Computational Tools for the Processing and Analysis of Time-course Metabolomic Data

Martin James Rusilowicz

Doctor of Engineering

University of York
Computer Science

December 2016

ABSTRACT

Modern, high-throughput techniques for the acquisition of metabolomic data, combined with an increase in computational power, have provided not only the need for, but also the means to develop and use, methods for the interpretation of large and complex datasets. This thesis investigates the methods by which pertinent information can be extracted from non-targeted metabolomic data and reviews the current state of chemometric methods. The analysis of real-world data and research questions relevant to the agri-food industry reveals several problems for which novel solutions are proposed. Three LC-MS datasets are studied: *Medicago*, *Alopecurus* and aged Beef, covering stress resistance, herbicide resistance and product misbranding. The new methods include preprocessing (batch correction, data-filtering), processing (clustering, classification) and visualisation and their use facilitated within a flexible data-to-results pipeline. The resulting software suite with a user-friendly graphical interface is presented, providing a pragmatic realisation of these methods in an easy to access workflow.

CONTENTS

LIST OF TABLES

LIST OF FIGURES

LIST OF ACCOMPANYING MATERIAL

The software provided in Chapter 6 is available online at `https://bitbucket.org/mjr129/metabolitelevels`.

The results of the *Alopecurus* clustering are too large to fit in the appendix, and are available alongside the software, at `https://bitbucket.org/mjr129/metabolitelevels/downloads/ThesisResults.zip`.

ACKNOWLEDGEMENTS

I am very thankful to the EPSRC for providing the funding this project (grant number EP/F001096/1).

On a more personal level, firstly, I would like to my supervisors, Julie Wilson and Simon O'Keefe. Thank you for your knowledge, your time, and your steadfast resolve, that I could not have produced this work without. Thank you also, to my Industrial supervisor, Adrian Charlton from Fera, as well as Michael Dickinson and James Donarski, for their inspiration for this work and their patience in helping me understand the many intricacies of the matter. Thank you also to James Cussens and Gordon Allison, for their time taken to check and recheck my work. Thank you to Jobie Kirkwood, Alan Millard, Elmira Esmaeili, Frances Drachenberg, Durdane Kocacoban, Christopher Timperley, Matthew Dale, Penelope Faulkner and all at YCCSA, for the many teas, discussions and hours spent scribbling ideas on a whiteboard. Thank you especially, to my fiancée, Edda, and to my family; Cathy, Ted, Emma, Rob and Frank, for their continued love and support. And finally, a big thank you to all for making this whole thing quite enjoyable.

AUTHOR'S DECLARATION

The work contained in this thesis is original and has not been submitted at this or any other institution, with the exception of the following:

The data and experimental information in 3 were provided by Michael Dickinson, Fera Science Ltd. (*Medicago* and *Alopecurus* datasets) and James Donarski, Fera Science Ltd (Beef dataset).

Chapter 4 incorporates elements from:

> M. Rusilowicz, M. Dickinson, A. Charlton, S. O'Keefe, and J. Wilson, "A batch correction method for liquid chromatography–mass spectrometry data that does not depend on quality control samples," *Metabolomics*, vol. 12, pp. 1-11, 2016.

Chapter 2 incorporates elements from:

> M. Rusilowicz, S. O'Keefe, A. Charlton, and J. Wilson, "Chemometrics Applied to NMR Analysis," in *Encyclopedia of Analytical Chemistry*, ed: John Wiley & Sons, Ltd, 2014.

Chapters 5 and 6 incorporate elements from:

> M. Rusilowicz, M. Dickinson, A. Charlton, S. O'Keefe, and J. Wilson, "MetaboClust: Interactive software for metabolomic time-series analysis," *Unpublished*.

1. INTRODUCTION

Metabolites are the small molecules that result from metabolism, the chemical reactions that occur within, and essentially govern, a biological organism. There two fundamental categories of metabolites, *primary metabolites*, which are the compounds essential to life, growth and development, and *secondary metabolites*, which are not *de facto* necessary for life, but play additional, often important, roles [1]. Under these definitions the amino acids, used in protein synthesis, would largely be considered primary metabolites, whilst compounds such as nicotine and cocaine, which act to deter insects from certain plant species, are an example of secondary metabolites [2, 3].

It has been suggested that somewhere between 200,000 and 1,000,000 different metabolites exist in the plant kingdom alone [4, 5], although the exact source of this figure does remain somewhat vague. Whilst traditional research has focussed largely on the analysis of individual elements, the sheer number of metabolites shifts the ratio of known-to-unknowns heavily in favour of the unknowns. The implication of this is that in the analysis of a biological system as a whole, there is not only a vast amount of data available, but the majority of that data are unidentified in terms of both their name and function.

In the same way in which a life-form's *genome* can be considered the entire set of genes for that organism, the *metabolome* can be considered the entire set of metabolites present in an organism and is the primary focus of study in this thesis. The genome and the metabolome form two sides of a much larger system, joined by the *transcriptome*, representing the set of genes actually expressed, and the *proteome*, representing the proteins transcribed and ultimately catalysing the chemical reactions of the metabolome. The system is by no means feed-forward, the regulation of many genes for instance, can be controlled through feed-backward *post-transcriptional regulation* at the translation phase [6] and certain control mechanisms only become apparent "further down" the hierarchy at the metabolic level [7].

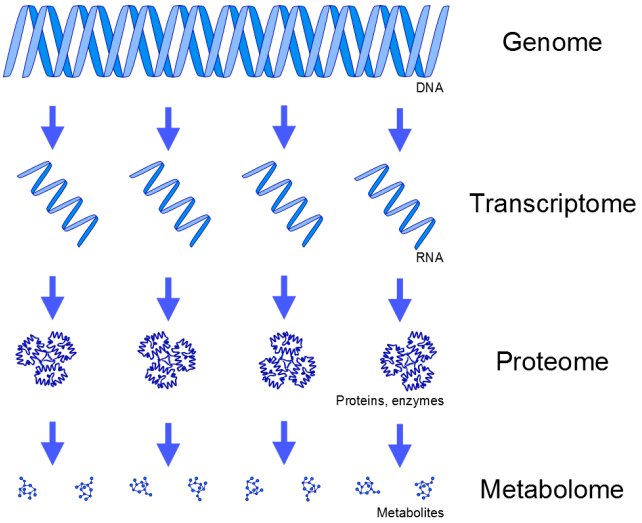


Fig. 1.1: Omics techniques used to gather information from an organism on a genome level scale.

Insights into one aspect of the system can therefore provide important information about the others. This all brings us to the field of Omics, which represents the use of bioinformatic statistical tools in the analysis of genomic, transcriptomic, proteomic and metabolomic data. The complete hierarchy is shown in Figure 1.1.

Modern high-throughput analyses have led to vast amounts of data being generated pertaining to organisms' genes and transcription, as well as the proteins and metabolites within their cells. Piecing back this information into a coherent map of regulatory processes and pathways that we can make use of is not a simple task. However, by doing so we can increase our understanding of fundamental biological mechanisms that have direct relevance to a large number of fields and research areas [8]. A great deal of study into metabolic pathways and mechanisms has already been performed, and there exist several databases with information on the coupling between one metabolite and another, the catalysing proteins and the gene sequence responsible.

In contrast to targeted "reductionist" studies, which seek to a deeper understanding of individual aspects of a system [9], large scale metabolomic analyses suffer from the fact that the identity of individual metabolic species may not be known up-front. In metabolomics, compounds can be

present at drastically different concentrations and there is no one method that can comprehensively identify all metabolites within an organism [10]. A number of techniques have therefore been developed to detect different ranges of metabolite identities and abundances. Two commonly applied methods of viewing the metabolome are NMR and chromatography-coupled MS. Chapter 2, presents an overview of these techniques and an up-to-date review of the statistical tools available in dealing with the metabolomic data obtained, ranging from data acquisition and preprocessing, through to identification and statistical analysis.

An important role of metabolomics is played in the field of agricultural biotechnology. The agricultural industry has many challenges, not limited to the growing population, changing climate, adaptation of pest species, counterfeit produce and environmental damage of certain farming practices. Metabolomics has many uses in both the assisting in the understanding of biological processes, such as the impact of genetic modification [11], as well the as identification of traits, such as source-species, in wine [12] or coffee [13].

Chapter 3 will introduce three datasets with a basis in agriculture and serves to highlights some of the issues encountered in metabolomics. The first dataset concerns drought and disease stress in legumes. Legumes are an important agricultural crop which act as both a human and a livestock food-source and are particularly sensitive to the effects of drought. Drought-stress resistance is impaired by coincident infection with the *Fusarium* pathogen. Here leaf and root samples from *Medicago truncatula* are investigated using LC-MS analysis. Several points of concern common to metabolic datasets are highlighted and discussed. In particular there exists a large degree of noise in the dataset, due to LC-MS "batch" effects, as well as natural biological variation. A number of "age-dependent" metabolites, whose concentration fluctuates regardless of the plants condition are also present. Finally a large number of metabolites are affected by stress, and a simple method of exploring these is called for.

The second dataset concerns *Alopecurus myosuroides*, a common agricultural weed affecting grain crops. A number of varieties of *Alopecurus* have developed resistance to one or many modern herbicides. The investigation of metabolites' presence in plant samples may provide key biomarkers for herbicide resistance, as well as potentially elucidating resistance mechan-

isms. This dataset is again characteristic of non-targeted LC-MS data and serves to reiterate several of the issues encountered with *Medicago*.

In the analysis of these two datasets it is found, in particular, that an experimental drift over time is exhibited, an unintended but common artefact encountered in LC-MS procedures. Standard correction methods using quality control samples here manifest in being sub-optimal in mitigating this drift. This sets the precedent for the Chapter 4, which presents a novel drift correction method for LC-MS studies. A rigorous evaluation contrasting different methods of correction is then performed using the available datasets.

