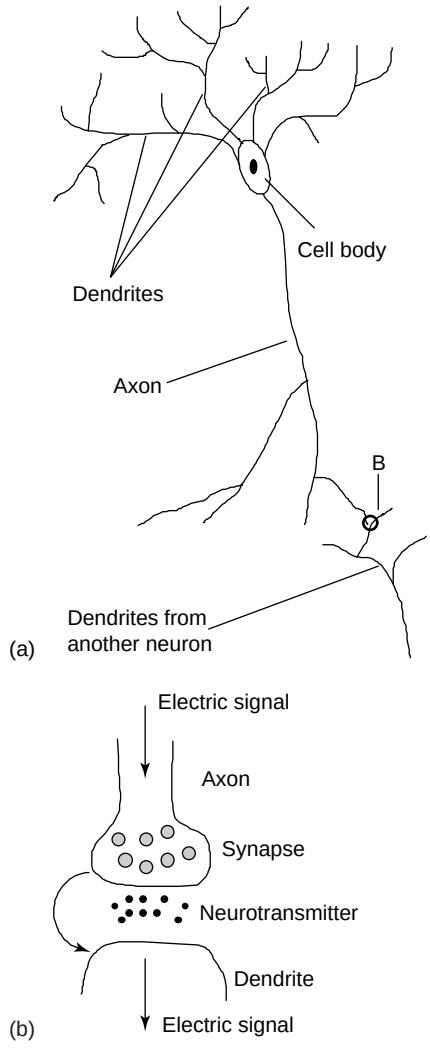

Figure 8 (a) A very simplified scheme of a biological neuron. Dendrites are filaments that extend from the cell body to other neurons where they receive signals at connection points called synapses, these dendrites then provide inputs to the cell body, the axon sends output signals. (b) The synapse is a special connection which can be strengthened or weakened to allow more, or less, signal through.

mass spectrometrist the capability of performing pattern recognition on very complex uninterpretable (at least to the naked eye) multivariate data.

For a MS analytical system, there are some mass spectra that have desired responses, which are known (i.e. the concentration of target determinands). These two types of data (the representation of the objects and their responses in the system) form pairs, which are usually called inputs and targets. The goal of supervised learning is to find a model or mapping that will correctly associate the inputs with the targets (Figure 7).

Thus the basic idea in these supervised learning neural network techniques is that there are minimally four

datasets to be studied, as follows. The training data consist of (a) a matrix of s rows and n columns in which s is the number of objects and n the number of variables (these are the normalized ion intensities at a particular mass-to-charge ratio for MS), and (b) a second matrix of target data, again consisting of s rows and typically 1 or two columns, in which the columns represent the variable(s) whose value(s) it is desired to know (these are the result(s) wanted) and which for the training set have actually been determined by some existing, benchmark method. This variable may be the concentration of a target determinand, and is always paired with the patterns in the same row in (a). The test data also consist of two matrices, (c) and (d), corresponding to those in (a) and (b) above, but the test set contains different objects. As the name suggests, this second pair is used to test the accuracy of the system; alternatively they may be used to cross-validate the model. That is to say, after construction of the model using the training set (a, b) the test data (c) (these may be new spectra) are then passed through the calibration model so as to obtain the model's prediction of results. These may then be compared with the known, expected responses (d).

6.5 Data Selection

As in all other data analysis techniques, these supervised learning methods are not immune from sensitivity to badly chosen initial data,[106] and so good modelling practice must be adopted.[109] Therefore, the exemplars for the training set must be carefully chosen; the adage is 'garbage in – garbage out'. It is known[83,104,105,107–109] that neural networks (and other supervised learning methods such as PLS) can over-fit data. For example, an over-trained neural network has usually learnt perfectly the stimulus patterns it has seen but cannot give accurate predictions for unseen stimuli, i.e. it is no longer able to generalize. For supervised learning methods accurately to learn and predict the concentrations of determinands in biological systems, or to identify new observations as being from something previously seen, the model must obviously be calibrated to the correct point. The reality is that in extension to normal chemometric practices detailed above the data should be split into three sets: (a) data used to calibrate the model; (b) data employed to cross-validate the model; and (c) spectra whose determinand concentration, or identities, were unknown and used to test the calibrated system. During calibration, the models would be interrogated with both the training and the cross-validation set and the error between the output seen and that expected calculated, thus allowing two calibration curves for the training and cross-validation sets to be drawn. When the error on the cross-validation data was lowest, the system will be deemed to have reached the best generalization point and then may be challenged with input stimuli whose determinand concentrations, or identities, are really unknown.

An alternative approach is to determine an acceptable error for the model, and construct ANN models that fit within this error. For many classification problems a relative root mean square error of calibration (RRMSEC) of 10% is usually sufficient (Equation 15):

$$\mathrm{RRMSEC} = \sqrt{\frac{\sum_{i=1}^{s}\sum_{j=1}^{p}(y_{ij} - \hat{y}_{ij})^2}{\sum_{i=1}^{s}\sum_{j=1}^{p}(y_{ij} - \bar{y}_j)^2}} \qquad (15)$$

for which RRMSEC is the relative root mean square standard error of calibration, $\hat{y}_{ij}$ is the neural network output of unit j and training object $\boldsymbol{I}$, y_{ij} is the corresponding target value. There are p outputs for the neural network model and s training objects about the class mean $\hat{y}_i$.

Many neural network models are often overly optimized and do not generalize well, even though monitoring sets or cross-validation are used. The caveat is that the prediction and training sets must be well designed and representative of the specific problem. With poorly designed training and prediction sets the neural networks will model hidden experimental factors that correlate with the target properties. A second problem occurs when the prediction set is used to configure the network – if the prediction set matches the training set too well the ANN model will overtrain, and if the prediction set is too dissimilar the network will undertrain.

Latin-partitioning is a useful experimental design tool for evaluating neural network models, the method constructs training and prediction set pairs for which each target object in the data is present once and only once in the prediction sets.[110] The method randomly partitions the data so that each target pattern is represented in the sets with equal proportionality. This method is important because the composition of the prediction set is a major source of variation for evaluating neural network models and by including all objects in the prediction set it will not be biased.

For quantitative determinations it is also imperative that the objects fill the sample space. If a neural network is trained with samples in the concentration range from 0% to 50% it is unlikely to give accurate estimates for samples whose concentrations are greater than 50%; that is to say, the network is unable to extrapolate.[109] Furthermore for the network to provide good interpolation it needs to be

trained with a number of samples covering the desired concentration range.[111]

6.5.1 Sensitivity Analysis of Artificial Neural Network Models

Sensitivity analysis may be used to probe ANN models and can lead to an understanding as to why they do not predict or generalize well. Furthermore, interpreting the sensitivities may lead to an understanding of causal relationships between input and output variables. Kowalski and Faber proposed methods on the quantitative measurement of sensitivities.[112] Ebhart et al.[113] proposed three comparable methods to calculate the mean square sensitivity, absolute value average sensitivity, and maximum sensitivity. Similarly, Howes and Crook[114] studied the three types of input influence, namely general influence, specific influence, and potential influence, on the network output. Most of these studies considered the effects of weight matrix in multilayer perceptrons (MLPs) models. Choi and Choi[115] defined the sensitivity of input perturbations as the ratio of output error to the standard deviation of each input perturbation, which involves complex weight calculations. Kovalishyn et al.[116] have proposed several sensitivity measurements to be used with cascade-correlation networks (CCNs) for variable selection. The sensitivity was measured by connection weights, or the second derivative of the error function with respect to the neuron weight. It was shown in this paper, the sensitivities measured from their definition were not stable with the dynamic growth of the network, and also sensitive to addition of noise. In Ikonomopoulos' study of importance of input variables with the wavelet adaptive neural network, the sensitivity of input variables was estimated by the ratio of the standard deviations of the prediction and the altered input.[117] It was found that with this method, sensitivity measurements were highly correlated with the input perturbation. Sung derived the sensitivity coefficient matrix for a backpropagation (BP) neural network with two hidden layers.[118] Other sensitivity analysis methods based on the input magnitude and functional measurements have also been proposed.[119,120]

Because neural network models are fundamentally nonlinear, the sensitivity will depend on the input values from which they are calculated. Harrington et al.[121] proposed using partial derivatives of the neural network output with respect to the input. They compare the sensitivity of the average input for each class with the average sensitivity of each class. This method provides a method for detecting input variables that are modeled by higher-ordered functions in the neural network model, and provides a quantitative measure of the input variables contribution for each target output. Weight vectors were not directly involved in the sensitivity measurement.

6.6 Cluster Analyses with Artificial Neural Networks

These analyses fall into the category of unsupervised learning, in which the relevant multivariate algorithms seek clusters in the data.[90] Recently there has been an interest in the use of neural computation methods, which can perform exploratory data analyses on multivariate data, the most commonly used are feature or self-organizing maps (SOMs) and auto-associative artificial neural networks (AAANNs).

6.6.1 Self-organizing Maps

These provide an objective way of classifying data through self-organizing networks of artificial neurons.[122–124] These neural networks are also referred to as Kohonen ANNs, after their inventor.[125] The SOM algorithm is very powerful and is now extensively used for data mining, representation of multidimensional data and the analysis of relationships between variables.[126]

The SOMs used to analyze mass spectra typically consist of a two-dimensional network of neurons arranged on a rectangular grid;[127–129] although a variety of output arrays (Figure 9) and hence neighborhoods are possible.

Consider the situation where a square two-dimensional Kohonen output layer is used (Figure 9b). Each neuron

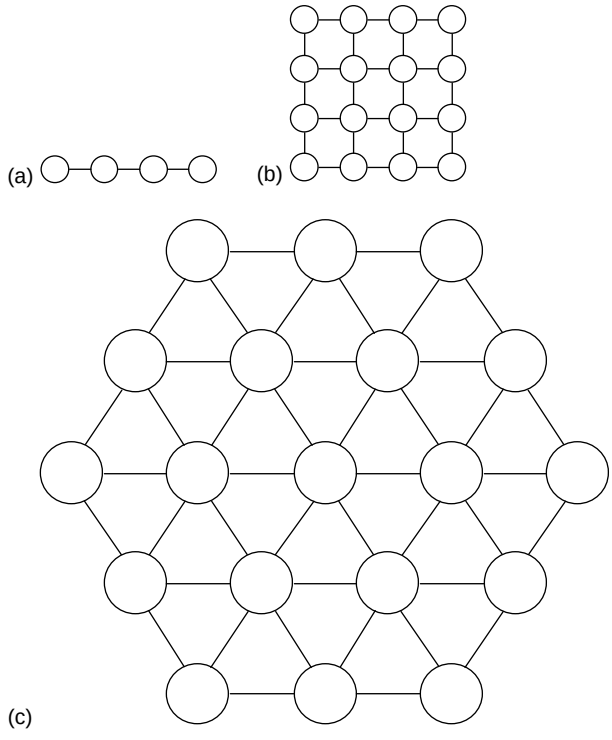

Figure 9 Commonly used SOM structures: (a) one-dimensional array, (b) two-dimensional rectangular network, (c) two-dimensional hexagonal network. The lines represent the neighborhoods.

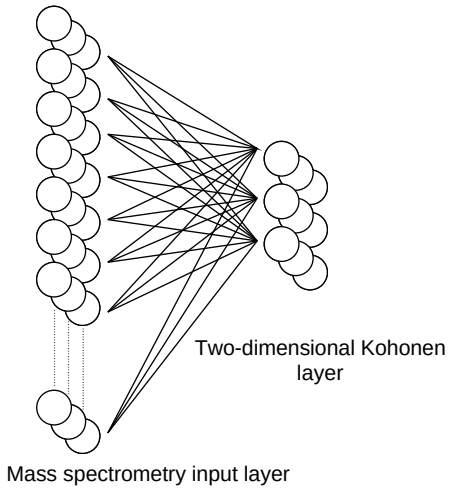

Figure 10 A simplified SOM. Nodes in the two-dimensional Kohonen layer are interconnected with each other (only a few are shown in this figure), such that an activation node also tends to activate surrounding nodes (Figure 3). The mass spectral data are applied to the input layer which activates a node or group of neighboring nodes in the Kohonen layer (represented here as having 3×3 nodes).

is connected to its eight nearest neighbors on the grid. The neurons store a set of weights (a weight vector), each of which corresponds to one of the inputs in the data. (Figure 10 shows a diagrammatic representation of a SOM). Thus, for mass spectral data consisting of 150 quantitative ion intensity measurements at particular m/z charges, each node stores 150 weights in its weight vector. Upon presentation of a mass spectrum (represented as a vector consisting of the 150 ion counts) to the network each neuron calculates its activation level. A node's activation level is defined as Equation (16):

$$\sqrt{\sum_{i=0}^{n}(\text{weight}_i - \text{input}_i)^2} \tag{16}$$

This is simply the Euclidean distance between the points represented by the weight vector and the input vector in n-dimensional space. Thus a node whose weight vector closely matches the input vector will have a small activation level, and a node whose weight vector is very different from the input vector will have a large activation level. The node in the network with the smallest activation level is deemed to be the winner for the current input vector.

During the training process the network is presented with each input pattern in turn, and all the nodes calculate their activation levels as described above. The winning node and some of the nodes around it are then allowed to adjust their weight vectors to match the current input vector more closely. The nodes included in the set, which are allowed to adjust their weights are said to belong to the neighborhood of the winner. The size of the winner's neighborhood is varied throughout the training process. Initially all of the nodes in the network are included in the neighborhood of the winner but, as training proceeds, the size of the neighborhood is decreased linearly after each presentation of the complete training set (all the mass spectra being analyzed), until it includes only the winner itself. The amount by which the nodes in the neighborhood are allowed to adjust their weights is also reduced linearly throughout the training period.

The factor, which governs the size of the weight alterations is known as the learning rate and is represented by α. The iterative adjustments to each item in the weight vector (where δw is the change in the weight) are made in accordance with Equation (17):

$$\delta w_i = -\alpha(w_i - i_i) \tag{17}$$

This is carried out for $i = 1$ to $i = n$, where in this case $n = 150$. The initial value for α is 1 and the final value is 0.

The effect of the learning rule (weight update algorithm) is to distribute the neurons evenly throughout the region of n-dimensional space populated by the training set.[122,123,125] This effect is displayed in Figure 11, which shows the distribution of a square network over an evenly populated two-dimensional square input space. The neuron with the weight vector closest to a given input pattern will win for that pattern and for any other input patterns that it is closest to. Input patterns that allow the same node to win are then deemed to be in the same group, and when a map of their relationship is drawn a line encloses them. By training with networks of increasing size, a map with several levels of groups or contours can be drawn. However, these contours may sometimes cross which

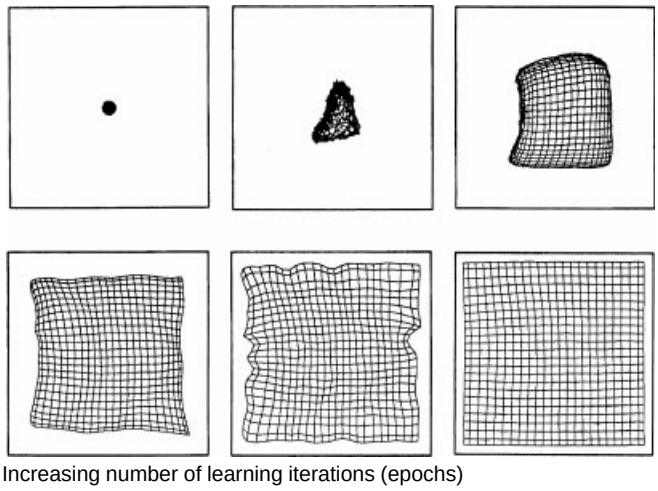

Figure 11 Representations of square networks distributed across an evenly distributed square input space.

appears to be due to failure of the SOM to converge to an even distribution of neurons over the input space.[130] Construction of these maps allows close examination of the relationships between the items in a training set.

A relatively recent paper by Belic and Gyergyek[131] compared several neural network methodologies for the recognition of MS data, based on simulated mass spectra samples and concluded that SOMs could be recommended for practical use in MS recognition. Somewhat earlier, however, Lohninger and Stancl[132] first showed that SOMs were better than *k*-nearest neighbor clustering for the discrimination of (real) mass spectra of steroids from eight distinct classes of chemical compounds. Goodacre et al. have also exploited SOMs successfully to carry out unsupervised learning on pyrolysis mass spectra, and hence the classification of canine *Propionibacterium acnes* isolates,[127] *P. acnes* isolated from man,[133] and plant seeds.[128]

6.6.2 Auto-associative Artificial Neural Networks

AAANNs are a neural network-based method again for unsupervised feature extraction and were pioneered by Kramer.[134,135] They consist of five layers containing processing nodes (neurons or units) made up of a layer of *x* input nodes, *x* output nodes (exactly the same as used in the input layer), and three hidden layers containing (in the example shown in Figure 12) 7, 3 and 7 nodes respectively; this may be represented as an *x*-7-3-7-*x* architecture. Adjacent layers of the network are fully interconnected, and the algorithm used to train these neural networks is the standard BP.[136–138] As these neural networks are auto-associative in nature, that is to say, during training the input and output layer are presented with identical multivariate data, a key feature of these networks is the data compression in the middle (third) bottle-neck layer of three nodes. The second and fourth layers each consist of nodes that map and demap the multivariate data, allowing feature extraction in the bottle-neck layer. Moreover this is equivalent to nonlinear PCA.[134,135] After training, each of the multivariate data used to train the AAANN is applied in turn to the input layer and the overall activation on the three nodes in the bottle-neck layer calculated. Plots of the activations of the nodes in the bottle-neck layer therefore allow clusters to be found in the data.

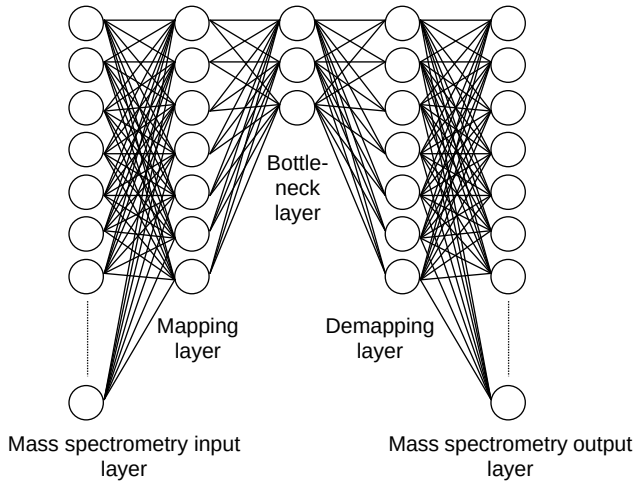

Figure 12 Architecture of an AAANN consisting of five layers. In the architecture shown, adjacent layers of the network are fully interconnected. The input and output layers are presented with identical mass spectral data. A key feature of the auto-associative network is the data compression in the middle (third) bottle-neck layer of three nodes. The second and fourth layers each consist of seven nodes and these map and demap the mass spectra allowing feature extraction in the bottle neck layer.

With reference to MS these AAANNs have only been applied to PyMS, to effect exploratory cluster analyses for the classification of plant seeds[128] and for the authentication of animal cell lines.[139] In the latter study this method of nonlinear PCA was particularly useful because the clusters observed were comparable with the groups obtained from the more conventional statistical approaches of hierarchical cluster analysis. This approach could detect the contamination of cell lines with low numbers of bacteria and fungi, and may plausibly be extended for the rapid detection of mycoplasma infection in animal cell lines.

Elsewhere, within spectroscopy, AAANNs have been used to reduce the dimensionality of the infrared (IR) spectra of polysaccharides and hence extract spectral features due to polysaccharides,[140] to detect plasmid instability using on-line measurements from an industrial fermentation producing a recombinant protein expressed by *Escherichia coli*,[141] and for knowledge extraction in chemical process control.[142] An optimal associative memory (OAM) was developed for removing backgrounds from mid-IR spectra.[143,144] The memory stores reference spectra and generates a best-fit reference spectrum when a sample IR scan is input. This method was extended to a fuzzy optimal associative memory (FOAM) by implementing fuzzy logic to near-IR spectra and was applied to calibration models of glucose in bovine plasma.[145]

6.7 Supervised Analysis with Artificial Neural Networks

As discussed above when the desired responses (targets) associated with each of the inputs (spectra) are known, the system is referred to as supervised. ANNs are very powerful at finding a mapping that will correctly associate mass spectra with known targets; these targets may be the identity of something, or be the quantitative amount of a substance. The two most exploited of the neural computational methods for these purposes are MLPs,

	Encoding in output layer				
	1	2	3	4	5
i	1	0	0	0	0
ii	0	1	0	0	0
iii	0	0	1	0	0
iv	0	0	0	1	0
v	0	0	0	0	1

Figure 13 Binary encoding the five nodes in the output layer on a MLP or radial basis function trained to classify one of five substances i–v.

using standard BP of error, and radial basis function neural networks (RBFs).

In MLPs and RBFs that are to be trained for identification purposes the training data used to calibrate the model (as detailed above) consist of (a) a matrix of s rows and n columns in which s is the number of objects and n the number of variables, and (b) a second matrix, again consisting of s rows and the same number of columns as there are classes to be identified. For identification these s rows are binary encoded as shown in Figure 13; these are the result(s) wanted and for which the training set have actually been determined by classical identification methods, and are always paired with the patterns in the same row in (a). Once trained, new input data can be passed through these ANNs, and the identities read off easily because a tabular format is employed in the classification encoding. Alternatively, for quantification purposes the output node (or nodes) would encode the amount of the substance(s) (in a mixture) that had been analyzed.

The following texts and books are recommended introductory texts to ANNs.[104,106,108,123,136,137,146–154] The following section briefly describes the salient features of both MLPs and RBFs.

6.7.1 Multilayer Perceptrons

The structure of a typical MLP is shown in Figure 14(a) and consists of three layers: MS data as the input layer, connected to an output layer encoded for identification or quantification purposes, via a single hidden layer. Each of the input nodes are connected to the nodes of the hidden layer using abstract interconnections (connections or synapses). These connections each have an associated real value, termed the weight (w_i), that scales the input (i_i) passing through them, this also includes the bias (ϑ), which also has a modifiable weight. Nodes sum the signals feeding to them (Net; Equation 18):

$$\begin{aligned}\text{Net} &= i_1w_1 + i_2w_2 + i_3w_3 + \cdots + i_iw_i + \cdots + i_nw_n \\ &= \sum_{i=1}^{n} i_iw_i + \vartheta \end{aligned} \tag{18}$$

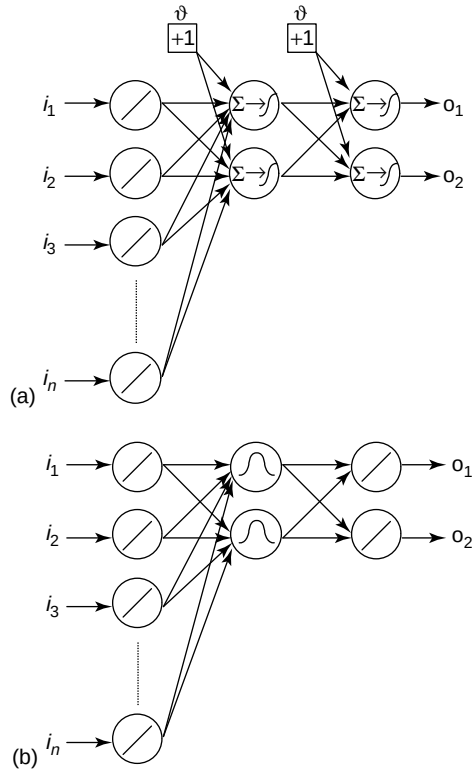

Figure 14 (a) A MLP neural network consisting of an input layer connected to two nodes in the output layer by a hidden layer (also containing two nodes). In the architecture shown, adjacent layers of the network are fully interconnected although other architectures are possible. Nodes in the hidden and output layers consist of processing elements which sum the input applied to the node and scale the signal using a sigmoidal logistic squashing function; also shown in squares is the bias. (b) RBFs consisting of an input layer connected to two nodes in the output layer by a hidden layer (also containing two nodes). The hidden layer consists of radially symmetric Gaussian functions.

The sum of the scaled inputs and the node's bias, are then scaled to lie between 0 and +1 by an activation function to give the nodes output (Out); this scaling is typically achieved using a sigmoidal (logistic) squashing function (Equation 19):

$$\text{Out} = \frac{1}{(1 + \exp^{-\text{Net}})} \tag{19}$$

These signals (Out) are then passed to the output nodes which sums them and in turn squashes this summation by the above sigmoidal activation function; the product of this node is then fed to the outside world.

For the training of the MLP the algorithm used most often is standard BP.[104,136–138,155] Briefly when the input is applied to the network, it is allowed to run until an output is produced at each output node. The differences between the actual and the desired output, taken over the entire training set are fed back through the network in

the reverse direction to signal flow (hence BP) modifying the weights as they go. This process is repeated until a suitable level of error is achieved.

One reason that MLPs are so attractive for the analysis of multivariate (spectral) data is that it has been shown mathematically[156–161] that an MLP neural network consisting of only one hidden layer, with an arbitrarily large number of nodes, can learn any, arbitrary (and hence nonlinear) mapping of a continuous function to an arbitrary degree of accuracy.

Counter-propagation neural networks use a Kohonen hidden layer that is coupled to a Grossberg output layer. These networks are hybrid in that the output layer trains by supervised delta learning and the input is unsupervised. These networks were the early precursors to the radial basis function networks. Harrington and Pack[162] modified the counterpropagation training algorithm so that both hidden and output layers were concomitantly optimized.

Training MLP networks is a very inefficient process, because all processing units are adjusted simultaneously. In addition, the number of hidden units and layers must be configured before training. The CCN developed by Fahlman and Lebiere[163] overcomes these limitations. They add hidden units as needed to reduce the calibration error. The CCN only adjusts a single processing unit or neuron at a time, and therefore trains faster than BP networks.

The problem with the perceptron model is that typically if the spectra form clusters in the data space, there are an infinite set of weight vectors orientations that will furnish zero calibration errors regardless of the dimensionality of the input data. A solution to this problem is to constrain the perceptron model. Temperature constraints that originated with the minimal neural network (MNN),[164] were implemented in BP neural networks with a single global temperature,[165] and local temperature constraints for individual perceptrons in the CCNs.[166] The temperature relates to the thermodynamic temperature employed in other methods such as simulated annealing (reference), and controls the magnitude of the weight vector length (i.e. **w** in Equation 3). The networks are trained so that the magnitude of first derivative of the objective function (e.g. error in BP and covariance in CCN) is maximized with respect to temperature. This constraint ensures that the derivative of the weight vector is large, to facilitate the training rate and the output of the perceptron remains continuous, which improves the reproducibility and the generalization capability of the networks.

6.7.2 Radial Basis Functions

By contrast, RBFs are hybrid neural networks encompassing both unsupervised and supervised learning.[108,167–173] They are also typically three-layer neural networks and, in essence, the sigmoidal squashing function is replaced by nonlinear (often either Gaussian or Mexican hat) basis functions or kernels (Figure 14b). The kernel is the function that determines the output of each node in the hidden layer when an input pattern is applied to it. This output is simply a function of the Euclidean distance from the kernel centre to the presented input pattern in the multidimensional space, and each node in the hidden layer only produces an output when the input applied is within its receptive field; if the input is beyond this receptive field the output is 0. This receptive field can be chosen and is radially symmetric around the kernel centre. Between them the receptive fields cover the entire region of the input space in which a multivariate input pattern may occur; a diagrammatic representation of this is given in Figure 15, where a two-dimensional input is mapped by eight radially symmetric basis functions. This is a fundamentally different approach from the MLP, in which each hidden node represents a nonlinear hyperplanar decision boundary bisecting the input space (Figure 15a). Thus RBFs have the advantage over gradient descent MLPs in that they have the ability to learn any arbitrary nonlinear mapping of a discontinuous function to an arbitrary degree of accuracy.[108,155,167]

The outputs of the RBF nodes in the hidden layer are then fed forward via weighted connections to the nodes in the output layer in a similar fashion to the MLP, and each output node calculates a weighted sum of the outputs from the nonlinear transfer from the kernels in the hidden layer. The only difference is that the output nodes of an RBF are normally linear, whilst those of the MLP more typically employ a sigmoidal or logistic (nonlinear) squashing function.