High throughput analyses are able to produce larger volumes of information than can be manually analysed in a reasonable time-frame. A certain amount of reliance is therefore placed on chemometric methods in order to make sense of the data. That there is "no free lunch" in data analysis is a fact which has been stated many times. It is one however, often presented as a disclaimer rather than as a problem to be tackled. In the course of the analyses described here, it becomes apparent that the choice of algorithms and parameters is largely subjective. The typical workflow applied to metabolomic studies has been well documented and can be described as a waterfall model, whereby each stage holds the previous in requisite.

Chapter 6 seeks to challenge this model, presenting an interactive software suite designed to open up the metabolic workflow and present the user with a means of rapidly changing the parameters of the previous stages of analysis. A deterministic variety of k-means++ is presented and implemented, which is suited to the manual exploration of data. Visual methods of comparing clustered data with known pathway information are described, implemented and demonstrated using the *Medicago* and *Alopecurus* datasets.

The third and final dataset presents a NMR study of "28-day matured beef". From the dataset it is postulated that the 28-day maturation process at a safe (cool) temperature can be emulated via a much shorter maturation in a warm environment. This presents a potential classification problem, identifying not only the storage-conditions of the beef during its maturation period, but also discovering potential biomarkers which could be useful in a field-test for mislabelled product.

The basic analysis of the Beef dataset indicates that whilst reasonable

separation between groups can be achieved, there exists a certain overlap between groups which cannot be easily resolved. Furthermore, no concise set of biomarkers responsible for the separation are present. Chapter 7 explores the use of strongly-typed genetic programming (ST-GP) in the analysis of the beef data. In contrast to other techniques GP offers a method of classification based on the selection of a small subset of spectral features. The addition of strong typing permits different types of data to be combined in the decision tree, offering a method of determining both storage condition and age from a single set of biomarkers.

2. CHEMOMETRICS APPLIED TO LC-MS AND NMR ANALYSIS

2.1 Introduction

Two of the foremost modern analytical chemical techniques include liquid chromatography–mass spectrometry and nuclear magnetic resonance spectroscopy, each of which is capable of providing a large amount of information about a chemical sample.

2.1.1 Mass spectrometry

Mass spectrometry (MS) has its roots in mass *spectroscopy*, whereby charged ion rays were directed onto a photographic plate for viewing and the basic principal remains unchanged. When changed particles in their gaseous state are exposed to a magnetic field they are deflected by differing amounts, dependant upon the mass and electronic charge of the particle. In MS the sample is ionised and passed through an analyser, which uses a magnetic field to separate the ionised particles by their mass-to-charge ratio, or m/z. The particles are then detected by a detector, which records their quantity. This produces a spectrum detailing the m/z values against abundance.

Modern ionisation techniques, such as electrospray ionisation (ESI) [14] allow the introduction of liquid samples and mass separation and detection techniques such as quadrupole [15] and orbitrap [16] allow for high resolution and acquisition times. Other techniques such as Fourier transform ion cyclotron resonance (FTICR) [17] permit detection and mass separation to occur concurrently.

2.1.2 Liquid chromatography

Chromatographic techniques aim to separate the chemical species within a sample based on the chemicals' rate of movement through a medium. Liquid chromatography (LC) developed upon the theories developed in gas chromatography (GC) [18], with the adaptation to support two liquid phases

[19]. Unlike GC this allowed the separation of peptides, previously difficult or impossible due to their breakdown under the high temperatures required to maintain the gaseous state. In LC the sample is combined with a solvent – the mobile phase – and passed through a column containing an adsorbent material – the fixed phase. Due to the differing interactions between the individual chemical species in the sample and the two phases, the chemicals pass through the column at different rates.

The discovery of LC later led to the development of high performance liquid chromatography (HPLC), or high-pressure liquid chromatography. By reducing the particle size [20–22], higher pressures could be obtained which permitted significantly faster elution times and experimental throughput.

2.1.3 LC-MS

Liquid chromatography separates compounds by retention time, t_r, whilst mass spectrometry separates compounds by the mass-to-charge ratio, m/z. The mass spectrometer is also responsible for detection, typically recording a measurement in the form of an induced charge, abundance or *intensity*. The combination of LC and MS (LC-MS) produces a three dimensional spectrum of $rt \times m/z \times intensity$. LC-MS is one of the foremost tools in metabolomic studies. It is particularly advantageous as an analytical tool due to its high analytical sensitivity and the ability is able to separate out and provide information along two axes on thousands of compounds in a single analytical run. LC-MS does however suffer from lower specificity and reproducibility than other techniques, such as nuclear magnetic resonance (NMR), with some compounds being "missed" due to ineffective ionisation (MS) or rapid elution from the column (LC), and differences in retention times being seen in different analyses of the same mixtures.

2.1.4 Nuclear magnetic resonance spectroscopy

NMR spectroscopy has a long history reaching back to the first successful experiments in the 1940s, with applications in chemistry, biochemistry, physics, and medicine [23–25]. NMR spectroscopy provides information about the physical and chemical properties of atoms within a sample by exploiting the fact that atomic nuclei absorb and emit electromagnetic radiation in the presence of a magnetic field at frequencies characteristic of the nuc-

leus' chemical environment. Biological applications of NMR spectroscopy date back to 1957 when the first ^{1}H NMR spectrum of a protein was published [26]. In the early 1970s, the technique was first used to characterize metabolism and has since become an essential tool in metabolomics [27, 28].

Although less sensitive than MS, NMR spectroscopy gives greater coverage in comparison to other techniques used for the analysis of complex mixtures, providing information about all metabolites with concentrations above the limit of detection [29]. The reproducibility of NMR provides substantial benefits in compound identification and quantitative analysis, and recent advances in the technology, including higher field magnets and cryogenically cooled probes that increase signal-to-noise ratios (SNR), mean that sensitivity is becoming less of an issue. Furthermore, NMR is a non-destructive technique and requires very little sample preparation, enabling the analysis of samples in a chemical environment similar to that in which they are naturally found [30].

2.1.5 NMR varieties

NMR spectroscopy can provide characteristic profiles of the metabolites present in a sample that can be used as fingerprints in pattern recognition procedures. By far the most common NMR technique used in metabolomic studies is the 1D proton NMR or ^{1}H NMR experiment, specific to hydrogen-1 nuclei, which is the nucleus with the highest receptivity. One-dimensional experiments are the most sensitive techniques and can be fast and simple to perform. 1D NMR has been used in numerous profiling studies, but many different types of 2- and 3-D NMR experiment also exist, providing information about chemical shifts, J-couplings, and diffusion coefficients. Although multidimensional techniques have the potential to provide more detailed information, this comes at the expense of significantly greater acquisition time. These methods are often used to aid the identification of compounds, however, new developments are making metabolite fingerprinting by 2D NMR a realistic aim. Additionally nuclei other than ^{1}H can be explored, with ^{13}C, ^{31}P, and ^{15}N being popular. A more detailed description of 2D and 3D NMR, as well as the use of other nuclei, can be found in the complementary paper [31].

2.1.6 Analysis

Interpretation of individual spectra can provide structural information about particular compounds, but statistical techniques are required to extract useful information from multiple spectra. Technological developments have led to a significant increase in the amount of data generated, with improved methods providing greater spectra resolution and sensitivity. New technology, advances in methodology, and increased computer power have increased the need to develop mathematical and statistical methods to analyse and interpret the complex datasets acquired. This has resulted in developments in the field of chemical informatics known as *chemometrics*.

Some of the statistical tools commonly used have been available for many years. For example, principal components analysis (PCA) was devised well over 100 years ago [32] but was originally restricted to two or three variables owing to the complex calculations involved. In 1933, Hotelling [33] published practical computing methods, although these could not be realized until the advent of the modern computer. The early use of statistics could only cope with 'long and thin' data matrices, i.e. a few variables for each of many observations, where as modern analytical techniques typically record many variables for few observations, giving 'short and fat' data matrices. The first chemometric studies were performed in the early 1970s [34, 35] when pattern recognition software was developed to determine chemical structure from simple NMR spectra [36]. Since then, and particularly over the past two decades, increased computational power has led to such tools being readily accessible on the desktop computer. NMR and LC-MS chemometric methods are now used in a wide range of applications including clinical diagnostics [37–40], toxicology [41–43], food science and traceability [44–49], monitoring genetic modification [11, 50, 51], predicting side effects of pharmaceuticals [52, 53] and their environmental effects [54], and process control [55, 56].

Targeted approaches in metabolomics seek quantitative information about specific compounds. NMR is advantageous in that the relative concentration of a chemical is directly proportional to the integrated peak area. After calibration to an internal standard, compounds within a sample spectrum can be identified and quantified by comparison with reference spectra from pure chemicals. Statistical analysis can then be used to interpret the results. Obtaining concentrations from LC-MS follows similar principles, though due to

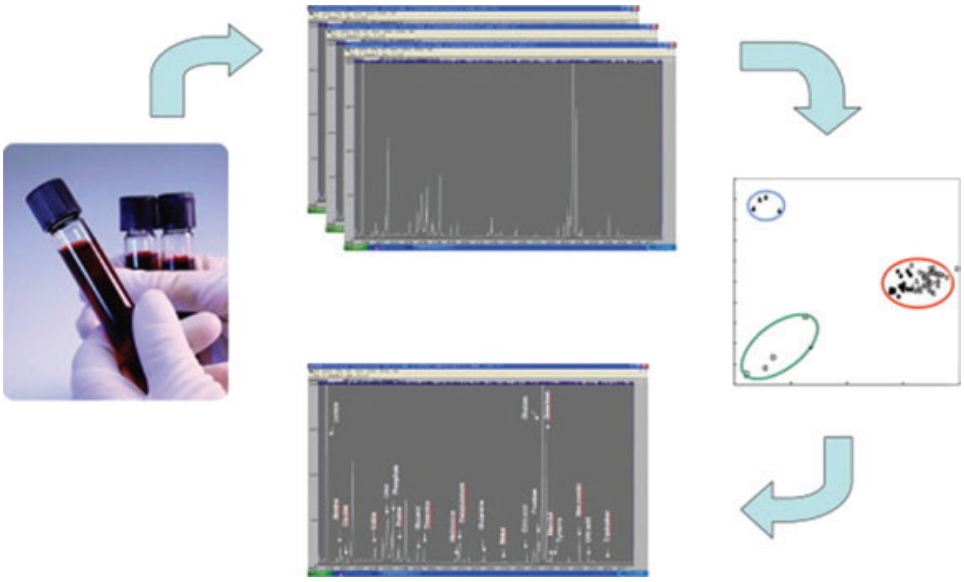
matrix effects – the interaction of the sample with the solvent – and differing ionisation potentials the relationship between the instrumental response and chemical concentration is not as straightforward [57], as discussed in Chapter 4.