Thus in the RBF training proceeds in two stages. Stage 1 involves unsupervised clustering of the input data, typically using the K-means clustering algorithm[90,172,174] to divide the high-dimensional input data into clusters. Next, kernel centers are placed at the mean of each cluster of data points. The use of K-means is particularly convenient because it positions the kernels relative to the density of the input data points. Next the receptive field is determined by the nearest neighbor heuristic where r_j (the radius of kernel j) is set to the Euclidean distance between w_j (the vector determining the centre for the jth RBF) and its nearest neighbor (k), and an overlap constant (Overlap) is used (Equation 20):

$$r_j = \text{Overlap} \times \min(\|w_j - w_k\|) \tag{20}$$

where $\|\cdots\|$ denotes a vector norm, or Euclidean distance. The overlap that often gives best results is where the edge of the radius of one kernel is at the centre of its nearest neighbor.[170]

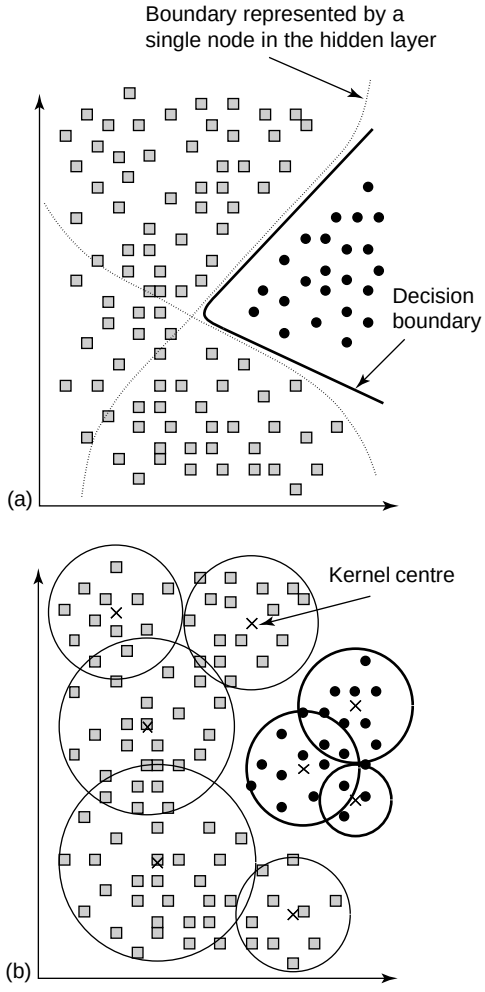

Figure 15 (a) Typical decision boundary for a classification problem created between two data classes by a MLP with two nodes in the hidden layer, for two input nodes. Each hidden node represents a nonlinear hyperplanar boundary and the node(s) in the output layer interpolate this to form a decision boundary. (b) The same classification problem modeled by eight radially symmetric basis functions. The width of each kernel function (referred to as its receptive field) is determined by the local density distribution of training examples.

The output from nodes in the hidden layer is dependent on the shape of the basis function and the one used was that of the Gaussian. Thus this value (R_j) for node j when given the ith input vector (i_i) can be calculated by (Equation 21):

$$R_j(i_i) = e^{-(i_i^2/r_j^2)} \tag{21}$$

Stage 2 involves supervised learning using simple linear regression. The inputs are the output values for all n basis functions ($R_1 - R_n$) for all the training input patterns to that layer ($i_1 - i_n$), and the outputs are identities binary encoded as shown in Figure 13. More recently, Walczak and Massart have used PLS as the linear regression method.[101] Wan and Harrington used SVD regression and reported a self-configuring RBF network that optimizes the number of kernel functions.[175]

6.8 Applications of Artificial Neural Networks to Pyrolysis Mass Spectrometry

PyMS involves the thermal degradation of nonvolatile complex molecules (such as bacteria) in a vacuum causing their cleavage to smaller, volatile fragments, separable by a mass spectrometer on the basis of their m/z.[91] The PyMS method allows the (bio-)chemically-based discrimination of microbial cells (and other organic material) and produces complex biochemical fingerprints (i.e. pyrolysis mass spectra) which are distinct for different bacteria. It is the automation of the instrumentation and ease of use that has lead to the widespread exploitation of PyMS as a taxonomic tool for whole-organism fingerprinting.[86,176] The analytically useful multivariate data (Figure 16) are typically constituted by a set of 150 normalized intensities versus m/z in the range 51 to 200,

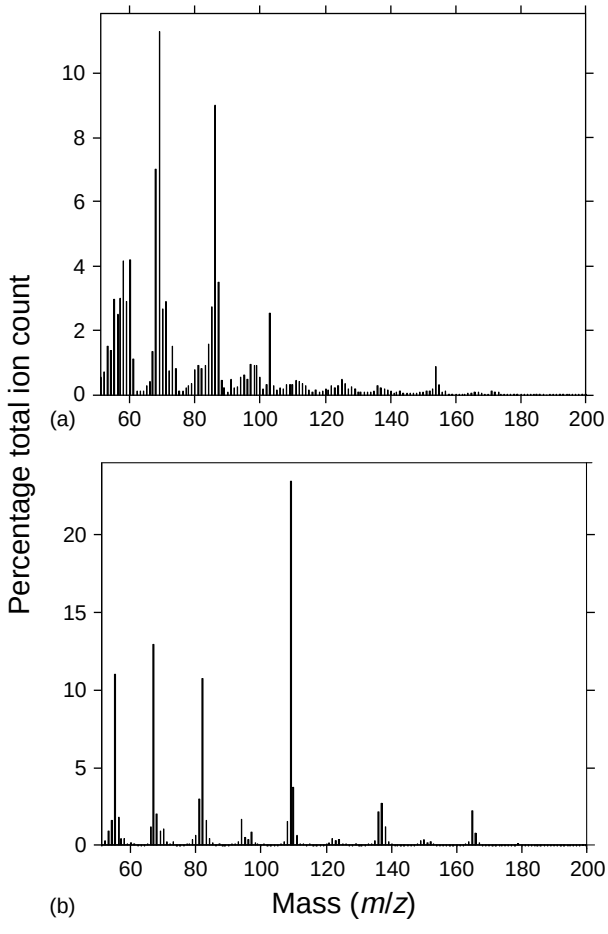

Figure 16 Pyrolysis mass spectra: (a) the bacterium *Bacillus cereus* DSM 31; (b) the simple (bio)chemical caffeine.

and these are applied to the nodes on the input layers of ANNs.

The first demonstration of the ability of ANNs to discriminate between biological samples using their pyrolysis mass spectra was for the qualitative assessment of the adulteration of extra virgin olive oils with various seed oils;[177,178] in this study, which was performed double-blind, neural networks were trained with the spectra from 12 virgin olive oils, coded '1' at the output node, and with the spectra from 12 adulterated oils, which were coded '0'. All oils in the test were correctly identified; in a typical run, the virgins were assessed with a code of 0.99976 ± 0.000146 (range $0.99954 - 1.00016$) and the adulterated olive oils in the test set with a code of 0.001079 ± 0.002838 (range $0.00026 - 0.01009$). This permitted their rapid and precise assessment, a task which previously was labour intensive and very difficult. It was most significant that the traditional unsupervised MVA of PCA, DFA and HCA failed to separate the oils according to their virginity or otherwise, but rather discriminated them on the basis of their cultivar (that is to say, the biggest difference in the mass spectra was due to the type of olive tree that the fruit came from, rather than the adulterant).

The use of PyMS with MLPs for the analysis of foodstuffs is becoming widespread[179] and has been investigated for identifying the geographical origin of olive oils,[180] for the characterization of cocoa butters[181,182] and milk,[183,184] classification of pork backfat[185] and European wines,[186] for differentiating between industrially made vinegar 'Aceto Balsamico di Modena' and traditionally produced vinegar 'Aceto Balsamico Tradizionale di Modena e di Reggio Emilia',[187] for detecting the adulteration of orange juice,[188] and for detecting caffeine in coffee, tea and cola drinks.[189]

Several studies have also shown that this combination of PyMS and MLPs are also very effective for the rapid identification of a variety of bacterial strains of industrial, clinical and veterinary importance.[87,190] For example, this approach has allowed the propionibacteria isolated from dogs to be correctly identified as human *Propionibacterium acnes*,[127] for detecting *Escherichia coli* isolates which produced verocytotoxins,[191] for distinguishing between *Mycobacterium tuberculosis* and *M. bovis*,[192] and for identifying streptomycetes recovered from soil,[193–195] oral abscess bacteria,[196] and fungi belonging to the genus *Penicillium* which were associated with cheese.[197] An example of the highly discriminatory nature of MLPs is that one can even use them to differentiate between methicillin-susceptible and methicillin-resistant *Staphylococcus aureus*;[198] the relevant difference is an alteration in a single penicillin-binding protein.[199,200] Similarly, MLPs can be used to discriminate the physiological difference between sporulated and vegetative *Bacillus* species,[201] and differentiating the verocytotoxin production status in *Escherichia coli*.[191]

RBF have been rather less widely applied to the analysis of mass spectral data. Kang et al.[202,203] have exploited RBFs to detect physiological changes in industrial fermentations of *Streptomyces* species, whereas Goodacre et al.[204] have used RBFs for the identification of common infectious agents associated with urinary tract infection from their MS, IR, and Raman spectra.

An analytical expression for the derivatives of the entropy function for MNNs was derived.[205] This expression was implemented for classifying pyrolysis MS/MS data and the results were compared to those obtained from a BP/ANN.[206] The MNNs differ from BP/ANNs in that they use localized processing and build classification trees with branches composed of multiple processing units. A global entropy minimization may be achieved at a branch by combining the processing logic using principles from fuzzy set theory. Weight vectors are adjusted using an angular coordinate system and gradients of the fuzzy entropy function. The branches are optimal with respect to fuzziness and can accommodate nonlinearly separable or ill-conditioned data. The most significant advantage of the MNNs is that relations among the training data and the mechanism of inference may be directly observed; thus rule-based classification trees have been constructed from the mass spectral daughter ions to discriminate between diesel smoke, dry yeast, *Escherichia coli*, MS-2 coliphage, grass pollen, *Bacillus subtilis*, fog oil, wood smoke, aldolase and *Bacillus globigii*.[206]

6.8.1 *Classification and Qualitative Analysis of Mass Spectra*

ANNs may be used to construct classification models from mass spectra. Once the classification models are built they may be used to rapidly screen large collections of mass spectra. The earliest application of perceptron models was applied to substructure recognition from mass spectra.[207] MLPs were first employed for recognizing functional groups from large collections of mass spectra by Curry and Rumelhart.[208] Werther et al.[209] demonstrated that classifiers based on RBF were better at recognizing functional groups than soft independent models for class analogies (SIMCA[210]), K-nearest neighbors,[211] and linear discriminant analysis[212] from mass spectra.[209] The combination of separation stages to mass spectrometers, such as chromatographic and electrophoretic, furnishes large collections of mass spectra. A fuzzy rule-building expert system (FuRES) was applied to screening GC/MS data of plastic recycling products.[213] The ES was capable

of classifying each mass spectral scan of the chromatographic run by degree of unsaturation (i.e. alkane, alkene, and diene) and furnished a separate chromatogram for each of the three classes. A FuRES has also been used to classify pyrolysis mass spectra[214] and IR spectra.[215]

Temperature-constrained cascade correlation networks (TCCCNs) were applied to the screening of GC/MS pesticide data. Substructures and toxicity ANN models were built for organophosphorus pesticides and applied to screening GC/MS data.[216] The TCCCN was applied to recognizing substructures of carbamates pesticides, and reported the Latin-partition method for evaluating ANN models.[110]

6.8.2 *Quantitative Analysis with Artificial Neural Networks*

All the above studies have been classification problems but perhaps the most significant application of ANNs to the analysis of MS data is to gain accurate and precise quantitative information about the chemical constituents of microbial samples. For example, it has been shown that it is possible using this method to measure the concentrations of binary and tertiary mixtures of cells of the bacteria *Bacillus subtilis*, *Escherichia coli*, and *Staphylococcus aureus*.[107,149,217] With regard to biotechnology, the combination of PyMS and ANNs can be exploited to quantify the amount of mammalian cytochrome b_5 expressed in *Escherichia coli*,[218] and to measure the level of metabolites in fermentor broths.[87,219] In related studies *Penicillium chrysogenum* fermentation broths were analyzed quantitatively for penicillins using PyMS and ANNs,[220] and this approach has also been used to monitor *Gibberella fujikuroi* fermentations producing gibberellic acid[221] and quantify the level of clavulanic acid produced by *Streptomyces clavuligerus*,[203] and to quantify the expression of the heterologous protein α2-interferon in *Escherichia coli*.[222] These and related chemometric approaches have been extended to work with a variety of high dimensional spectroscopic methods,[223] including those based on IR,[222,224] Raman,[221,225] dielectrics,[226] and flow cytometric measurements.[227]

6.8.3 *Instrument Reproducibility*

For MS to be used for the routine identification of microorganisms new (spectral) fingerprints must be able to be compared to those previously collected. However, the major problem with most analytical instruments is that long-term reproducibility is poor and interlaboratory reproducibility abysmal, and so the biochemical or genetic fingerprints of the same material analyzed at two different times are different. Because of the uncertainties over the long-term reproducibility of the PyMS system (defined as over 30 days), PyMS has really been limited within clinical microbiology to the typing of short-term outbreaks where all micro-organisms are analyzed in a single batch.[86,228]

After tuning the instrument, to correct for drift one would need to analyze the same standards at the two different times and use some sort of mathematical correction method. This could simply be subtracting the amount of drift from new spectra collected; however, this assumes that the drift is uniform (linear) with time, which is obviously not the case. This method also relies on the variables (characters) being void of noise, which is also not the case. An alternative method would be to transform the spectra to look like the spectra of the same material previously collected using a method that was (a) robust to noisy data and (b) able to perform nonlinear mappings. ANNs carry out nonlinear mappings, while being able to map the linearities, and are purported to be robust to noisy data. These mathematical methods are therefore ideally suited to be exploited for the correction of mass spectral drift.