In contrast to targeted approaches, non-targeted, or chemometric, approaches do not initially attempt to identify particular metabolites. Instead, statistical techniques and pattern recognition methods are used to identify spectral features that show consistent trends or provide discrimination between classes. These features can provide a fingerprint for a metabolic state to be used, for example, for disease diagnostics, and individual features of interest can then be related to specific compounds and metabolic pathways using database searches [58]. Figure 2.1 illustrates the difference between the chemometric and quantitative approaches to metabolomics.

The traditional reductionist approach in biological research [9] divides the overall system into successively smaller components and investigates the effects of each on the biological system. The resources required for testing and modelling individual components put constraints on the exploratory research that can be conducted and imposes a reliance on prior knowledge. Non-targeted metabolomics offers a more holistic approach, providing a snapshot of an organism's metabolite composition that can be used to characterize phenotypes in a high-throughput analysis. A recent method termed targeted profiling attempts to combine aspects of targeted and non-targeted methods [59]. Experimental spectra are compared with a database of known metabolites to provide a more immediate source of information in terms of metabolites and their concentrations rather than a peak list.

2.2 Data preprocessing

Changes in the experimental environment, sample preparation, instrumental variation, and background noise can lead to unintended differences between observations. For successful statistical analysis, sources of variation should be controlled wherever possible and multiple spectra should be processed in the same manner to prevent additional variation being introduced. Preprocessing steps, applied in either the time domain or the frequency domain, can reduce the impact of unwanted artefacts. Methods include noise removal and baseline correction, the alignment of peaks to account for drift in peak



(a) Metabolite identification

(b) Metabolite identification

Fig. 2.1: (a) Schematic illustration of the chemometric approach to metabolomics. In this example, the spectra obtained from multiple blood samples are processed using principal component analysis. After identifying significant differences, the most informative peaks in the spectra are identified using a variety of methods. (b) Schematic illustration of the quantitative approach to metabolomics. In this example, the biofluid spectrum is annotated and the compounds in the sample are identified and quantified. This information is then used to perform multivariate statistical analysis allowing the most important biomarkers and pathways to be identified. (Reproduced from Ref. [27]. Copyright 2008, Elsevier.)

positions between spectra, as well as normalization and scaling, which can account for variation in the concentration of diluted samples and prevent large peaks dominating the analysis. The aim is to prepare the data for pattern recognition procedures and the term "pattern vector formation", coined in an early chemometric study [37], provides a good description.

2.2.1 Water removal

Water, abundant in biological systems, and typically used as an analytical solvent, contains ^{1}H, which can interfere with and obscure the intended target of analysis. In NMR this leads to a large water peak which can dominate the rest of the spectrum. A variety of techniques have been proposed to reduce the effects of this peak, including presaturation, water suppression by gradient tailored excitation (WATERGATE) [60], WET [61], and excitation sculpting [62], each with their own advantages and disadvantages. Perhaps the simplest approach however is simply to truncate the interfering water region at around 4.77ppm [63], depending on the chemical environment, during the data preprocessing stage.

Due to a second degree of separation along the m/z axis the solvent is less of an issue in LC-MS, however, early column effluent, which may contain a complex mixture of compounds is often discarded to simplify the dataset and protect the mass spectrometer from contamination [64, 65]).

2.2.2 Apodisation

The observed (time domain) NMR signal generated by the oscillations in the radiofrequency detection coil is referred to as the free induction decay (FID). Various mathematical functions can be applied to the FID before Fourier transformation and can dramatically increase the quality of the spectra obtained [66]. These *apodisation* or *window* functions weight the decay in the time domain in order to maximize the signal-to-noise ratio (SNR) in the frequency domain. The line broadening introduced by apodisation can accommodate minor peak shifts, but can introduce problems with overlapping peaks, particularly with complex mixtures. There is generally a tradeoff between noise reduction and resolution, although the application of separate functions for the real and imaginary parts of the FID has been proposed as a method to improve both the SNR and the resolution [66]. An investig-

ation of various apodisation functions in ultrafast 2D NMR also found that sensitivity could be improved without significantly compromising resolution [67]. The study found a Gaussian function to be particularly effective at removing the distortions present in ultrafast spectra. The choice of function can affect the results and the process of apodisation may require optimization.

Whilst far more commonly seen in NMR studies, apodisation of the FID is also applicable to studies involving Fourier transform ion cyclotron resonance mass spectrometry (FTICR-MS). For example the software application presented in [68] allows the comparison of raw LC-MS spectra and uses an apodisation function prior to the the Fourier transform of the FID. Zhang et al. have provided a detailed analysis of the effects different apodisation functions on the FTICR-MS spectra of complex petroleum mixtures, noting that apodisation, like for NMR, increased resolving power can be obtained at the cost of lower dynamic range. Results are presented for a number of window functions in the original paper [69].

2.2.3 Denoising

Thermal interference, magnetic field variations, instrumental instability, electrical interference and the digitization of analogue signals can all contribute to the presence of noise in both LC-MS and NMR spectra. Recent advances, such as cryogenically cooled probes in NMR, higher column pressures in LC and more advanced mass separation and detection in MS have helped to reduce electronic noise, however most analytical instruments produce a signal even in the absence of an analyte [70]. Certain optimisations, such as removal of the asymmetric 'sinc wiggles' that can appear owing to truncation of the FID can be performed [71], and the SNR can be improved by increasing the number of scans, which should in theory incrementally reduce the noise to zero. In practice however, random noise can be reduced, but not eliminated owing to technical limitations and only a finite number of scans being possible.

The point at which real-valued data falls below the noise level is generally referred to as the limit of detection (LOD) [70]. Very small random fluctuations have been shown to cause significant variation in the clustering of otherwise identical spectra [72]. A careful choice of threshold, below which the signal is set to zero, was shown to alleviate the problem in this

case and produced almost perfect clustering in PCA. The choice of threshold necessarily changes the variance between samples and this simple form of noise reduction requires judgement to optimally reduce noise while retaining real information.

The wavelet transform is another widely used technique that can be used to reduce spectral noise [73, 74]. The spectra are represented by wavelet functions, or wavelets, that are localized in both position (or time) and frequency and the wavelet decomposition allows features to be considered at different scales. Denoising is achieved by thresholding under the assumption that the highest frequencies correspond to noise. In global thresholding, a single threshold determined from the highest frequency wavelet coefficients is applied globally to all wavelet coefficients, whereas level-dependent thresholding uses a different value for each wavelet level [75].

The two dimensional nature of LC-MS allows some additional scope for noise detection, and therefore removal. For instance, a denoising algorithm – matched filtration with experimental noise determination (MEND) – using a *matched filter* to locate a vacant extracted-ion chromatogram (EIC) – a trace along a particular m/z – has been suggested. After locating a matching EICs the power-density spectrum for the noise can be calculated and used to reduce the noise in the dataset [76]. A similar process involves the use of a clustering method – density-based spatial clustering of applications with noise (DBSCAN) – to identify noise by the elimination of features (not-noise).

2.2.4 Baseline correction in NMR

Inappropriate sampling, phase correction, and the application of filters to the FID can all contribute toward baseline distortion in NMR spectroscopy and can be a major source of error in quantitative analysis [77]. Although most can be avoided by setting the experimental parameters correctly, some baseline offset is likely to remain. Furthermore, correction methods that require manual tuning of parameters are prone to variation between users and can introduce bias in quantification. Many automated baseline correction algorithms have been proposed for both 1D and 2D experiments. Errors in the first point of the FID contribute to a constant baseline offset, and while this can be easily corrected, errors in the sampling of the next few points contribute to a baseline 'roll' that is significantly harder to resolve

[78]. A variety of baseline correction algorithms has been developed both within the time-domain, such as linear prediction to reconstruct the first few points of the FID [79], and in the frequency domain, involving approximation and subtraction of the baseline. Accurate recognition of the baseline is necessary for it to be modelled and filters are often applied to ensure the estimated baseline passes through the centre of the data in noise regions, but does not follow peaks [80]. Dietrich et al. [81] used a standard numeric derivative for baseline recognition after smoothing with a moving average filter. This can lead to small peaks being smoothed out, but more advanced smoothing algorithms significantly increase computer time and can be prohibitively slow with 2D spectra. However, the continuous wavelet transform (CWT) provides good baseline smoothing without the computational expense of other algorithms [82].

Other methods fit the baseline in noise-only regions and then construct a final baseline by connecting these fragments by straight lines [83]. Such algorithms can have problems in signal-dense spectra such as those from complex mixtures, destroying the line shape of high intensity peaks. To overcome this, Chang et al. [84] combined the identification of signals using a high pass filter with Lorentzian line-shape modelling. An alternative method, shown to be effective with crowded spectra, fits a curve to the lowest intensities using a penalized smoothing algorithm and does not require differentiation between noise and signal [85].

New baseline correction algorithms continue to be developed, often combining or extending aspects of earlier methods. A combination of three baseline recognition algorithms is used to provide well-recognized baseline points from noise regions as well as 'quasi-baseline points' in the signal-dense regions [86]. The sets of points are used in an iterative algorithm to provide baseline correction that avoids the negative regions sometimes produced by other methods. Recent algorithms are designed to handle spectra with large numbers of peaks effectively and are therefore suitable for application to the spectra of complex mixtures in metabolomics studies.

2.2.5 Alignment

In the case of NMR, chemical shifts result from nuclear spin transitions occurring in magnetic fields. Chemical structure and molecular interactions can change the chemical shift of a nucleus and the resulting spec-

tral differences provide useful information and are the fundamental reason for the use of NMR. However, changes in experimental conditions, such as differences in temperature, pH, and ionic strength, lead to undesirable chemical shift variation [87]. Shifts are perhaps even more commonplace in chromatography-coupled MS techniques, where changes to the environment and contamination of the column over time alter the analytes' affinity with the sorbent surface and thus affect the elution time. Furthermore, the masses recorded by the spectrometer are themselves not immune to shifting [88], creating a shift along both m/z and t axes for chromatography coupled mass spectrometry (XC-MS) spectra. While rigorous sample preparation and experimental protocols allow environmental factors, such as temperature and pH, to be regulated, some variation in peak positions will remain. Some column contamination is inevitable in LC-MS and the ionic strength of an NMR sample cannot easily be controlled. Uncorrected peak shifts have been shown to significantly affect multivariate analyses such as PCA [89] and partial least squares discriminant analysis (PLS-DA) [90].

Two common procedures for automated peak alignment of both NMR and LC-MS data include dynamic time warping (DTW) [91], originally used in speech processing, and covariance-optimized warping (COW) – also called correlation optimized time warping [92]. Both are pairwise alignment methods and require a target spectrum to be chosen as the reference to which spectra are to be matched. DTW uses distance as a similarity measure between two signals and is sensitive to differences in peak intensities [93], which led to the use of the correlation coefficient in COW. COW performs a piecewise alignment of data segments. However this is computationally expensive and can be sensitive to baseline distortion. To remedy this, a prior peak-picking algorithm stage has been suggested as an alignment focal point which can provide an effective and computationally less intensive alternative [94]. An approach requiring less manual intervention, the component detection algorithm (CODA) has also been proposed to select areas of less noise as the focus of the alignment [95]. Improvements over COW have also been suggested. For instance a Bayesian approach, which simultaneously corrects the baseline, has been demonstrated by the authors to outperform both DTW and COW in terms of both correlation between spectra as well as execution times [96].

As the direction in which individual peaks in spectra shift is disparate,

local alignment procedures have been developed in which small regions of the spectrum, containing at least one peak, are aligned individually. The alignment of 1D spectra, such as those originating from ^{1}H NMR are simpler – though by no means simple – to align. The partial linear fit (PLF) algorithm [97] fits close peaks together in order to keep multiplets together, but can have problems with biological data [98]. Peak alignment using a genetic algorithm (PAGA) optimizes the fit of each segment using the correlation coefficient [99]. A segment size parameter needs to be specified and some peaks can remain unaligned if the segments are too large or are split at boundaries with small segments. The use of fast Fourier transforms for rapid computation of cross-correlation improves the computational efficiency of the algorithm, making it suitable for use with very large datasets [100]. Segment sizes are refined using the recursive segment-wise peak alignment (RSPA) method of Veselkov et al. [90]. The method has been shown to accommodate smaller peaks, as well as large peaks, better than other widely applied alignment methods. Interval-correlation-shifting (icoShift) is a similar segment-based algorithm [101] that can be combined with existing methods such as RSPA and allows interactive segment selection. Figure 2.2 illustrates the alignment provided by the icoShift algorithm. Most alignment methods rely on the choice of a suitable reference spectrum, which can be that of a specified sample or calculated from multiple spectra, for example, the average over all spectra to be processed. Variable reference alignment allows the prerequisite for a common reference spectrum to be relaxed as multiple spectra can be used as segment-specific target spectra. [98]. The method identifies regions that are 'most similar', providing segment boundaries that can be input to procedures such as icoShift. A method specific to LC-MS, which avoids the selection of a reference spectrum, estimates the m/z and rt values and performs alignment based on this information [102].

Issues with peak shifts between spectra still exist, and Vu and Laukens [103] provide a comparison of approaches to peak alignment. However, it has been shown that metabolically relevant information may be present in the peak shifts and that modelling these in pattern recognition gives only slight reduction in the prediction ability of the model [104]. Giskeødegårda et al. [105] also showed that useful class information can be extracted from both intensities and peak shifts.

It has also been suggested that the global information within the full

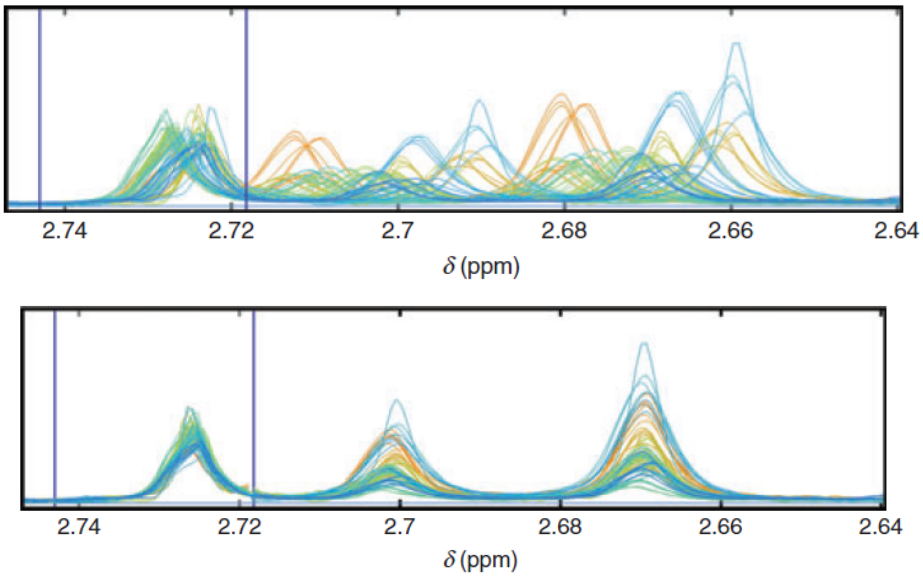


Fig. 2.2: (a, b) Alignment of NMR spectra with the icoShift algorithm, using user-defined intervals. (Reproduced from Ref. [101]. Copyright 2010, Elsevier.)

set of spectra can be useful to the alignment of local peaks. For instance a Bayesian model which also estimates *rt* variability is able to make use of complementary information of the entire spectrum for alignment [106], although internal standards are required in order to predict the prior probabilities in the dataset used. Another method suggested has been to use supplementary information from tandem MS to identify the most certain peaks, using nonlinear robust ridge regression to establish correspondence [107].

Warping methods (i.e. those that shift the dimensional planes) have however been criticised. The warping affects the spectra at a systematic level and thus cannot correct changes at the component level [108], for instance where shifts are unique to a particular chemical species. Correspondence algorithms have been suggested as a better alternative, and do not distort the original data. However they are computationally complex and are more sensitive to variation in the parameters, requiring additional optimisation.

2.2.6 Binning

As an alternative to alignment, unwanted chemical shift variance can be dealt with by binning the data, a method that can also account for variations in peak width. In its simplest form, the spectrum is divided into equally spaced bins over which the intensities are summed [109]. The method, also known as *bucketing*, removes the effects of small shifts by reducing the resolution of the data and thereby decreases computational cost and simplifies further analysis. However, the reduced variables do not necessarily correspond to peaks, making the interpretation of multivariate analyses more difficult. Furthermore, peaks can be split by bin boundaries creating additional variation and information can be lost owing to multiple peaks being assigned to the same bin [110]. Custom bins have been used to improve results, but this is a time consuming and somewhat subjective process, requiring a degree of knowledge about the peaks in the spectra [59].

Nonequidistant or adaptive binning can also be achieved in automated procedures [111]. This approach requires a reference spectrum to be created by taking either average or maximal intensities over all spectra. This reference spectrum is then smoothed using a non-decimated wavelet transform before the minima are used to determine bin boundaries. The method significantly reduces intraclass variation in comparison to uniform fixed-length binning and allows regions containing only noise to be identified and excluded from further analysis. The AI (adaptive intelligent) binning algorithm also uses variable bin sizes, but avoids the need for a reference spectrum by determining the bins in an iterative procedure [112]. Initially a single bin covers the full spectral width, which is divided optimally to produce two new bins. The process is repeated, recursively identifying new bins by subdivision of existing bins, with a metric used to accept or reject split bins. The AI binning algorithm is shown to outperform uniform (0.04 ppm) binning and use of the full spectra in terms of predictive accuracy.

An alternative way to avoid the problem of split peaks at bin boundaries is to allow overlapping bins. A binning technique has been proposed that uses Gaussian functions to weight the contribution of peaks according to their distance to the bin centres [113]. The method is shown to be robust to peak shifts, while retaining the information required for classification and multivariate analysis.

An interesting comparison between spectral binning and wavelet denois-

ing showed that PCA performed directly on the wavelet coefficients gave better results than PCA on binned data when applied to a series of 2D NMR spectra of proteins with different ligands present [114]. The wavelet-PCA scheme is also found to detect outliers better, although this may be due to shifts in frequencies bridging bin boundaries rather than the effects of noise as standard equal length bins were used.

In the context of LC-MS, binning can be performed along individual EICs, through the use of standard 1D binning methods. However binning methods for use on 2D have also been proposed. The basic procedure is outlined in [115], and involves drawing "areas" around identified peaks. As for 1D binning drawbacks of this method have been noted. In particular peaks may be split between different bins or multiple peaks may be combined into the same bin [116]. The same is also applicable to 2D NMR. For instance, the freely available software, rNMR [117], allows visualization of multiple spectra and the rectangular regions of interest (ROI) can be considered 2D bins.

2.2.7 Feature Extraction

Feature extraction involves the determination of values representing particular aspects of interest, such as the peaks of single compounds. The line between binning and feature extraction is not always apparent. Non-equidistant binning, for instance, could also be viewed as feature extraction, rather than noise removal if the bins directly relate to peaks.