Goodacre and Kell[229,230] have found that neural networks can be used successfully to correct for instrumental drift; identical materials were analyzed by PyMS at dates from 4 to 20 months apart, but neural network models produced at earlier times could not be used to give accurate estimates of determinand concentrations or bacterial identities. Calibration samples common to the two datasets were run at the two times, and ANNs set-up in which the inputs were the 150 new calibration masses and the outputs were the 150 calibration masses from the old spectra. Such associative nets could thus be used as signal-processing elements to effect the transformation of data acquired one day to those that would have been acquired on a later date. A further study[231] has shown that one can also affect calibration transfer between laboratories using this approach. These results show clearly that for the first time PyMS can be used to acquire spectra which could be compared to those previously collected and held in a database. It should seem obvious that this approach is not limited solely to PyMS but is generally applicable to any analytical tool which is prone to instrumental drift (which cannot be compensated for by tuning).

6.9 Concluding Remarks

Within MS the move from a stare-and-compare approach to the analysis of highly dimensional multivariate data necessitates the use of chemometrics; particularly when quantitative information is sought from biological systems. The application of ANNs for quantitative and qualitative analyses is becoming more accepted within MS, especially because these neural computational methods clearly present themselves as extremely powerful and valuable tools for the analysis of complex data.

7 OPTIMIZATION TECHNIQUES IN MASS SPECTROMETRY

7.1 Introduction

Parameter optimization is often required in MS. It can be employed in the design of the instrument, tuning of conditions during operation, or calibration of mass scales in data analysis. Good sources of information on general optimization techniques are readily available.[232–235] Therefore, a specific example is provided, illustrating how optimization is advancing MS capability. This example is the application of simplex optimization to mass calibration in TOF instruments.

The simplex method is one of the most popular of several mathematical techniques for optimizing multivariate systems. Developed in 1947, it is composed of successive tests and variation of independent parameters until the system is determined to be unbounded or optimized.[234] Whereas the original method used a graphical representation, current methods rely on high-speed computing. The advances in computers allow the determination of optimal conditions in complicated systems involving a large number of independent parameters.

7.2 Time-of-flight Mass Spectroscopy Mass Calibration

Because of its simplicity and unlimited mass range, TOF instrumentation is particularly well-suited for the analysis of MALDI and electrospray-generated macromolecular ions. Recent advances in TOF technology have facilitated the attainment of high resolution in MALDI/TOF experiments,[236–239] but the method's utility critically depends on its ability to measure ion masses accurately. An ion's flight time can be expressed as an exact function of its mass if information about its formation time, location, and initial velocity is available. Most often, this relationship is expressed to some level of approximation. To zeroth order, the TOF is given by Equation (22):

$$\text{TOF} = k \times \text{mass}^{1/2} \tag{22}$$

This is accurate only when ions are formed with zero initial velocity. Non-zero initial velocities introduce significant complications in the relationship between TOF and mass.

It is currently popular to express TOF as a multiterm polynomial function of mass.[240–242] In this expansion the $\text{mass}^{1/2}$ term is a dominant contributor, but additional terms are needed to produce accurate results. An infinite series is needed to reach arbitrary accuracy. In the polynomial curve-fitting approach, mass spectra of a variety of known calibrant samples are recorded, ion flight times are measured and the coefficients in the polynomial relating TOF to mass are derived using a least-squares minimization routine. When other samples of known mass are introduced into the instrument and their masses are derived using the polynomial relationship, mass accuracies in the parts per million realm have been demonstrated.[243] Nevertheless, to achieve high-quality results with this method, it is necessary that the masses to be measured are near those of calibrants and it is best if they are bracketed by the latter. Attempts to extrapolate the mass calibration over a range that extends beyond that of the calibrants generally leads to poor results.

As an alternative to fitting TOF to an arbitrary polynomial function of mass, it is possible to use elementary physics to calculate the flight times of ions based on instrumental voltages, distances and other operating parameters. Besides not using arbitrary parameters, this approach incorporates the correct physical relationship between TOF and mass, and thus it should extrapolate more accurately into mass regions that are far from the calibrants. Although the merits of this approach are easy to envisage, the stumbling block to its use may be equivalently obvious. For ions initiating their trajectories in a TOF instrument with nonzero velocities, being accelerated in more than one field, drifting in a field-free region and, possibly being postaccelerated or decelerated, the relationship between TOF and mass contains a number of parameters whose exact values are not known well enough to provide the basis for an accurate mass calibration equation. For example, suppose that a singly charged lysozyme ion having a mass of 14 306 Da is accelerated to 20 keV total energy and drifts through a 1.00 m long field-free flight tube. The overall flight time is calculated to be 67 147.2 ns. An error in the high voltage of 10 V leads to a flight time shift of 15 ns. Likewise, an error in the drift tube length of 0.3 mm corresponds to a time shift of 18 ns. These numbers are equivalent to mass errors of 6.5 and 7.8 Da or 460 and 550 parts per million, respectively. Fortunately, accurate values for these imprecisely known instrument parameters can be derived through a simplex optimization procedure. This leads to both a mass calibration equation and a computationally accurate description of the instrument.

7.2.1 Use of the Simplex Algorithm

The minimization used in the simplex algorithm involves computing residual errors between an array of experimental and calculated flight times. The algorithm reiteratively optimizes the instrument parameters in order to minimize the difference between experimental flight times and those calculated using values of the instrument parameters. Any residual error function may be used in the minimization routine. For example, the sum of the squares of differences between experimental and calculated flight times has been used. Two of the input parameters needed by the Simplex algorithm are the characteristic length vector and the fit tolerance. Vector components are typically

matched to measurement uncertainties. The fit tolerance represents the desired conditions for termination of the optimization and is based on expected error between experimental and optimized calculations. Too small a value increases the iterative requirements of the calculation; too large a number causes the simplex navigation to terminate before a minimum is found.

The Simplex calculation constrains ion behavior to physically meaningful values as it is based on exact electrostatic equations. The significance of this can best be demonstrated through direct comparison with the curve-fitting approach to mass calibration. Theoretical flight times for 101 ions having masses between 100 and 1100 Da are initially computed using the exact electrostatic TOF expression. When the resulting flight times are fitted to an optimized three- or five-term polynomial function of mass, the residual errors are rather small, as displayed by the curves in Figure 17. Note that the scale of this graph is rather expanded. However, when the Simplex algorithm is employed for the same purpose, the calibration is seen to be noticeably improved. Although curve-fitting with an *n*-term polynomial exactly matches the theoretical data at the *n* points, discrepancies can be noted between these points. Furthermore the exact electrostatic calculations and the fitted polynomial diverge significantly at masses on the low and high ends of the calibration range. The quality of results obtained by curve-fitting does vary

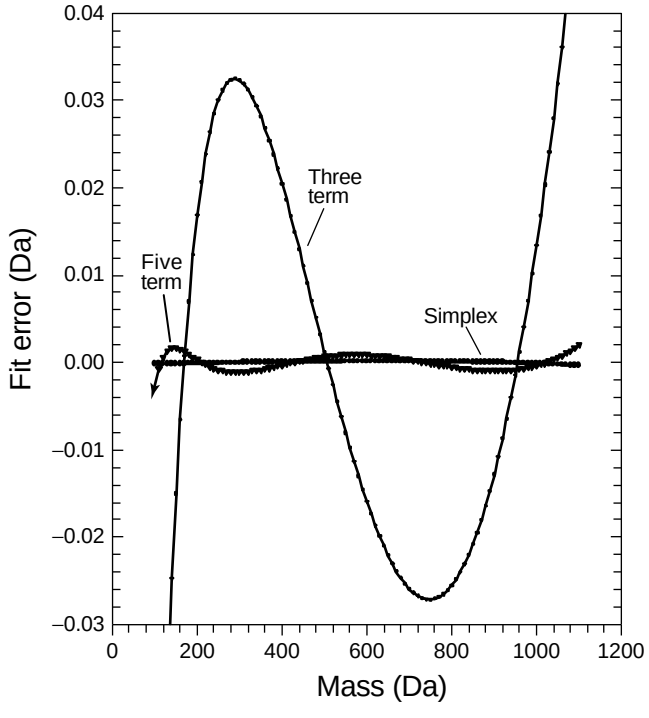

Figure 17 Comparison of polynomial curve fitting and simplex optimization mass calibration approaches when applied to theoretical data.

depending on the polynomial function form and in some cases can match that attained by Simplex. However, the general trends just noted are still observed.

An alternative way to compare the capabilities of Simplex-based mass calibration with the curve-fitting approach is to examine experimental data. A particularly interesting case involves a sample containing a mixture of alkylthioate-coated gold nanoparticles. The challenging aspect of this calibration problem is that gold nanoparticles are laser-desorbed and ionized without being incorporated into a matrix.[244] Consequently, the resulting ions have a velocity distribution that is not characteristic of MALDI ions. In fact, gold nanoparticle ions have lower desorption velocities than their MALDI counterparts. Unfortunately, in the 10–20 kDa mass range, the most familiar mass calibration standards are MALDI-generated protein ions. Thus, an inherent incompatibility between sample and calibrants exists. To deal with this, the parameters chosen for simplex optimization must be carefully chosen. In general, the best ones to optimize are those that are subject to the largest measurement error. Extraction voltage, the ion drawout pulse delay, and the length of the flight tube are all obvious choices. Optimization of this collection of parameters normally yields an accurate mass calibration for a mixture of peptides for which a typical initial ion velocity is $600\,m\,s^{-1}$. However, for the gold sample, the Simplex optimization would not converge when ions were assigned this initial velocity. If the initial gold nanoparticle ion velocity is changed to $100\,m\,s^{-1}$, the algorithm does converge, leading to RMS mass errors for about 30 ion peaks of 0.24 Da. This corresponds to 17 ppm error, which is respectable considering the velocity differences between MALDI-generated protein ions and gold ions. It is noteworthy that in this example, the Simplex algorithm provided more than just a means to perform mass calibration. It also yielded information about the average velocity with which the nanoparticles desorbed.

7.2.2 Mass Calibration Extrapolation

As noted above, calibration of most mass spectra is performed by surrounding the peaks of interest with a good set of known standards. It may seem unreasonable to expect a calibration method to remain accurate in mass regions extrapolated beyond the range of the calibrants. However, this is one of the virtues of the simplex approach. Six peaks in the gold nanoparticle mass spectrum were used to calibrate the spectrum. As displayed in Figure 18, a five-term polynomial curve fit established an acceptable relationship between masses and flight times for ions in the 13–14 kDa range. The simplex approach performed comparably well within this range. However, at masses below 13 kDa or above 14 kDa,

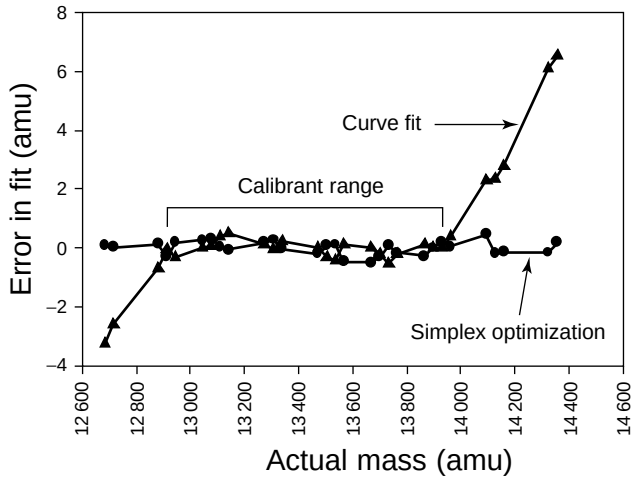

Figure 18 Comparison of polynomial curve fitting and simplex optimization mass calibration approaches when applied to experimentally measured gold nanoparticle mass spectra.

the high-order polynomial curve clearly changes slope and errors relative to the exact TOF calibration and the more realistic Simplex curve increase dramatically.

7.2.3 Conclusions

In summary, a simplex approach to calibrate MALDI/TOF mass spectra appears to be both robust and easily executed. It yields results that are comparable to those obtained with polynomial curve fitting for routine applications, and it excels under more difficult situations such as when samples and calibrants are fundamentally different, when calibrant peaks are incorrectly assigned, or when the mass range of interest must be extended beyond that of the calibrants. It should therefore be of utility in a wide variety of applications.