The Lorentzian spectrum reconstruction algorithm deconvolutes a 1D spectrum into a series of Lorentzian functions [118]. The method first identifies peaks from the local minima in the second derivative so that peaks appearing as shoulders as well as simple maxima are recognized. The parameters of the overlapping Lorentzian functions are then estimated to provide a model for the spectrum that facilitates spectral assignments. A similar feature extraction method has been implemented for 2D spectra. The Lorentzian-like properties of NMR peaks are particularly suited to this and are exploited to fit peaks in ^{1}H–^{13}C HSQC NMR in a 2D adaptive binning algorithm [119]. A modified Lorentzian is used to model the peaks in a reference spectrum obtained as the median over all spectra to be processed. The footprints of these modelled peaks form elliptical bins that can be applied to the spectra in the dataset to provide variables for subsequent

multivariate analysis. This method has been applied to the analysis of ^{1}H-^{13}C HSQC spectra from rat brain tissue after intraperitoneal injection with [U–^{13}C]- glucose and from those injected with normal ^{12}C–glucose. Here, cross-referencing with the ^{13}C and ^{1}H chemical shift correlations obtained for just 16 standard metabolites allowed 39 of the 105 peaks used in the analysis to be associated with a known metabolite.

Xi et al. [120, 121] describe a method to assist the identification of metabolites in a sample using a database of COSY and HSQC spectra from known metabolites. Matches to the reference compounds are scored, allowing for possible displacement of the peaks. The method has been shown to be effective with defined mixtures of amino acids and complex biological samples. The same authors have also demonstrated an automated peak identification protocol for HSQC spectra. Although applied for the quantification of known metabolites, the potential for use in metabolomics is recognized.

Unlike binning methods that reduce the resolution of spectra and lose information when overlapping peaks are incorporated into the same bin, deconvolution can extract relevant information for each peak individually or fit the spectral signatures (i.e. the set of peaks) obtained for reference compounds. While integration over bins leads to errors in quantification owing to overlapping peaks and is sensitive to baseline distortions, peak shapes can still be fitted and the concentration of the compound determined with reasonable accuracy [122].

Weljie et al. [59] fit a combination of Lorentzian peak shape models from a database of known metabolites to the target spectrum and provide quantification by comparison with an internal standard. The method, termed *targeted profiling*, was compared to spectral binning using 45 custom bins representing 11 compounds. The bins were manually adjusted to ensure peaks and peak clusters occurred in the same bin for all spectra and that peaks were not split between bins. They found the binned data to be more sensitive to noise and artefacts related to water suppression and baseline deviations. Results from PCA were found to be more consistent when using targeted profiling and binning was found to be particularly ineffective with low-concentration compounds. Although targeted profiling greatly simplifies interpretation of the results, the method is currently unrealistic for non-targeted studies. With estimates of over 100,000 metabolites in humans

alone, comprehensive profiling this way is not possible and methods that can allow as yet unknown resonances to be considered in analyses are required.

Feature extraction in LC-MS, like binning, can be performed along individual EICs. Use can be made of individual isotope patterns which can help to locate specific peaks [123]. MEND, also used for denoising, identifies peak by a gaussian peak shape combined with a matched filter [76]. A number of software packages using use both free and proprietary algorithms have been developed for the task of identifying LC-MS peaks, including XCMS [124], MZMine [125], MetAlign [126] and Progenesis QI [127].

2.2.8 Normalization

Normalization is used to correct for differences in overall concentration in order to make samples comparable with each other. In addition to variation due to the amount of material in the samples, differences in factors such as pulse calibration can cause inter-sample variation. If the data matrix is arranged so that each row represents a spectrum with the variables (data points, peaks, or metabolite concentrations) in columns, then normalization is a row operation, in which each spectrum is multiplied by a constant. This constant is often determined using an internal standard, a fixed volume of which is added to each sample. Calibration standards should not interact with the sample and have a resonance that does not overlap with those of the sample. Internal standards however can be expensive [128] and alternatively the spectral intensities can be adjusted to a reference peak within the sample, which is expected to be the same for all samples, such as creatinine in urine [129]. A simple form of normalization, termed *integral normalization* or *normalization to constant sum*, ensures that the sum over all intensities is the same for each spectrum [130]. Other, more complex normalization procedures, such as probabilistic quotient normalization [131], have been developed to overcome the artefacts introduced as the most abundant metabolites affect the scaling of all metabolites. The scaling constant for each spectrum is calculated by considering the distribution of the quotients of intensities in comparison to those of a reference spectrum. Alternative methods include histogram matching [132], adapted from methods used in image processing, where the histogram is obtained as the number of signals within each intensity range, and group aggregating normalization [133], in which samples are normalized so that they cluster close to their group centres in PCA.

2.2.9 Batch correction

As noted earlier, LC-MS can suffer from relatively poor reproducibility. Shifts can not only occur along m/z and t axes, requiring complex alignment, but changes to *intensity* also occur. Proper care of the analytical run, including cleaning, conditioning and calibration can reduce, but not eliminate this issue [134]. Samples are often run in batches, interspersed with the relevant cleaning and conditioning events. However, this can lead to other sources of technical variation, such as differences in the operating conditions under which the acquisitions of the individual batches are performed. These issues are often remedied in the data-preprocessing stage, inclusion of identical quality control (QC) samples provide a target by which the shift in intensity can be monitored, and therefore corrected. A more detailed overview of current practices and problems is given in Chapter 4.

2.2.10 Variable Scaling

As larger peaks naturally have greater variance than small peaks [130], variance-based methods such as PCA can be dominated by high abundance metabolites. Similarly, large variables will contribute more to distance metrics in cluster analysis. Mean-centring is a column operation in which the mean value is subtracted to give each variable a zero mean. This removes the offset between high and low abundance metabolites, but does not change the variance, and is used in combination with variable scaling techniques [135]. For example, autoscaling, widely used in metabolomics, involves dividing the mean-centred variable by the standard deviation, giving each variable unit variance (uv-scaling). Variable scaling allows all variables to have equal influence in multivariate analysis, but can also scale up unwanted noise peaks. Pareto Scaling, in which the standard deviation is replaced by its square root, is frequently used as a less intensive scaling method as larger variables are reduced more than smaller variables. Vast (variable stability) scaling weights down the influence of variables with greater variance, giving greater influence to variables that change less [136]. Other scaling techniques include range scaling, which uses the range over which a variable is observed as a scaling factor, and level scaling that uses the mean.

In a comparison of scaling techniques, range and autoscaling were found to give the most biologically sensible results in PCA [135]. Other methods

were found to be too dependent on the mean or the fold change, and vast scaling led to results that were difficult to interpret. Purohit et al. [137] found that a generalized log transformation of the variables produced more normally distributed data that was more suitable for multivariate analysis. An additional parameter was introduced into this so-called Glog transformation to reduce the scaling of noise and shown to improve classification results [138]. In comparison to autoscaling and Pareto scaling with NMR datasets, Glog transformation is shown to give consistently better results. All scaling techniques were found to give better separation in PCA-LDA in comparison to unscaled data. However, results vary between studies and there is no one-size-fits-all scaling technique.

2.2.11 Workflow

The choice of preprocessing techniques depends on the data, the focus of the investigation, and the analysis methods to be used, and a single approach will not be appropriate for all cases even within a particular technique [129]. Preprocessing steps may be applied to the raw (time-domain) data, if available, to improve the quality of the FID and hence the corresponding spectra, or to the Fourier transformed (frequency-domain) data to directly correct for artefacts in the spectra. The aim is always to reduce intersample variation owing to effects unrelated to the biology and hence increase interpretability of the results.

Preprocessing steps are normally performed before multivariate analyses in order to obtain the best results from such methods. However, PCA is a multivariate method used widely for data visualization and has been exploited within preprocessing methods. For example, scaling constants for normalization have been determined by minimizing the Euclidean distance between each spectrum and its experimental group centre in PCA space [133], and Stoyanova et al. [139] have used PCA to detect regions requiring alignment.

The results of some techniques may depend on prior procedures, for example, peak alignment may perform better after baseline correction [93] and, while there is a predefined order to most techniques, this is not always the case. For example, data can be denoised before binning or afterwards by discarding noisy bins.

2.3 Data Analysis

The analysis of complex mixtures requires the use of statistical methods in order to extract meaningful information. This usually involves multivariate techniques, but univariate methods can also identify individual variables that differ significantly between groups and can be particularly useful in targeted studies.

2.3.1 Univariate Approaches

Univariate analyses can be applied to each variable in the spectra separately. Student's t-test allows a hypothesis to be tested by comparing a test statistic with Student's t distribution and can be used, for example, to determine whether two groups (populations) are significantly different from each other. The t-test makes certain assumptions about the data, notably that the data for the two populations follow a normal distribution. The t-test also assumes that the groups are representative of a random sample of the population, and that the data follow a continuous or ordinal scale. Despite these assumptions it has been noted that the test itself is robust to all but large deviations from these [140]. Whilst the t-test can be used for the two-group scenario, when several groups are involved Analysis of variance (ANOVA) is used to test differences in means. Assumptions of ANOVA follow those of the t-test: that data follow a normal distribution and that variances for all groups are similar. With so many variables, the probability of finding what appear to be significant differences between groups increases dramatically and has led to exploratory analyses being termed *data dredging* or *fishing expeditions* [141]. A p-value of 0.05 means there is a 5% likelihood of obtaining such a difference by chance, so that, with 1000 random variables, as many as 50 might be expected to appear significant owing to chance [30]. Adaptations to p-values are often applied when multiple statistical tests are performed on the same data set to reduce the high false positive rate [142, 143]. In [144] the Bonferroni correction is used to reduce the false positive rate of the t-test in an LC-MS studying comparing targeted vs non-targeted approaches in the analysis of kidney transplant patients with good and impaired renal function. However, in parametric tests such as the t-test and ANOVA, assumptions are made about the distribution of the data, and p-values will be meaningless if the assumptions are invalid. Although these tests are quite

robust to non-normally distributed data, they are particularly sensitive to outliers. Considering such reservations, the tests can be used to highlight or corroborate variables that may be of interest. Verwaest et al. [145] use the t-test for feature selection for input to a support vector machine (SVM) learning algorithm and, in the reverse scenario, Nevedomskaya et al. [146] use univariate tests to confirm or reject variables identified in multivariate analyses.

Nonparametric methods remove the need for assumptions on the distribution of the data. The Mann–Whitney U-test, for example, uses ranks rather than the original data and has been applied to NMR data in a targeted analysis to compare the levels of certain metabolites in patients with malabsorption syndrome (MAS) with those of controls [147]. Pears et al. [148] used univariate tests to determine the significance of metabolite changes identified by multivariate analysis. In addition to the t-test, the nonparametric equivalents of the t-test, the Kolmogorov–Smirnov test, and of one-way ANOVA, the Kruskal–Wallis test, were used as well as an F-test to compare the variance of the disease and control groups. Both multivariate and univariate methods were able to classify the ^{1}H NMR spectra in the study.

Correlation may also be used as a univariate method, for example, to reveal metabolites associated with particular patterns over time. Associations between different biochemical variables and ^{1}H NMR spectra intensities were investigated using the Spearman correlation coefficient in a study on diabetic nephropathy [149]. In an evaluation of preprocessing protocols, De Meyer et al. [150] use Pearson correlation to assess how well the variables obtained by different methods relate to metabolite concentrations derived from standard chemical analyses.

2.3.2 Multivariate Approaches

The usefulness of querying individual variables in megavariate data sets has been questioned [30, 151] owing to the high false positive rate associated with multiple tests. Multivariate analysis of variance (MANOVA) extends ANOVA to allow multiple dependent variables, which may or may not be correlated. While ANOVA tests the significance of the difference in means between two or more groups, MANOVA tests the significance of the difference between two or more vectors of means. However, megavariate datasets

result in problems with singularity of the covariance matrix and assumptions that are violated [152]. An alternative generalization, analysis of variance simultaneous component analysis (ASCA) uses an approach similar to PCA to separate the variance originating from different factors and their interactions. The method has been applied to the analysis of metabolomics data, including LC-MS and NMR [153–155]. Multilevel simultaneous component analysis (MSCA) also allows confounded factors to be analysed separately. MSCA has been shown to remove within-subject variation and thus allow between group effects to be differentiated, facilitating biomarker discovery in both LC-MS and ^{1}H NMR [156, 157].

Multivariate methods allow combinations of variables to be considered and may be used for data reduction and visualization or for discrimination and classification. Unsupervised methods can be used in exploratory analyses to identify trends or clustering in the data, but do not attempt to relate observations to a particular class label or response. Supervised methods, on the other hand, aim to associate input features with predetermined categories and can suffer from over-fitting. With many more variables than observations, it is possible to find discriminatory combinations for the available data that would not generalize and it is vital that supervised analyses be validated.

2.3.2.1 Cluster Analysis

Cluster analysis is an exploratory data analysis tool. Although it can be used in classification by associating a class with different clusters, it is primarily an unsupervised method, used to divide data into groups, or clusters, so that the degree of association is strong between members of the same cluster and weak between members of different clusters. Clustering is described in more detail in Chapter 5.

2.3.2.2 Principal Component Analysis

PCA is one of the most widely used multivariate techniques [158]. PCA aims to reduce the dimensionality of the data to a few characteristic dimensions for visualization and further analysis. This smaller set of variables, or components, should retain the important information in the data, effectively summarizing the data. Therefore, the principal components (PCs) are

chosen so that each successive component accounts for maximal variance in the data, not already explained by previous components. This is achieved from the eigenvalue decomposition of the data covariance matrix [159]. The eigenvector corresponding to the largest eigenvalue gives the coefficients in the best 1D approximation to the data, the first PC. The coefficients, known as the *loadings*, show how much each of the original variables contributes to the PC. Better approximations are achieved using more components, with the kth PC determined by the eigenvector corresponding to the kth largest eigenvalue. Data reduction is achieved as most of the variance in the data is explained by the first few PCs. The values of the new variables for each observation are called *scores* and *score plots* for just the first two PCs are often used for visualization.

PCA is an unsupervised method that can be used to identify patterns in the data. If most of the variance in the data is due to interclass differences then clustering according to class should be apparent in the first few PCs. The loadings for the relevant PC can be used to determine the original variables contributing most to any patterns or trends. However, no information about the groups of interest is used to obtain the components and it may be that most of the variance is related to other factors, or simply noise. PCA can also be used to identify potential outliers that may skew the results in further analysis. In fact, the sensitivity of PCA to outliers has led to the development of more robust variants [160]. Just as the median of absolute deviation about the median, or median absolute deviation (MAD), provides a more robust estimator of scale than the standard deviation, the use of a robust covariance matrix allows PCA to be less affected by outliers. A measure of robustness is provided by the 'breakdown point', which gives the maximal number of outliers that can be tolerated as a percentage of the observations in the data set. This percentage decreases as the dimensionality of the data increases, and with high-dimensional data, the algorithms also become intractable. Alternative robust methods are based on projection pursuit (PP), a statistical technique to find the directions onto which the projected data maximize some condition, such as non-normality. In the algorithm proposed by Croux and Ruiz-Gazen [161], these directions are determined by maximizing a robust estimate of the variance. However, this and similar algorithms can be inaccurate with the mega-variate datasets common in chemometrics, and various more stable algorithms offering

improved computational efficiency, such as the GRID algorithm of Croux et al. [162], have been reported since. Recently, Xu et al. [163] have proposed a method termed outlier pursuit that is designed to identify rather than to ignore outliers.

2.3.2.3 Sparse Principal Components Analysis

When PCA reveals patterns in the data associated with the research question, the loadings for the relevant PCs can be used to determine the spectral features responsible. However, the loadings often show that very many features contribute similarly to the variance so that the results are difficult to interpret in terms of metabolic changes. Sparse PCA provides modified components that have few loadings with non-zero values. The SCoTLASS method of Jolliffe and Uddin [164] includes additional constraints on the coefficients when maximizing the variance. In addition to the constraint that the squared coefficients should sum to one, the sum of the absolute values of the coefficients must also be less than a tuning parameter, t. There is a trade-off between the percentage of variance explained and the sparseness of the loadings and the algorithm is computationally expensive, which makes multiple attempts to find the optimal value of t impractical and alternative methods have been proposed. In the same way that variable selection is used in linear regression to produce interpretable models, Zou et al. [165] use a penalized least squares method (LASSO) to impose a constraint on the coefficients to derive components with sparse loadings.

To avoid problems with poor performance due to structural dependencies within the data that are unrelated to the patterns of interest, Allen proposed a generalized principal components analysis (GPCA) [166]. A low-rank matrix approximation is used to obtain a decomposition that directly accounts for structural relationships with penalties used to promote sparsity. Non-negativity constraints have also been introduced to improve interpretability of PCA [167]. In comparison to standard PCA, non-negative principal components analysis (NPCA) was found to be less sensitive to noise and to give better feature extraction when applied to an NMR metabolic data set, with peaks originating from the same compound appearing in the same PC in NPCA, but not standard PCA [168]. Sparseness, known structural dependencies, and non-negativity have been combined to give a sparse form of GPCA termed sparse non-negative generalized principal components ana-

lysis (SGPCA) and its use in metabolomic studies has been demonstrated [169]. The method was found to provide better clustering of samples according to biological relationships and results that were easier to interpret owing to improved feature selection. Although less variance is explained by the components than with either PCA or GPCA, SGPCA was found to provide greater dimensionality reduction when accounting for the number of features selected (i.e. better reduction per feature). However, the sparse calculation does increase the computational power required.

2.3.2.4 Multiway Principal Components Analysis

Multiway techniques have been designed for use with multidimensional data, such as batch processes for which the data can be arranged in a three-way array (batch × variable × time). Multivariate analysis is performed by unfolding the data array in a suitable way and then carrying out traditional analyses such as PCA [170]. Examples of application to three-way arrays of NMR data are given in Pedersen et al. [171].

In a study using ^{1}H NMR to investigate metabolic relationships between tissue types, standard PCA showed that the difference between the tissue types exceeded the differences between individuals and therefore dominated the analysis [172]. However, multiway principal components analysis (MPCA) allowed the variance between the individuals to be assessed and therefore metabolic correlations between different organs inferred.

2.3.2.5 Hierarchical Principal Components Analysis

Multiblock methods allow datasets from different techniques to be analysed simultaneously. In contrast to variable concatenation, in which the data sets are combined to give a single supermatrix, multiblock, or high-level data fusion, involves two levels of multivariate analysis. The first stage is performed on the individual data blocks to provide a scores matrix from each experimental technique and the second is performed on the matrix formed by concatenation of these scores. In one of the first applications of multiblock PCA, Forshed et al. [173] combined NMR data with LC-MS data to give enhanced between group separation. In addition to patterns within individual blocks, multiblock hierarchical methods can reveal common trends across data blocks that could be obscured by other sources of

variation with simple concatenation. By considering melon cultivars as distinct blocks in hierarchical principal components analysis (HPCA) with gas chromatography—electron impact—time of flight mass spectrometry (GC-EI-TOFMS) data, Biais et al. [174] were able to identify common traits in the spatial localization of metabolites across cultivars as well as discriminatory metabolites specific to individual cultivars. An extended HPCA that combined data blocks for each cultivar obtained by GC-MS and by ^{1}H NMR showed correlations in the methods that could be used to aid identification of metabolites. Figure 2.3 shows scores and loadings plots for the HPCA.

2.3.2.6 Supervised Principal Components Analysis

PCA can be combined with other techniques, such as LDA or regression, to provide a supervised method. An alternative way to use PCA for classification and prediction was devised by Wold [175], in which separate PC models are obtained for each class in the training data set and the fit of a new observation is calculated for each class model. The observation is then assigned to the class for which the model gives the best fit. If the fit to every modelled class is poor (according to the residual variance), then an observation need not be assigned. Such an observation may be an outlier or may belong to a class that is not represented in the training data.