ABBREVIATIONS AND ACRONYMS

AAANN	Auto-associative Artificial Neural Network
ADC	Analog-to-digital Converter
AI	Artificial Intelligence
ALS	Alternating Least Squares
ANN	Artificial Neural Network
BP	Backpropagation
CCN	Cascade-correlation Network
CVA	Canonical Variates Analysis
CVs	Canonical Variates
dBEST	Database of Expressed Sequence Tags
DC	Direct Current
EFA	Evolving Factor Analysis
ES	Expert System
ESI	Electrospray Ionization
FA	Factor Analysis
FAB/MS	Fast Atom Bombardment Mass Spectrometry
FOAM	Fuzzy Optimal Associative Memory
FTMS	Fourier Transform Mass Spectrometry
FuRES	Fuzzy Rule-building Expert System
GC	Gas Chromatography
GC/MS	Gas Chromatography/Mass Spectrometry
HPLC	High-performance Liquid Chromatography
ICR	Ion Cyclotron Resonance
IND	Indicator Function
IR	Infrared
LI/MS	Laser Ionization Mass Spectrometry
MALDI	Matrix-assisted Laser Desorption/Ionization
MLP	Multilayer Perceptron
MNN	Minimal Neural Network
MS	Mass Spectrometry
MS/MS	Tandem Mass Spectrometry
MVA	Multivariate Analysis
OAM	Optimal Associative Memory
PCA	Principal Component Analysis
PCR	Principal Component Regression
PFTBA	Perfluorotributylamine
PLS	Partial Least Squares
PyMS	Pyrolysis Mass Spectrometry
RBF	Radial Basis Function Neural Networks
RF	Radiofrequency
RRMSEC	Relative Root Mean Square Error of Calibration
SIMCA	Soft Independent Modeling for Class Analogies
SIMS	Secondary Ion Mass Spectrometry
SOM	Self-organizing Map
SVD	Singular Value Decomposition
TCCCN	Temperature-constrained Cascade Correlation Network
TOF	Time-of-flight
WFA	Window Factor Analysis

RELATED ARTICLES

Biomolecules Analysis **(Volume 1)**
Mass Spectrometry in Structural Biology

Chemical Weapons Chemicals Analysis **(Volume 2)**
Gas Chromatography/Mass Spectrometry in Analysis of Chemicals Related to the Chemical Weapons Convention

Clinical Chemistry **(Volume 2)**
Gas Chromatography and Mass Spectrometry in Clinical Chemistry

Forensic Science **(Volume 5)**
Mass Spectrometry for Forensic Applications • Pyrolysis Gas Chromatography in Forensic Science

Nucleic Acids Structure and Mapping **(Volume 6)**
Mass Spectrometry of Nucleic Acids

Peptides and Proteins **(Volume 7)**
Capillary Electrophoresis/Mass Spectrometry in Peptide and Protein Analysis • Matrix-assisted Laser Desorption/Ionization Mass Spectrometry in Peptide and Protein Analysis

Pesticides **(Volume 7)**
Gas Chromatography/Mass Spectrometry Methods in Pesticide Analysis

Petroleum and Liquid Fossil Fuels Analysis **(Volume 8)**
Mass Spectrometry, Low-resolution Electron Impact, for the Rapid Analysis of Petroleum Matrices

Pharmaceuticals and Drugs **(Volume 8)**
Mass Spectrometry in Pharmaceutical Analysis

Polymers and Rubbers **(Volume 9)**
Pyrolysis Techniques in the Analysis of Polymers and Rubbers

Process Instrumental Methods **(Volume 9)**
Mass Spectrometry in Process Analysis

Pulp and Paper **(Volume 10)**
Pyrolysis in the Pulp and Paper Industry

Mass Spectrometry **(Volume 13)**
Mass Spectrometry: Overview and History • Electron Ionization Mass Spectrometry • Literature of Mass Spectrometry • Time-of-flight Mass Spectrometry

REFERENCES

1. P.H. Winston, *Artificial Intelligence*, 2nd edition, Addison-Wesley, London, 1984.
2. E. Rich, *Artificial Intelligence*, McGraw-Hill, New York, 1983.
3. Application Report 002, 'Townsend Discharge Nitric Oxide CI GC/MS for Hydrocarbon Analysis of the Middle Distillates', Bear Instruments, Santa Clara, California, 1999.
4. S.E. Stein, 'An Integrated Method for Spectrum Extraction and Compound Identification from Gas Chromatography/Mass Spectrometry Data', *J. Am. Soc. Mass Spectrom.*, **10**(8), 770–781 (1999).
5. P. Ausloos, C.L. Clifton, S.G. Lias, A.I. Mikaya, S.E. Stein, D.V. Tchekhovskoi, O.D. Sparkman, V. Zaikin, Damo Zhu, 'The Critical Evaluation of a Comprehensive Mass Spectral Library', *J. Am. Soc. Mass Spectrom.*, **199**(10), 287–299 (1999).
6. J.R. Chapman, *Computers in Mass Spectrometry*, Academic Press, London, 1978.
7. H.E. Duckworth, R.C. Barber, V.S. Venkatasubramanian, *Mass Spectroscopy*, Cambridge University Press, 1990.
8. W.R. Smythe, J. Mattauch, 'A New Mass Spectrometer', *Phys. Rev.*, **40**, 429–433 (1932).
9. W.C. Wiley, I.H. McLaren, 'Time-of-flight Mass Spectrometer with Improved Resolution', *Rev. Sci. Inst.*, **26**, 1150–1157 (1955).
10. W. Paul, H.P. Reinhard, U. von Zahn, 'Das elektrische Massenfilter als Massenspektrometer und Isotopentrenner', *Z. Phys.*, **152**, 143–182 (1958).
11. R.F. Bonner, G. Lawson, J.F.J. Todd, 'Ion–Molecule Reaction Studies with Quadrupole Ion Storage Trap', *Int. J. Mass Spectrom. Ion Phys.*, **197**, 197–203 (1972/73).
12. J.D. Baldeschwieler, H. Benz, P.M. Llewellyn, 'Ion–Molecule Reactions in an Ion Cyclotron Resonance Mass Spectrometer', *Advances in Mass Spectrometry*, ed. E. Kendrick, Institute of Petroleum, London, Vol. 4, 113–120, 1968.
13. M.B. Comisarow, A.G. Marshall, 'Theory of Fourier Transform Ion Cyclotron Resonance Mass Spectroscopy. I. Fundamental Equations and Low Pressure Line Shape', *J. Chem. Phys.*, **64**, 110–119 (1976).
14. S. Savory (ed.), *Artificial Intelligence and Expert Systems*, Halsted Press, Chichester, 1988.
15. R.A. Edmunds, *The Prentice Hall Guide to Expert Systems*, Prentice Hall, Englewood Cliffs, NJ, 1988.
16. J.M. Gillette, 'Mass Spectrometer Data Reduction Program for IBM 650', *Anal. Chem.*, **31**(9), 1518–1521 (1959).
17. J. Lederberg, G.L. Sutherland, B.G. Buchanan, E.A. Feigenbaum, A.V. Robertson, A.M. Duffield, C. Djerassi, 'Applications of Artificial Intelligence for Chemical Inference. I. The Number of Possible Organic Compounds. Acyclic Structures Containing C, H, O and N', *J. Am. Chem. Soc.*, **11**, 2973–2976 (1968).
18. Mass Spectrometry Data Centre; Imperial Chemical Industries, *Eight Peak Index of Mass Spectra: The Eight Most Abundant Ions in 66,720 Mass Spectra, Indexed by Molecular Weight, Elemental Composition and Most Abundant Ions*, The Centre, Royal Society of Chemistry, Nottingham, UK, 1983.
19. D.J. Pappin, 'Peptide Mass Fingerprinting Using MALDI-TOF Mass Spectrometry', *Methods Mol. Biol.*, **64**, 165–173 (1997).

20. R.S. Brown, J.J. Lennon, 'Sequence-specific Fragmentation of Matrix-assisted Laser-desorbed Protein/Peptide Ions', *Anal. Chem.*, **67**, 3990–3999 (1995).
21. J. Qin, B.T. Chait, 'Identification and Characterization of Posttranslational Modifications of Proteins by MALDI Ion Trap Mass Spectrometry', *Anal. Chem.*, **69**, 4002–4009 (1997).
22. A. Shevchenko, I. Chenushevich, W. Ens, K.G. Standing, B. Thomson, M. Wilm, M. Mann, 'Rapid 'de novo' Peptide Sequencing by a Combination of Nanoelectrospray, Isotope Labeling and A Quadrupole/Time-of-flight Mass Spectrometer', *Rapid Commun. Mass Spectrom.*, **11**, 1015–1024 (1997).
23. A. Shevchenko, O.N. Jensen, A.V. Podtelejnikov, F. Sagliocco, M. Wilm, O. Vorm, P. Mortensen, A. Shevchenko, H. Boucherie, M. Mann, 'Linking Genome and Proteome by Mass Spectrometry: Large Scale Identification of Yeast Proteins from Two Dimensional Gels', *Proc. Nat. Acad. Sci. USA*, **93**, 14 440–14 445 (1996).
24. J.R. Yates, 'Mass Spectrometry. From Genomics to Proteomics', *Trends Genet.*, **16**, 5–8 (2000).
25. M.S. Boguski, T.M. Lowe, C.M. Tolstoshev, 'dbEST – Database for ''Expressed Sequence Tags'' ', *Nat. Genet.*, **4**, 332–333 (1993).
26. W. Zhang, B.T. Chait, 'ProFound – An Expert System for Protein Identification Using Mass Spectrometric Peptide Mapping Information', *Anal. Chem.*, **72**, in press (2000).
27. R.C. Beavis, B.T. Chait, 'Matrix-assisted Laser Desorption Ionization Mass Spectrometry of Proteins', *Methods Enzymol.*, **270**, 519–551 (1996).
28. D.H. Patterson, G.E. Tarr, F.E. Regnier, S.A. Martin, 'C-terminal Ladder Sequencing Via Matrix-assisted Laser Desorption Mass Spectrometry Coupled with Carboxypeptidase Y Time-dependent and Concentration-dependent Digestions', *Anal. Chem.*, **67**, 3971–3978.
29. H. Hotelling, 'Analysis of a Complex of Statistical Variables into Principal Components', *J. Educat. Psychol.*, **24**, 417–441 (1933).
30. R.M. Wallace, 'Analysis of Absorption Spectra of Multicomponent Systems', *J. Phys. Chem.*, **64**, 899–901 (1960).
31. R.M. Wallace, S.M. Katz, 'A Method for the Determination of Rank in the Analysis of Rank in the Analysis of Absorption Spectra of Multicomponent Systems', *J. Phys. Chem.*, **68**(12), 3890–3892 (1964).
32. P.T. Funke, E.R. Malinowski, D.E. Martire, L.Z. Pollara, 'Application of Factor Analysis to the Prediction of Activity Coefficients of Nonelectrolytes', *Sep. Sci.*, **1**(6), 661–676 (1966).
33. P.H. Weiner, E.R. Malinowski, A.R. Levinstone, 'Factor Analysis of Solvent Shifts in Proton Magnetic Resonance', *J. Phys. Chem.*, **74**(26), 4537–4542 (1970).
34. P.H. Weiner, D.G. Howery, 'Factor Analysis of Some Chemical and Physical Influences in Gas–Liquid Chromatography', *Anal. Chem.*, **44**(7), 1189–1194 (1972).
35. P.H. Weiner, C.J. Dack, D.G. Howery, 'Retention Index-structure Relationships for Alcohols Using Factor Analysis', *J. Chromatogr.*, **69**, 249–260 (1972).
36. P.H. Weiner, J.F. Parcher, 'Prediction of Some Physical Properties of Organic Molecules by Factor Analysis of Gas Chromatographic Retention Indices', *Anal. Chem.*, **45**(2), 302–307 (1973).
37. S. Wold, K. Andersson, 'Major Components Influencing Retention Indices in Gas Chromatography', *J. Chromatogr.*, **80**, 43–59 (1973).
38. D. Macnaughtan, Jr, L.B. Rogers, G. Wernimont, 'Principal-component Analysis Applied to Chromatographic Data', *Anal. Chem.*, **44**(8), 1421–1427 (1972).
39. R.W. Rozett, E.M. Petersen, 'Methods of Factor Analysis of Mass Spectra', *Anal. Chem.*, **47**(8), 1301–1308 (1975).
40. J.B. Justice, Jr, T.L. Isenhour, 'Factor Analysis of Mass Spectra', *Anal. Chem.*, **47**(13), 2286–2288 (1975).
41. R.W. Rozett, E.M. Petersen, 'Classification of Compounds by the Factor Analysis of Their Mass Spectra', *Anal. Chem.*, **48**(6), 817–825 (1976).
42. G.L. Ritter, S.R. Lowry, T.L. Isenhour, C.L. Wilkins, 'Factor Analysis of the Mass Spectra of Mixtures', *Anal. Chem.*, **48**(3), 591–595 (1976).
43. F.J. Knorr, J.H. Futrell, 'Separation of Mass Spectra of Mixtures by Factor Analysis', *Anal. Chem.*, **51**(8), 1236–1241 (1979).
44. P.C. Tway, L.J. Cline Love, H.B. Woodruff, 'A Totally Automated Data Acquisition/Reduction System for Routine Treatment of Mass Spectrometric Data by Factor Analysis', *Anal. Chim. Acta*, **117**, 45–52 (1980).
45. H.L.C. Meuzelaar, J. Haverkamp, F.D. Hileman, *Pyrolysis Mass Spectrometry of Recent and Fossil Biomaterials*, Elsevier, Amsterdam, 1982.
46. W. Windig, J. Haverkamp, P.G. Kistemaker, 'Interpretation of Sets of Pyrolysis Mass Spectra by Discriminant Analysis and Graphical Rotation', *Anal. Chem.*, **55**(1), 81–88 (1983).
47. W. Windig, H.L.C. Meuzelaar, 'Nonsupervised Numerical Component Extraction from Pyrolysis Mass Spectra of Complex Mixtures', *Anal. Chem.*, **56**(13), 2297–2303 (1984).
48. H.J.H. MacFie, C.S. Gutteridge, 'Comparative Studies on some Methods for Handling Quantitative Data Generated by Analytical Pyrolysis', *J. Anal. Appl. Pyrolysis*, **4**, 175–204 (1982).
49. L.V. Vallis, H.J. MacFie, C.S. Gutteridge, 'Comparison of Canonical Variates Analysis with Target Rotation and Least-squares Regression as Applied to Pyrolysis Mass Spectra of Simple Biochemical Mixtures', *Anal. Chem.*, **57**(3), 704–709 (1985).
50. J.T. Magee, 'Whole-organism Fingerprinting', in *Handbook of New Bacterial Systematics*, eds. M. Goodfellow, A.G. O'Donnell, Academic Press, London, 383–427, 1993.