2.3.2.7 Discriminant Analysis

The aim of PCA is to find linear transformations that maximize variance, which may not coincide with the best separation of groups. Linear discriminant analysis (LDA) is a supervised technique that uses class information to find linear transformations that maximize group separation. The between-group scatter is maximized, whereas the within-groups scatter is minimized to achieve a reduced dimensional subspace in which the groups are maximally separated [176]. However, when the features are collinear or the number of dimensions exceeds the number of observations, LDA can have problems owing to singularity of the scatter matrices and variable selection may be necessary before LDA [177]. Alvarez et al. use ANOVA to select the LC-MS variables to feed into LDA [178] whilst Imre et al. use forward stepwise variable selection in their selection of a suitable group of N-glycans that can be used to discriminate between cancerous/non-cancerous samples of the

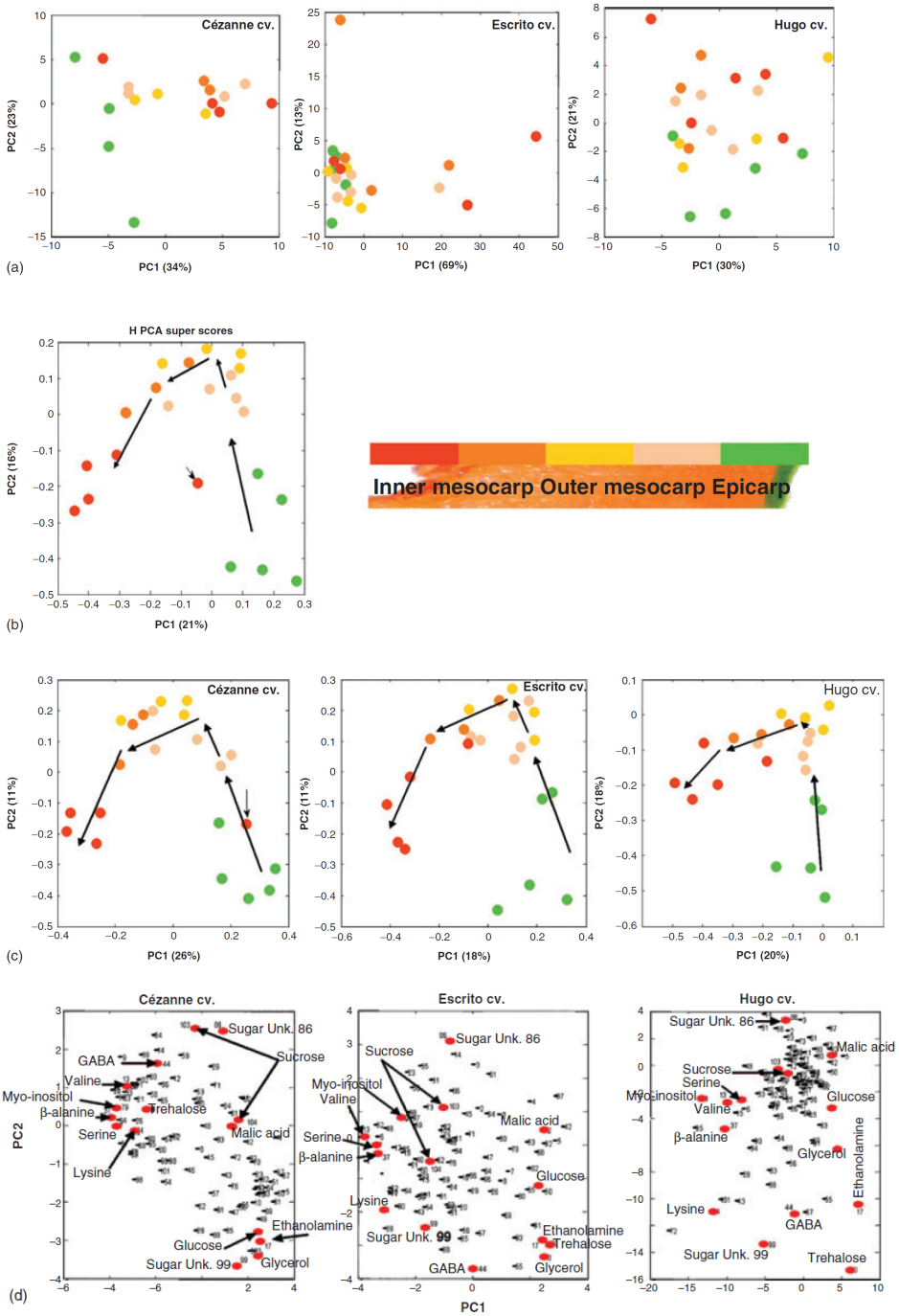


Fig. 2.3: Hierarchical PCA of GC-EI-TOFMS data from different sections of three melon cultivars (Cézanne, Escrito, and Hugo). (a) Standard PCA. (b) Multiblock HPCA super scores plot. (c) HPCA block scores plot. (d) HPCA block loadings plot (the most influential metabolites, selected by N-way ANOVA, are labelled). (Reproduced from Ref. [174]. Copyright 2009, Elsevier.)

LC-MS data. With ^{1}H NMR, Faule et al. [179] selected only the predominant peaks in the spectrum and showed that discriminant analysis applied to these gave good separation between olive oils of different botanical origin. In chemometric studies, discriminant analysis is often applied in combination with other techniques. PCA can be used for data reduction and LDA used with the PC scores as variables to identify class boundaries (hyperplanes). For example, PCA-LDA has been used to identify discriminatory features in ^{1}H NMR spectra from instant coffee samples [13] and sea bream [180]. This method relies on PCA for initial feature selection, but important indicators for a particular class may have low variance and be ignored by PCA [180]. Barker and Rayens [181] show that partial least squares (PLS) is a natural alternative and should provide better data reduction than PCA when discrimination is the goal.

2.3.2.8 Partial Least Squares

The nonlinear iterative partial least squares (NIPALS) algorithm developed by Wold [182] was originally used to calculate PCs without the need for the covariance matrix, but was later adapted as an alternative to principal components regression (PCR) for over-determined regression problems [181]. Partial least squares regression (PLSR) generalizes multiple linear regression without imposing the constraints of other methods, such as canonical correlation analysis (CCA), so that more prediction variables (latent structures) can be extracted. PLS, sometimes referred to as *projection to latent structures*, allows a set of response variables to be modelled by linear combinations of the independent (predictor) variables accounting for maximum variance and is able to handle noisy, correlated, and incomplete data [183]. While PLSR models continuous predictor variables, PLS-DA, models discrete classes and is a well-established and commonly used technique in chemometric analysis.

Linear combinations of the predictor variables provide X-scores with coefficients known as X-weights and multiplying just a few X-scores by the X-loadings provides good approximations to the predictor variables. Similarly, good approximations can be obtained for multivariate response variables, with associated Y-scores, Y-weights, and Y-loadings. The loadings show the relationship between the scores and the original variables and the weights show the exchange of information between the predictor variables

and the response variables or 'scores exchange'. Multiplying the X-scores by the Y-weights gives good predictors of the response variables that can be expressed in terms of a multiple regression model with coefficients that can be used to identify the most discriminatory spectral features. Alternatively, the variable importance in projection (VIP) score for each predictor can be used to provide a summary of its importance to several latent structures [184]. Shao and Li [185] compared PCA-LDA and PLS-DA for application in the quality control of fruit and vegetables. They found that PLS-DA was more useful for discrimination than PCA-LDA and performed well when used with NMR data to predict firmness. PLS-DA can also be used as a measure of separation, for instance [186] use PLS-DA to compare against a novel LC-MS sample preparation technique, using cross validation over 100 iterations to obtain the classification error.

2.3.2.9 Orthogonal Partial Least Squares

Like PCA, the PLS model is obtained by maximizing variance and can be difficult to interpret. A modification of PLS, termed orthogonal partial least squares (OPLS), has been proposed, which separates the variance in the model into two parts [187]. The first is the variation that is common to both the data matrix and the response matrix and is therefore of most interest for classification and prediction. The other part, the so-called structured noise, is the variance specific to the datamatrix and not related to the response matrix. Filtering out the uncorrelated noise leads to a model that is easier to interpret and allows the structure of the noise to be analysed separately, for example, using PCA. In classification, with a response matrix consisting of zeros and ones indicating class membership, OPLS separates the within-groups variance and the between-groups variance. The method can also be applied to time series with time as the response to extract the variance related to time [187]. OPLS has been combined with discriminant analysis (OPLS-DA), for instance to identify metabolites resulting from traditional Chinese preparations to be considered for further research [188].

An improved OPLS algorithm, *O2PLS*, not only includes the orthogonal signal correction (OSC) filter that allows the variance specific to the data matrix to be analysed separately, but also removes structured noise from the response matrix [189]. When a single response vector is involved, the two methods are the same. Variance specific to the response may result, for

example, when pure standards cannot be obtained for various constituents and separate analysis could be important in assessing the quality of the prediction model. The O2PLS model is predictive in both directions.

A novel use of O2PLS, termed *tO2PLS*, proposed by Kirwan et al. [190] operates on the transpose of the data matrix and therefore allows patterns between samples rather than variables to be investigated. The study utilized the bidirectionality of O2PLS to find features in ^{1}H NMR spectra, which could discriminate between black bream exposed to the hormone 17β-estradiol and control fish. The two experimental groups were considered individual data blocks and O2PLS used to separate the spectral features that were common to both and those that were unique to one group. The method was found to perform better than OPLS-DA, which separated groups but failed to indicate specific features. The authors also suggest that the method offers a form of automated preprocessing [190], as the OSC allows the influence of the mean spectra across both datasets to be eliminated, providing an optimized form of centring and scaling.

An evaluation of OPLS models concluded that the method could be used with ^{1}H NMR spectra without reducing the resolution of the data by binning to account for variation in peak positions [104]. Simulations showed that peak shifts only slightly reduced the prediction ability of the model and the authors suggest that the method could be used to model the peak shifts, thereby providing potentially useful information on physiochemical variations in biofluids that are lost when realignment is used. However, it has also been argued that OPLS can never outperform partial least squares (PLS) and any improvement in performance must be due to 'unfair' differences in the comparison [191].

2.3.2.10 Partial least squares modifications

As with PCA, various modifications have been made to the PLS algorithm, including several that introduce sparseness to provide a model that is more easily interpreted. Lê Cao et al. [192] use the NIPALS algorithm with a penalty based on the number of variables, to give a sparse partial least squares (SPLS), which produces latent structures with few nonzero loadings. The method has been used with ^{1}H NMR spectra to detect potential geographical biomarkers for *Salvia miltiorrhiza*, or red sage, used widely in traditional Chinese medicine for the treatment of cardiovascular and cerebrovascular

diseases [193].