51. G. Blomquist, E. Johansson, B. Soderstrom, S. Wold, 'Data Analysis of Pyrolysis–Chromatograms by Means of SIMCA Pattern Recognition', *J. Anal. Appl. Pyrolysis*, **1**, 53–65 (1979).
52. F.R. Di Brozolo, R.W. Odom, P.d.B. Harrington, K.J. Voorhees, 'Organic Polymer Analysis by Laser Ionization Mass Spectrometry and Pattern Recognition Techniques', *J. Appl. Polym. Sci.*, **41**, 1737–1752 (1990).
53. M. Lamberto, M. Saitta, 'Principal Component Analysis in Fast-atom-bombardment Mass-spectrometry of Triacylglycerols in Edible Oils', *J. Am. Oil Chem. Soc.*, **72**, 867–871 (1995).
54. R. Goodacre, J.K. Heald, D.B. Kell, 'Characterization of Intact Microorganisms Using Electrospray Ionization Mass Spectrometry', *FEMS Microbiol. Lett.*, **176**, 17–24 (1999).
55. P. Zheng, P.B. Harrington, A. Craig, R. Felming, 'Cluster Analysis for Variable Alignment of High Resolution Data', *Anal. Chim. Acta*, **310**, 485–492 (1995).
56. P.J. Tandler, J.A. Butcher, H. Tao, P.B. Harrington, 'Chemometric Analysis of Plastic Recycling Products', *Anal. Chim. Acta*, **312**(3), 231–244 (1995).
57. W.H. Press, S.A. Teukolsky, W.T. Vetterling, B.P. Flannery, 'Numerical Recipes in C', in *Numerical Recipes in C. The Art of Scientific Computing*, Cambridge University Press, 59–70, 1994.
58. M.B. Seaholtz, R.J. Pell, K.E. Gates, 'Comments on the Power Method', *J. Chemom.*, **4**, 331–334 (1990).
59. E.R. Malinowski, 'Determination of the Number of Factors and the Experimental Error in a Data Matrix', *Anal. Chem.*, **49**(4), 612–617 (1977).
60. J.H. Kalivas, P. Lang, 'Condition Numbers, Iterative Refinement and Error Bounds', *J. Chemom.*, **3**, 443–449 (1989).
61. J.H. Kalivas, P.M. Lang, 'Interrelationships Between Sensitivity and Selectivity Measures for Spectroscopic Analysis', *Chemom. Intell. Lab. Syst.*, **32**, 135–149 (1996).
62. L.R. Crawford, J.D. Morrison, 'Computer Methods in Analytical Mass Spectrometry Empirical Identification of Molecular Class', *Anal. Chem.*, **40**(10), 1469–1474 (1968).
63. R.H. Lindeman, P.F. Merenda, R.Z. Gold, *Introduction to Bivariate and Multivariate Analysis*, Scott, Foresman & Co., New York, 444, 1980.
64. B.F.J. Manly, *Multivariate Statistical Methods: A Primer*, Chapman & Hall, London, 1994.
65. C. Cai, P.B. Harrington, 'Linear Discriminant Classification of Fourier and Wavelet Compressed IMS Data', *Appl. Spectrosc.*, (submitted).
66. W.J. Dixon, *Biomedical Computer Programs*, University of California Press, Los Angeles, 1975.
67. H.F. Kaiser, 'The Varimax Criterion for Analytic Rotation in Factor Analysis', *Psychometrika*, **23**, 187–200 (1958).
68. P. Geladi, B.R. Kowalski, 'Partial Least-squares Regression: A Tutorial', *Anal. Chim. Acta*, **185**, 1–17 (1986).
69. S. Wold, P. Geladi, K. Esbensen, J. Ohman, 'Principal Components- and PLS-analyses Generalized to Multiway (Multi-order) Data Arrays', *J. Chemom.*, **1**, 41–56 (1987).
70. R. Goodacre, M.J. Neal, D.B. Kell, 'Rapid and Quantitative Analysis of the Pyrolysis Mass Spectra of Complex Binary and Tertiary Mixtures Using Multivariate Calibration and Artificial Neural Networks', *Anal. Chem.*, **66**, 1070–1085 (1994).
71. T.J. McAvoy, H.T. Su, N.S. Wang, M. He, J. Horvath, H. Semerjian, 'A Comparison of Neural Networks and Partial Least-squares for Deconvoluting Fluorescence Spectra', *Bioltechnol. Bioeng.*, **40**, 53–62 (1992).
72. W.P. Carey, K.R. Beebe, E. Sanchez, P. Geladi, B. Kowalski, 'Chemometric Analysis of Multisensor Arrays', *Sens. Actuators*, **9**, 223–234 (1986).
73. L. Tucker, 'Some Mathematical Notes on Three-mode Factor Analysis', *Psychometrika*, **31**, 279–311 (1966).
74. E.R. Malinowski, *Factor Analysis in Chemistry*, 2nd edition, John Wiley, New York, 1991.
75. S.D. Brown, 'Chemical Systems Under Indirect Observation: Latent Properties and Chemometrics', *Appl. Spectrosc.*, **49**(12), 14A–31A (1995).
76. H.B. Keller, D.L. Massart, 'Evolving Factor Analysis', *Chemom. Intell. Lab. Syst.*, **12**, 209–224 (1992).
77. P.K. Hopke, 'Target Transformation Factor Analysis', *Chemom. Intell. Lab. Syst.*, **6**, 7–19 (1989).
78. J. Toft, 'Evolutionary Rank Analysis Applied to Multidetectional Chromatographic Structures', *Chemom. Intell. Lab. Syst.*, **29**(2), 189–212 (1995).
79. F. Brakstad, 'The Feasibility of Latent Variables Applied to GC–MS Data', *Chemom. Intell. Lab. Syst.*, **29**, 157–176 (1995).
80. R. Bro, 'PARAFAC. Tutorial and Applications', *Chemom. Intell. Lab. Syst.*, **38**(2), 149–171 (1997).
81. H. Martens, T. Næs, *Multivariate Calibration*, John Wiley, Chichester, 1989.
82. H. Mark, *Principles and Practice of Spectroscopic Calibration*, John Wiley & Sons, New York, 1991.
83. H. Martens, T. Næs, *Multivariate Calibration*, John Wiley, Chichester, 1989.
84. C. Chatfield, A.J. Collins, *Introduction to Multivariate Analysis*, Chapman & Hall, London, 1980.
85. C.S. Gutteridge, L. Vallis, H.J.H. MacFie, 'Numerical Methods in the Classification of Microorganisms by Pyrolysis Mass Spectrometry', in *Computer-assisted Bacterial Systematics*, eds. M. Goodfellow, D. Jones, F. Priest, Academic Press, London, 369–401, 1985.
86. J.T. Magee, 'Whole-organism Fingerprinting', in *Handbook of New Bacterial Systematics*, eds. M. Goodfellow, A.G. O'Donnell, Academic Press, London, 383–427, 1993.

87. R. Goodacre, D.B. Kell, 'Pyrolysis Mass Spectrometry and its Applications in Biotechnology', *Curr. Opin. Biotechnol.*, **7**, 20–28 (1996).
88. I.T. Jolliffe, *Principal Component Analysis*, Springer-Verlag, New York, 1986.
89. D.R. Causton, *A Biologist's Advanced Mathematics*, Allen and Unwin, London, 1987.
90. B.S. Everitt, *Cluster Analysis*, Edward Arnold, London, 1993.
91. H.L.C. Meuzelaar, J. Haverkamp, F.D. Hileman, *Pyrolysis Mass Spectrometry of Recent and Fossil Biomaterials*, Elsevier, Amsterdam, 1982.
92. D.L. Massart, B.G.M. Vandeginste, S.N. Deming, Y. Michotte, L. Kaufmann, *Chemometrics: A Textbook*, Elsevier, Amsterdam, 1988.
93. R.G. Brereton, *Multivariate Pattern Recognition in Chemometrics*, Elsevier, Amsterdam, 1992.
94. S.D. Brown, S.T. Sum, F. Despagne, B.K. Lavine, 'Chemometrics', *Anal. Chem.*, **68**, R21–R61 (1996).
95. D.L. Massart, B.G.M. Vandeginste, L.M.C. Buydens, S. DeJong, P.J. Lewi, J. Smeyers-Verbeke, *Handbook of Chemometrics and Qualimetrics: Part A*, Elsevier, Amsterdam, 1997.
96. B.K. Lavine, 'Chemometrics', *Anal. Chem.*, **70**, R209–R228 (1998).
97. S. Wold, N. Kettaneh-Wold, B. Skagerberg, 'Nonlinear PLS Modelling', *Chem. Intell. Lab. Sys.*, **7**, 53–65 (1989).
98. A. Höskuldsson, 'Quadratic PLS Regression', *J. Chemom.*, **6**, 307–334 (1992).
99. V.M. Taavitsainen, P. Korhonen, 'Nonlinear Data Analysis with Latent Variable', *Chem. Intell. Lab. Sys.*, **14**, 185–194 (1992).
100. S. Wold, 'Nonlinear Partial Least-squares Modeling. 2. Spline Inner Relation', *Chem. Intell. Lab. Sys.*, **14**, 71–84 (1992).
101. B. Walczak, D.L. Massart, 'The Radial Basis Functions – Partial Least Squares Approach as a Flexible Non-linear Regression Technique', *Anal. Chim. Acta*, **331**, 177–185 (1996).
102. A. Berglund, S. Wold, 'INLR, Implicit Non-linear Latent Variable Regression', *J. Chemom.*, **11**, 141–156 (1997).
103. R.A. Heikka, K.T. Immonen, P.O. Minkkinen, E.Y.O. Paatero, T.O. Salmi, 'Determination of Acid Value, Hydroxyl Value and Water Content in Reactions Between Dicarboxylic Acids and Diols Using Near-infrared Spectroscopy and Non-linear Partial Least Squares Regression', *Anal. Chim. Acta*, **349**, 287–294 (1997).
104. P.D. Wasserman, *Neural Computing: Theory and Practice*, Van Nostrand Reinhold, New York, 1989.
105. R. Goodacre, D.B. Kell, 'Rapid and Quantitative Analysis of Bioprocesses Using Pyrolysis Mass Spectrometry and Neural Networks – Application to Indole Production', *Anal. Chim. Acta*, **279**, 17–26 (1993).
106. J. Zupan, J. Gasteiger, *Neural Networks for Chemists: An Introduction*, VCH, Weinheim, 1993.
107. R. Goodacre, M.J. Neal, D.B. Kell, 'Rapid and Quantitative Analysis of the Pyrolysis Mass Spectra of Complex Binary and Tertiary Mixtures Using Multivariate Calibration and Artificial Neural Networks', *Anal. Chem.*, **66**, 1070–1085 (1994).
108. C.M. Bishop, *Neural Networks for Pattern Recognition*, Clarendon Press, Oxford, 1995.
109. D.B. Kell, B. Sonnleitner, 'GMP – Good Modelling Practice: An Essential Component of Good Manufacturing Practice', *Trends Biotechnol.*, **13**, 481–492 (1995).
110. C. Wan, P.B. Harrington, 'Screening GC–MS Data for Carbamate Pesticides with Temperature Constrained-Cascade Correlation Networks', *Anal. Chim. Acta*, (in Press).
111. R. Goodacre, A.N. Edmonds, D.B. Kell, 'Quantitative Analysis of the Pyrolysis Mass Spectra of Complex Mixtures Using Artificial Neural Networks – Application to Amino Acids in Glycogen', *J. Anal. Appl. Pyrol.*, **26**, 93–114 (1993).
112. B.R. Kowalski, K.F. Faber, 'Comments on a Recent Sensitivity Analysis of Radial Base Function and Multilayer Feed-forward Neural Network Models', *Chem. Intell. Lab. Syst.*, **34**, 293–297 (1996).
113. R.C. Ebhart, I. Cloete, J.M. Zurada, 'Determining the Significance of Input Parameters Using Sensitivity Analysis', *Lecture Notes Computer Science*, **930**, 382–388 (1995).
114. P. Howes, N. Crook, 'Using Input Parameter Influences to Support the Decision of Feedforward Neural Networks', *Neurocomputing*, **24**, 191–206 (1999).
115. J.Y. Choi, C.H. Choi, 'Sensitivity Analysis of Multilayer Perceptron with Differentiable Activation Functions', *IEEE Trans. Neural Networks*, **3**, 101–107 (1992).
116. V.V. Kovalishyn, I.V. Tetko, A.I. Luik, V.V. Kholodovych, A.E.P. Villa, D.J. Livingstone, 'Neural Network Studies. 3. Variable Selection in the Cascade-correlation Learning Architecture', *J. Chem. Inf. Comput. Sci.*, **38**, 651–659 (1998).
117. A. Ikonomopoulos, 'Wavelet Decomposition and Radial Basis Function Networks for System Monitoring', *IEEE Trans. Neural Sci.*, **45**, 2293–2301 (1998).
118. A.H. Sung, 'Ranking Importance of Input Parameters of Neural Networks', *Expert System with Applications*, **15**, 405–411 (1998).
119. T.D. Gedeon, 'Data Mining of Inputs: Analyzing Magnitude and Functional Measures', *Int. J. Neural Syst.*, **8**, 209–218 (1997).
120. M. Koda, 'Stochastic Sensitivity Analysis Method for Neural-network Learning', *Int. J. Syst. Sci.*, **26**, 703–711 (1995).
121. P.B. Harrington, C. Wan, A. Urbas, 'Generalized Sensitivity Analysis of Neural Network Models', *J. Am. Chem. Soc.*, (submitted).

122. R. Hecht-Nielsen, *Neurocomputing*, Addison-Wesley, Massachusetts, 1990.
123. J. Hertz, A. Krogh, R.G. Palmer, *Introduction to the Theory of Neural Computation*, Addison-Wesley, California, 1991.
124. T. Kohonen, *Self-organizing Maps*, ed. S.S.i.I. Sciences, Springer, Berlin, Heidelberg, New York, Vol. 30, 1997.
125. T. Kohonen, *Self-organization and Associative Memory*, Springer-Verlag, Berlin, 1989.
126. E. Oja, S. Kaski, *Kohonen Maps*, Elsevier, Amsterdam, 1999.
127. R. Goodacre, M.J. Neal, D.B. Kell, L.W. Greenham, W.C. Noble, R.G. Harvey, 'Rapid Identification Using Pyrolysis Mass Spectrometry and Artificial Neural Networks of *Propionibacterium acnes* Isolated from Dogs', *J. Appl. Bacteriol.*, **76**, 124–134 (1994).
128. R. Goodacre, J. Pygall, D.B. Kell, 'Plant Seed Classification Using Pyrolysis Mass Spectrometry with Unsupervised Learning; the Application of Auto-associative and Kohonen Artificial Neural Networks', *Chem. Intell. Lab. Sys.*, **34**, 69–83 (1996).
129. R. Goodacre, A.C. McGovern, N. Kaderbhai, E.A. Goodacre, 'Chemometric Analyses with Self Organizing Feature Maps: A Worked Example of the Analysis of Cosmetics Using Raman Spectroscopy', in *Kohonen Maps*, eds. E. Oja, S. Kaski, Elsevier, Amsterdam, 335–347, 1999.
130. E. Erwin, K. Obermayer, K. Schulten, 'Self-organizing maps: Ordering, Convergence Properties and Energy Functions', *Biol. Cyber.*, **67**, 47–55 (1992).
131. I. Belic, L. Gyergyek, 'Neural Network Methodologies for Mass Spectra Recognition', *Vacuum*, **48**, 633–637 (1997).
132. H. Lohninger, F. Stancl, 'Comparing the Performance of Neural Networks to Well Established Methods of Multivariate Data Analysis – the Classification of Mass Spectral Data', *Fresenius J. Anal. Chem.*, **344**, 186–189 (1992).
133. R. Goodacre, S.A. Howell, W.C. Noble, M.J. Neal, 'Subspecies Discrimination Using Pyrolysis Mass Spectrometry and Self-organizing Neural Networks of *Propionibacterium acnes* Isolated from Normal Human Skin', *Zbl. Bakt. – Int. J. Med. M.*, **284**, 501–515 (1996).
134. M.A. Kramer, 'Non Linear Principal Components Analysis Using Auto-associative Neural Networks', *AIChE J.*, **37**, 233–243 (1991).
135. M.A. Kramer, 'Autoassociative Neural Networks', *Comput. Chem. Eng.*, **16**, 313–328 (1992).
136. D.E. Rumelhart, J.L. McClelland, *Parallel Distributed Processing, Experiments in the Microstructure of Cognition*, PDP Research Group, MIT Press, Cambridge, MA, Vols. I and II, 1986.
137. P.J. Werbos, *The Roots of Back-propagation: From Ordered Derivatives to Neural Networks and Political Forecasting*, John Wiley, Chichester, 1994.
138. Y. Chauvin, D.E. Rumelhart, *Backpropagation: Theory, Architectures, and Applications*, Erlbaum, Hove, UK, 1995.
139. R. Goodacre, D.J. Rischert, P.M. Evans, D.B. Kell, 'Rapid Authentication of Animal Cell Lines Using Pyrolysis Mass Spectrometry and Auto-associative Artificial Neural Networks', *Cytotechnol.*, **21**, 231–241 (1996).
140. S.P. Jacobsson, 'Feature Extraction of Polysaccharides by Low-dimensional Internal Representation Neural Networks and Infrared Spectroscopy', *Anal. Chim. Acta*, **291**, 19–27 (1994).
141. G. Montague, J. Morris, 'Neural Network Contributions in Biotechnology', *Trends Biotechnol.*, **12**, 312–324 (1994).
142. D.R. Kuespert, T.J. McAvoy, 'Knowledge Extraction in Chemical Process Control', *Chem. Eng. Commun.*, **130**, 251–264 (1994).
143. B. Wabuyele, P.B. Harrington, 'Optimal Associative Memory for Baseline Correction of Infrared Spectra', *Anal. Chem.*, **66**, 2047–2051 (1994).
144. B. Wabuyele, P.B. Harrington, 'Quantitative Comparison of Optimal and Bidirectional Associative Memories for Background Prediction of Infrared Spectra', *Chemom. Intell. Lab. Syst.*, **29**, 51–61 (1995).
145. B. Wabuyele, P.B. Harrington, 'Fuzzy Optimal Associative Memories', *Appl. Spectrosc.*, **50**, 35–42 (1996).
146. W.G. Baxt, 'Application of Artificial Neural Networks to Clinical Medicine', *The Lancet*, **346**, 1135–1138 (1995).
147. A. Cawsey, *The Essence of Artificial Intelligence*, Prentice Hall, London, 1998.
148. R. Dybowski, V. Gant, 'Artificial Neural Networks in Pathological and Medical Laboratories', *Lancet*, **346**, 1203–1207 (1995).
149. R. Goodacre, M.J. Neal, D.B. Kell, 'Quantitative Analysis of Multivariate Data Using Artificial Neural Networks: A Tutorial Review and Applications to the Deconvolution of Pyrolysis Mass Spectra', *Zbl. Bakt. – Int. J. Med. M.*, **284**, 516–539 (1996).
150. S. Haykin, *Neural Networks*, Macmillan, New York, 1994.
151. B.D. Ripley, 'Neural Networks and Related Methods for Classification', *J. Roy. Stats. Soc. Ser. B*, **56**, 409–437 (1994).
152. B.D. Ripley, *Pattern Recognition and Neural Networks*, Cambridge University Press, Cambridge, 1996.
153. M.D. Richard, R.P. Lippmann, 'Neural Network Classifiers Estimate Bayesian *a Posteriori* Probabilities', *Neural Computation*, **3**, 461–483 (1991).
154. P.K. Simpson, *Artificial Neural Systems*, Pergamon Press, Oxford, 1990.
155. S.S. Haykin, *Neural Networks: A Comprehensive Foundation*, Macmillan, New York, 1994.
156. G. Cybenko, 'Approximation by Superposition of a Sigmoidal Function', *Mathematical Control Signals Systems*, **2**, 303–314 (1989).

157. K. Funahashi, 'On the Approximate Realization of Continuous-mappings by Neural Networks', *Neural Networks*, **2**, 183–192 (1989).
158. K. Hornik, M. Stinchcombe, H. White, 'Multilayer Feedforward Networks are Universal Approximators', *Neural Networks*, **2**, 359–368 (1989).
159. K. Hornik, M. Stinchcombe, H. White, 'Universal Approximation of an Unknown Mapping and Its Derivatives Using Multilayer Feedforward Networks', *Neural Networks*, **3**, 551–560 (1990).
160. H. White, 'Connectionist Nonparametric Regression – Multilayer Feedforward Networks Can Learn Arbitrary Mappings', *Neural Networks*, **3**, 535–549 (1990).
161. H. White, *Artificial Neural Networks: Approximation and Learning Theory*, Blackwell, Oxford, 1992.
162. P.B. Harrington, B.W. Pack, 'FLIN: Fuzzy Linear Interpolating Network', *Anal. Chim. Acta*, **277**, 189–197 (1993).
163. S.E. Fahlman, C. Lebiere, *The Cascade-correlation Learning Architecture*, Carnegie Mellon University, Pittsburgh, 1991.
164. P.B. Harrington, 'Fuzzy Rule-building Expert Systems: Minimal Neural Networks', *J. Chemom.*, **5**, 467–486 (1991).
165. P.B. Harrington, 'Temperature Constrained Backpropagation Neural Networks', *Anal. Chem.*, **66**, 802–807 (1994).
166. P.B. Harrington, 'Temperature-constrained Cascade Correlation Networks', *Anal. Chem.*, **70**, 1297–1306 (1998).
167. D.S. Broomhead, D. Lowe, 'Multivariable Functional Interpolation and Adaptive Networks', *Complex Systems*, **2**, 312–355 (1988).
168. J. Moody, C.J. Darken, 'Fast Learning in Networks of Locally-tuned Processing Units', *Neural Computation*, **1**, 281–294 (1989).
169. R. Beale, T. Jackson, *Neural Computing: An Introduction*, Adam Hilger, Bristol, 1990.
170. A. Saha, J.D. Keller, 'Algorithms for Better Representation and Faster Learning in Radial Basis Functions', in *Advances in Neural Information Processing Systems*, ed. D. Touretzky, Morgan Kaufmann Publishers, San Mateo, CA, 482–489, Vol. 2, 1990.
171. J. Park, I.W. Sandberg, 'Universal Approximation Using Radial Basis Function Networks', *Neural Computation*, **3**, 246–257 (1991).
172. D.R. Hush, B.G. Horne, 'Progress in Supervised Neural Networks – What's New Since Lippmann', *IEEE Signal Processing Magazine*, **10**, 8–39 (1993).
173. M.F. Wilkins, C.W. Morris, L. Boddy, 'A Comparison of Radial Basis Function and Backpropagation Neural Networks for Identification of Marine Phytoplankton from Multivariate Flow Cytometry Data', *Comput. Appl. Biosci.*, **10**, 285–294 (1994).
174. R.O. Duda, P.E. Hart, *Pattern Classification and Scene Analysis*, Wiley, New York, (1973).
175. C. Wan, P.B. Harrington, 'Self-configuring Radial Basis Function Neural Network for Chemical Pattern Recognition', *J. Chem. Inf. Comput. Sci.*, **39**, 1049–1056 (1999).
176. R. Goodacre, 'Characterization and Quantification of Microbial Systems Using Pyrolysis Mass Spectrometry: Introducing Neural Networks to Analytical Pyrolysis', *Microbiol. Eur.*, **2**, 16–22 (1994).
177. R. Goodacre, D.B. Kell, G. Bianchi, 'Neural Networks and Olive Oil', *Nature*, **359**, 594–594 (1992).
178. R. Goodacre, D.B. Kell, G. Bianchi, 'Rapid Assessment of the Adulteration of Virgin Olive Oils by Other Seed Oils Using Pyrolysis Mass Spectrometry and Artificial Neural Networks', *J. Sci. Food Agric.*, **63**, 297–307 (1993).
179. C. Guillou, M. Lipp, B. Radovic, F. Reniero, M. Schmidt, E. Anklam, 'Use of Pyrolysis Mass Spectrometry in Food Analysis: Applications in the Food Analysis Laboratory of the European Commissions' Joint Research Centre', *J. Anal. Appl. Pyrol.*, **49**, 329–335 (1999).
180. G.J. Salter, M. Lazzari, L. Giansante, R. Goodacre, A. Jones, G. Surricchio, D.B. Kell, G. Bianchi, 'Determination of the Geographical Origin of Italian Extra Virgin Olive Oil Using Pyrolysis Mass Spectrometry and Artificial Neural Networks', *J. Anal. Appl. Pyrol.*, **40/41**, 159–170 (1997).
181. E. Anklam, M.R. Bassani, T. Eiberger, S. Kriebel, M. Lipp, R. Matissek, 'Characterization of Cocoa Butters and Other Vegetable Fats by Pyrolysis Mass Spectrometry', *Fresenius J. Anal. Chem.*, **357**, 981–984 (1997).
182. B.S. Radovic, M. Lipp, E. Anklam, 'Classification of Cocoa Butters Using Pyrolysis Mass Spectrometry', *Rapid Commun. Mass Spectrom.*, **12**, 783–789 (1998).
183. R. Goodacre, 'Use of Pyrolysis Mass Spectrometry with Supervised Learning for the Assessment of the Adulteration of Milk of Different Species', *Appl. Spectrosc.*, **51**, 1144–1153 (1997).
184. M.A.E. Schmidt, B.S. Radovic, M. Lipp, G. Harzer, E. Anklam, 'Characterization of Milk Samples with Various Whey Protein Contents by Pyrolysis–Mass Spectrometry (Py–MS)', *Food Chem.*, **65**, 123–128 (1999).
185. J.L. Berdague, C. Rabot, M. Bonneau, 'Rapid Classification of Backfat Samples Selected According to Their Androstenone Content Pyrolysis–Mass Spectrometry', *Sciences Des Aliments*, **16**, 425–433 (1996).
186. L. Montanarella, M.R. Bassani, O. Breas, 'Chemometric Classification of some European Wines Using Pyrolysis Mass Spectrometry', *Rapid Commun. Mass Spectrom.*, **9**, 1589–1593 (1995).
187. E. Anklam, M. Lipp, B. Radovic, E. Chiavaro, G. Palla, 'Characterization of Italian Vinegar by Pyrolysis Mass Spectrometry and a Sensor Device ('Electronic Nose')', *Food Chem.*, **61**, 243–248 (1998).
188. R. Goodacre, D. Hammond, D.B. Kell, 'Quantitative Analysis of the Adulteration of Orange Juice with

Sucrose Using Pyrolysis Mass Spectrometry and Chemometrics', *J. Anal. Appl. Pyrol.*, **40/41**, 135–158 (1997).
189. R. Goodacre, R.J. Gilbert, 'The Detection of Caffeine in a Variety of Beverages Using Curie-point Pyrolysis Mass Spectrometry and Genetic Programming', *Analyst*, **124**, 1069–1074 (1999).
190. R.G.W. Kenyon, E.V. Ferguson, A.C. Ward, 'Application of Neural Networks to the Analysis of Pyrolysis Mass Spectra', *Zentralblatt fur Bakteriologie–Int. J. Med. Microbiol. Virol. Parasitol. Infect. Dis.*, **285**, 267–277 (1997).
191. P.R. Sisson, R. Freeman, D. Law, A.C. Ward, N.F. Lightfoot, 'Rapid Detection of Verocytotoxin Production Status in *Escherichia coli* by Artificial Neural Network Analysis of Pyrolysis Mass Spectra', *J. Anal. Appl. Pyrol.*, **32**, 179–185 (1995).
192. R. Freeman, R. Goodacre, P.R. Sisson, J.G. Magee, A.C. Ward, N.F. Lightfoot, 'Rapid Identification of Species within the *Mycobacterium tuberculosis* Complex by Artificial Neural Network Analysis of Pyrolysis Mass Spectra', *J. Med. Microbiol.*, **40**, 170–173 (1994).
193. J. Chun, E. Atalan, A.C. Ward, M. Goodfellow, 'Artificial Neural Network Analysis of Pyrolysis Mass Spectrometric Data in the Identification of *Streptomyces* Strains', *FEMS Microbiol. Lett.*, **107**, 321–325 (1993).
194. J. Chun, E. Atalan, S.B. Kim, H.J. Kim, M.E. Hamid, M.E. Trujillo, J.G. Magee, G.P. Manfio, A.C. Ward, M. Goodfellow, 'Rapid Identification of *Streptomycetes* by Artificial Neural Network Analysis of Pyrolysis Mass Spectra', *FEMS Microbiol. Lett.*, **114**, 115–119 (1993).
195. J.S. Chun, A.C. Ward, S.O. Kang, Y.C. Hah, M. Goodfellow, 'Long-term Identification of Streptomycetes Using Pyrolysis Mass Spectrometry and Artificial Neural Networks', *Zentralblatt fur Bakteriologie–Int. J. Med. Microbiol. Virol. Parasitol. Infect. Dis.*, **285**, 258–266 (1997).
196. R. Goodacre, S.J. Hiom, S.L. Cheeseman, D. Murdoch, A.J. Weightman, W.G. Wade, 'Identification and Discrimination of Oral Asaccharolytic *Eubacterium* spp. Using Pyrolysis Mass Spectrometry and Artificial Neural Networks', *Cur. Microbiol.*, **32**, 77–84 (1996).
197. T. Nilsson, M.R. Bassani, T.O. Larsen, L. Montanarella, 'Classification of Species in the Genus *Penicillium* by Curie Point Pyrolysis Mass Spectrometry Followed by Multivariate Analysis and Artificial Neural Networks', *J. Mass Spectrom.*, **31**, 1422–1428 (1996).
198. R. Goodacre, P.J. Rooney, D.B. Kell, 'Discrimination Between Methicillin-resistant and Methicillin-susceptible *Staphylococcus aureus* Using Pyrolysis Mass Spectrometry and Artificial Neural Networks', *J. Antimicr. Chemother.*, **41**, 27–34 (1998).
199. P.E. Reynolds, C. Fuller, 'Methicillin Resistant Strains of *Staphylococcus aureus*; Presence of Identical Additional Penicillin Binding Protein in all Strains Examined', *FEMS Microbiol. Lett.*, **33**, 251–254 (1986).
200. D.M. O'Hara, P.E. Reynolds, 'Antibody Used to Identify Penicillin Binding Protein 2' in Methicillin Resistant Strains of *Staphylococcus aureus* (MRSA)', *FEBS Lett.*, **212**, 237–241 (1987).
201. R. Goodacre, B. Shann, R.J. Gilbert, É.M. Timmins, A.C. McGovern, B.K. Alsberg, D.B. Kell, N.A. Logan, 'The Detection of the Dipicolinic Acid Biomarker in *Bacillus* Spores Using Curie-point Pyrolysis Mass Spectrometry and Fourier Transform Infrared Spectroscopy', *Anal. Chem.*, **72**, 119–127 (2000).
202. S.G. Kang, R.G.W. Kenyon, A.C. Ward, K.J. Lee, 'Analysis of Differentiation State in *Streptomyces albidoflavus* SMF301 by the Combination of Pyrolysis Mass Spectrometry and Neural Networks', *J. Biotechnol.*, **62**, 1–10 (1998).
203. S.G. Kang, D.H. Lee, A.C. Ward, K.J. Lee, 'Rapid and Quantitative Analysis of Clavulanic Acid Production by the Combination of Pyrolysis Mass Spectrometry and Artificial Neural Network', *J. Microbiol. Biotechnol.*, **8**, 523–530 (1998).
204. R. Goodacre, É.M. Timmins, R. Burton, N. Kaderbhai, A.M. Woodward, D.B. Kell, P.J. Rooney, 'Rapid Identification of Urinary Tract Infection Bacteria Using Hyperspectral, Whole Organism Fingerprinting and Artificial Neural Networks', *Microbiology*, **144**, 1157–1170 (1998).
205. P.B. Harrington, 'Minimal Neural Networks: Differentiation of Classification Entropy', *Chem. Intell. Lab. Syst.*, **19**, 143–154 (1993).
206. P.D. Harrington, 'Minimal Neural Networks – Concerted Optimization of Multiple Decision Planes', *Chem. Intell. Lab. Syst.*, **18**, 157–170 (1993).
207. P.C. Jurs, B.R. Kowalski, T.L. Isenhour, 'Computerized Learning Machines Applied To Chemical Problems – Molecular Formula Determination From Low Resolution Mass Spectrometry', *Anal. Chem.*, **41**, 21–27 (1969).
208. B. Curry, D.E. Rumelhart, 'MSNet: A Neural Network That Classifies Mass Spectra', *Tetrahedron Comput. Methodol.*, **3**, 213 (1990).
209. W. Werther, H. Lohninger, F. Stancl, K. Varmuza, 'Classification of Mass Spectra – A Comparison of Yes/no Classification Methods for the Recognition of Simple Structural Properties', *Chemom. Intell. Lab. Syst.*, **22**, 63–76 (1994).
210. G. Blomquist, E. Johansson, B. Soderstrom, S. Wold, 'Data Analysis of Pyrolysis-Chromatograms by Means of SIMCA Pattern Recognition', *J. Anal. Appl. Pyrolysis*, **1**, 53–65 (1979).
211. R. Todeschini, 'K-nearest Neighbor Method: The Influence of Data Transformations and Metrics', *Chemom. Intell. Lab. Syst.*, **6**, 213–220 (1989).
212. R.H. Lindeman, P.F. Merenda, R.Z. Gold, *Introduction to Bivariate and Multivariate Analysis*, Scott, Foresman & Co., New York, 1980.

213. P.J. Tandler, J.A. Butcher, H. Tao, P.B. Harrington, 'Chemometric Analysis of Plastic Recycling Products', *Anal. Chim. Acta*, **312**, 231–244 (1995).
214. B.K. Alsberg, R. Goodacre, J.J. Rowland, D.B. Kell, 'Classification of Pyrolysis Mass Spectra by Fuzzy Multivariate Rule Induction – Comparison with Regression, K-nearest Neighbor, Neural and Decision-tree Methods', *Anal. Chim. Acta*, **348**, 389–407 (1997).
215. B.K. Alsberg, W.G. Wade, R. Goodacre, 'Chemometric Analysis of Diffuse Reflectance–Absorbance Fourier Transform Infrared Spectra Using Rule Induction Methods: Application to the Classification of *Eubacterium* Species', *Appl. Spectrosc.*, **52**, 823–832 (1998).
216. C. Cai, P.B. Harrington, 'Prediction of Substructure and Toxicity of Pesticides with Temperature Constrained-cascade Correlation Network from Low-resolution Mass Spectra', *Anal. Chem.*, **71**, 4134–4141 (1999).
217. É.M. Timmins, R. Goodacre, 'Rapid Quantitative Analysis of Binary Mixtures of *Escherichia coli* Strains Using Pyrolysis Mass Spectrometry with Multivariate Calibration and Artificial Neural Networks', *J. Appl. Microbiol.*, **83**, 208–218 (1997).
218. R. Goodacre, A. Karim, M.A. Kaderbhai, D.B. Kell, 'Rapid and Quantitative Analysis of Recombinant Protein Expression Using Pyrolysis Mass Spectrometry and Artificial Neural Networks – Application to Mammalian Cytochrome b_5 in *Escherichia coli*', *J. Biotechnol.*, **34**, 185–193 (1994).
219. R. Goodacre, S. Trew, C. Wrigley-Jones, M.J. Neal, J. Maddock, T.W. Ottley, N. Porter, D.B. Kell, 'Rapid Screening for Metabolite Overproduction in Fermentor Broths Using Pyrolysis Mass Spectrometry with Multivariate Calibration and Artificial Neural Networks', *Biotechnol. Bioeng.*, **44**, 1205–1216 (1994).
220. R. Goodacre, S. Trew, C. Wrigley-Jones, G. Saunders, M.J. Neal, N. Porter, D.B. Kell, 'Rapid and Quantitative Analysis of Metabolites in Fermentor Broths Using Pyrolysis Mass Spectrometry with Supervised Learning: Application to the Screening of *Penicillium chrysogenum* Fermentations for the Overproduction of Penicillins', *Anal. Chim. Acta*, **313**, 25–43 (1995).
221. A.C. McGovern, D. Broadhurst, J. Taylor, R.J. Gilbert, N. Kaderbhai, M.K. Winson, D.A. Small, J.J. Rowlard, D.B. Kell, R. Goodacre, 'Monitoring of Complex Industrial Bioprocesses for Metabolite Concentrations Using Modern Spectroscopies and Machine Learning: Application to Gibberellic Acid Production', *Biotechnol. Bioeng.*, (submitted).
222. A.C. McGovern, R. Ernill, B.V. Kara, D.B. Kell, R. Goodacre, 'Rapid Analysis of the Expression of Heterologous Proteins in *Escherichia coli* Using Pyrolysis Mass Spectrometry and Fourier Transform Infrared Spectroscopy with Chemometrics: Application to α2-interferon Production', *J. Biotechnol.*, **72**, 157–167 (1999).
223. A.D. Shaw, M.K. Winson, A.M. Woodward, A.C. McGovern, H.M. Dowey, N. Kaderbhai, D. Broadhurst, R.J. Gilbert, J. Taylor, E.M. Timmins, B.K. Alsberg, J.J. Rowland, R. Goodacre, D.B. Kell, 'Rapid Analysis of High-dimensional Bioprocesses Using Multivariate Spectroscopies and Advanced Chemometrics', in *Advances in Biochemical Engineering/Biotechnology*, ed. T. Scheper, Springer-Verlag, Berlin, 83–114, Vol. 66, 2000.
224. M.K. Winson, R. Goodacre, A.M. Woodward, É.M. Timmins, A. Jones, B.K. Alsberg, J.J. Rowland, D.B. Kell, 'Diffuse Reflectance Absorbance Spectroscopy Taking in Chemometrics (DRASTIC). A Hyperspectral FT–IR-based Approach to Rapid Screening for Metabolite Overproduction', *Anal. Chim. Acta*, **348**, 273–282 (1997).
225. A.D. Shaw, N. Kaderbhai, A. Jones, A.M. Woodward, R. Goodacre, J.J. Rowland, D.B. Kell, 'Noninvasive, On-line Monitoring of the Biotransformation by Yeast of Glucose to Ethanol Using Dispersive Raman Spectroscopy and Chemometrics', *Appl. Spectrosc.*, **53**, 1419–1428 (1999).
226. A.M. Woodward, A. Jones, X.Z. Zhang, J. Rowland, D.B. Kell, 'Rapid and Non-invasive Quantification of Metabolic Substrates in Biological Cell Suspensions Using Non-linear Dielectric Spectroscopy with Multivariate Calibration and Artificial Neural Networks. Principles and Applications', *Bioelectrochem. Bioenerg.*, **40**, 99–132 (1996).
227. H.M. Davey, D.B. Kell, 'Flow Cytometry and Cell Sorting of Heterogeneous Microbial Populations – the Importance of Single Cell Analyses', *Microbiol. Rev.*, **60**, 641–696 (1996).
228. M. Goodfellow, 'Inter-strain Comparison of Pathogenic Microorganisms by Pyrolysis Mass Spectrometry', *Binary – Comp. Microbiol.*, **7**, 54–60 (1995).
229. R. Goodacre, D.B. Kell, *Composition Analysis*, UK Patent, 1995; International Patent #WO 96/42058 of 27 December, 1996; US5946640 of 31 August, 1999.
230. R. Goodacre, D.B. Kell, 'Correction of Mass Spectral Drift Using Artificial Neural Networks', *Anal. Chem.*, **68**, 271–280 (1996).
231. R. Goodacre, É.M. Timmins, A. Jones, D.B. Kell, J. Maddock, M.L. Heginbothom, J.T. Magee, 'On Mass Spectrometer Instrument Standardization and Interlaboratory Calibration Transfer Using Neural Networks', *Anal. Chim. Acta*, **348**, 511–532 (1997).
232. W.H. Press, B.P. Flannery, S.A. Teukolsky, W.T. Vettering, *Numerical Recipes*, Cambridge University Press, Cambridge, 1986.
233. W. Spendley, G.R. Hext, F.R. Himsworth, *Technometrics*, **4**, 441–461 (1962).
234. M.J.D. Powell, *Comput J.*, **7**, 155–162 (1965).
235. J.A. Nelder, R. Mead, *Comput. J.*, **7**, 308–313 (1965).

236. S.M. Colby, T.B. King, J.P. Reilly, *Rapid Commun. Mass Spectrom.*, **8**, 865–868 (1994).
237. R.S. Brown, J.J. Lennon, *Anal. Chem.*, **67**, 1998–2003 (1995).
238. R.M. Whittal, L. Li, *Anal. Chem.*, **67**, 1950–1954 (1995).
239. M.L. Vestal, P. Huhasz, S.A. Martin, *Rapid Commun. Mass Spectrom.*, **9**, 1044–1050 (1995).
240. C.C. Vera, R. Zubarev, H. Ehring, P. Hakansson, B.U.R. Sunqvist, *Rapid Comm. Mass Spectrom.*, **10**, 1429–1432 (1996).
241. P. Juhasz, M.L. Vestal, S.A. Martin, *J. Am. Soc. Mass Spectrom.*, **8**, 209–217 (1997).
242. M. Vestal, P. Juhasz, *J. Am. Soc. Mass Spectrom.*, **9**, 892–911 (1998).
243. R.D. Edmonson, D.H. Russell, *J. Am. Soc. Mass Spectrom.*, **7**, 995–1001 (1996).
244. R.J. Arnold, J.P. Reilly, *J. Am. Chem. Soc.*, **120**, 1528–1532 (1998).