The relationship between PLS and LDA was investigated by Barker and Rayens [181], who concluded that LDA would typically outperform PLS whenever it could actually be implemented and noted that a version of PLS-DA based on the between-groups variance would be better suited to discrimination. Sabatier et al. [194] provided such an extension of PLS-DA termed generalized partial least squares-discriminant analysis (GPLS-DA) based on the eigenanalysis of a matrix equivalent to that of LDA. A more general regularized method, regularised PLS (RPLS), includes penalties to encourage sparsity or smoothness, but also adds constraints to account for non-negativity and known data structures (such as ordered chemical shifts in NMR spectroscopy) [195]. The method was demonstrated with simulated and experimental NMR data and shown to increase computational efficiency as well as provide better feature selection and prediction accuracy in comparison to other PLS algorithms.

Interval partial least squares (iPLS) has been suggested as a method to highlight important spectral regions in metabolomic analyses [196]. Spectra are partitioned into equidistant intervals and the root mean squared error of cross-validation (RMSECV) used to identify the regions that are most relevant for prediction of the dependent variables by PLS. Subsequent optimization of the intervals provides a graphical means of variable selection, highlighting important regions of the spectrum.

2.3.2.11 Canonical Correlation Analysis

CCA, also known as canonical variate analysis (CVA), is a statistical method used to show relationships between two data matrices [197]. A linear combination of variables is chosen from each data set such that the correlation between the two data sets is maximized. These linear combinations are the first canonical variates and the coefficients involved are the first canonical weights. Subsequent canonical pairs can be extracted whereby the correlation is maximized subject to being uncorrelated with previous canonical pairs. However, CCA can only be applied to full rank data matrices, whereas rank deficient matrices often occur in chemometrics owing to highly correlated variables.

Yamamoto et al. [198] demonstrate the connection between CCA and PLS and show that a regularized canonical correlation analysis (RCCA)

can be applied in metabolomics. In comparison to PLSR, they found that RCCA required significantly fewer latent variables to obtain a good predictive model. The PLS algorithm has been used within CCA by Nørgaard et al. [199] in a method they call extended canonical variate analysis (ECVA). No dimensionality reduction is required before analysis and the canonical variates can be used in LDA for classification.

The same principle as used in iPLS has been applied to extended canonical variate analysis (ECVA), resulting in iECVA, used with fluorescence spectroscopy for the classification of breast cancer samples [200] and to reveal genotype-specific spectral regions in near infra-red (NIR) data [201]. Both iECVA and iPLS were applied to ^{1}H NMR spectra to investigate the metabolic effects of onion intake, and the same two dietary biomarkers were identified by both methods [202].

2.3.2.12 Kernel Methods

A kernel function can be used to transform linear algorithms, such as PLS and CCA, into nonlinear algorithms by replacing the dot product between two vectors with a kernel function. This allows observations to be mapped into a higher dimensional space in which they may be more easily separated, without explicitly computing the mapping [203]. The use of kernel functions can dramatically increase the computational efficiency of algorithms, as shown by Yamamoto et al. [198] using a linear kernel used with CCA. A kernel PLS (KPLS) algorithm was developed for data sets with many observations and shown to be significantly faster than the classical PLS algorithm [204]. A similar algorithm for dealing with the opposite case, i.e. data sets with fewer observations than variables followed [205]. A variant of OPLS – kernel orthogonal PLS (KOPLS), which uses a Gaussian kernel function, has been shown to improve predictive performance, in both classification and regression examples, when used to model nonlinear relationships between descriptor and response variables. The performance of multiblock PLS-DA has been compared to Consensus OPLS-DA, a variety of KOPLS-DA, in the analysis of LC-MS spectra [206]. The predictive performance of the two multiblock models was noted to be similar although the results of the kernel method were found to be more interpretable. The KOPLS algorithm has been used with ^{1}H NMR spectra to compare the metabolic signatures of patients with overt and potential Celiac disease with controls

[207]. The KOPLS provides a nonlinear data reduction step and classification is conducted on the resulting scores using a SVM algorithm. The SVM (support vector machine) algorithm is a supervised model that seeks the hyperplane which optimally separates the target groups [208]. Primarily used for classification, in addition to linear separation, SVMs allow data to be transformed into higher-dimensional space using a kernel-function. The *support vectors* can be seen as the subset points representing data most difficult to classify. Optimization of the kernel function parameter is nontrivial, and an automated procedure using simulated annealing has been incorporated in the KOPLS algorithm [209]. The algorithm was tested with three different NMR data sets and compared to a linear OPLS model [210]. The prediction results obtained using the KOPLS algorithm were as good as, if not better than those obtained with the linear algorithm, but the authors acknowledge that OPLS may perform better with problems that are truly linear if insufficient observations are available for training.

2.3.2.13 Evolutionary Computing

Evolutionary learning algorithms are techniques based on the process of natural selection and are popular for solving optimization problems [211]. An initial population of random solutions is created as the first generation and new solutions are produced from these using operations that mimic reproduction and mutation. Solutions are evaluated using a fitness function and the fittest survive to the next generation so that optimal solutions evolve over a number of generations. In a genetic algorithm (GA), each possible solution is encoded in a string (chromosome), often a binary string consisting only of 0s and 1s, but other representations are possible. Forshed et al. [99] use a GA to optimize the alignment of peaks in NMR spectra, whilst Yeo et al. use a genetic algorithm for the calibration of LC-MS analysis parameters in order to reduce the need for expert opinion, using the algorithm to select parameters for peak resolution and k-means clustering, amongst others [212]. The large search space however is noted as a problem, and seeding the process with a set of reasonable starting parameters is suggested. GAs can be applied to classification problems, using the number of correct classifications achieved on the training set as a fitness function. Johnson et al. [213] were able to identify the regions in FT-IR spectra that allowed discrimination between control and salt-treated tomato varieties. Genetic programming is

also combined with other techniques, for instance it has been used for both feature selection and feature construction of LC-MS data in order to reduce data dimensionality and improve the performance of k nearest neighbours (kNN) and naive Bayes (NB) classifiers [214].

In genetic programming (GP), the candidate solutions are composite functions usually represented by tree structures [215]. Although the method has been used successfully with ^{1}H NMR spectra from brain biopsy extracts to classify tumours [216], the size of the search space can result in very long training times, and a two-stage GP has been proposed to overcome this problem [217]. The first stage serves as a feature selection method to provide a reduced search space for the second stage. Only variables selected a number of times in trees giving at least 90% correct classification are used in the second stage. This not only reduces the search space for the problem, leading to faster convergence to an optimal classification rate, but also reduces the risk of overfitting. The genetic algorithm (GA) revealed marker resonances in ^{1}H NMR spectra that were able to distinguish between genetically modified barley and null-segregant lines. Although it has been suggested that GP techniques are likely to be problem specific and will differ significantly in performance with different data sets [218], the two-stage GA has also been successfully applied in food authenticity studies [219, 220]. As evolutionary computing methods identify spectral features directly and do not involve a transformation of the variables, as is the case with many multivariate methods, the results are more easily interpreted.

2.3.2.14 Artificial Neural Networks

As evolutionary algorithms mimic the process of natural selection, artificial neural networks (ANNs) are inspired by the processes in the brain. They consist of a graph of interconnected processing units that act as virtual neurons. The output from each neuron is typically a function of the weighted sum of its inputs, where the weights are learnt from the data through either supervised or unsupervised learning techniques (or some combination of the two). There are a number of ANN algorithms governing how the data is processed and, in chemometric applications, the model is often chosen based on experience of what processes work [221]. In early applications to NMR data, neural networks were trained to recognize the chemical shift patterns in ^{1}H NMR spectra of sugar alditols [222] and ^{13}C NMR spectra of acyclic alkanes

[223] and to locate cross-peaks in a 2D ^{1}H NMR correlation spectroscopy (COSY) spectrum [224]. More recently, ANNs have been applied to NMR spectra to address a range of problems including classification of chemical reaction types [225] and prediction of antioxidant activity in plant varieties [226]. Studies have also used ANNs to predict retention times in LC/LC-MS analysis. Radial basis functions, probabilistic neural networks, generalised regression networks and multilayer perceptrons have been used to predict the retention times (LC) of compounds based on their SMILES molecular descriptors [227, 228]. Like GAs, neural networks have been combined with other analysis methods. For example, Rezzi et al. [177, 229] found that sample classification of fish and olive oils using PCA and a probabilistic neural network gave better results and required fewer components than PCA-LDA.

Self-organizing maps (SOMs) are an unsupervised form of ANN in which the network attempts to approximate the distribution of the data. The data are projected nonlinearly from the original variables onto a 2D map of the network nodes. The data that are close in the original data space are also close in the 2D reduced space. This provides a convenient method for visualizing the data allowing any natural similarities and groupings to be recognized. Kaartinen et al. [230] showed that clinically relevant classification of plasma lipoprotein lipids could be achieved using SOM analysis of ^{1}H NMR spectra. However, this also highlighted some problems with the use of SOMs, in that they do not perform well with small numbers of samples, particularly when the data is not completely accurate. These problems may be alleviated somewhat by reducing the number of variables by only using the relevant part of the spectra.The sensitivity of SOMs has however led to them being suggested as a means of outlier detection [231]. SOMs will group observations based on the similarities of their spectra, but will also highlight how well attached observations are to the group, with those falling outside suggesting potential outliers.

2.3.2.15 Correlation-Based Techniques

The correlation between spectral features can be used to highlight connectivities, such as links between atoms sharing spin systems in the same molecule as would be seen in two-dimensional NMR spectroscopy (TOCSY) spectra, providing the motivation for statistical total correlation spectroscopy

(STOCSY) [232]. Correlations are plotted as 2D contour maps, similar to 2D COSY plots and can also show biological covariance (e.g. molecules sharing the same pathway) as well as negative correlations. Cloarec et al. [232] applied STOCSY to ^{1}H NMR spectra from different mouse strains to study insulin resistance. They used OPLS to highlight 'driver' peaks and demonstrated the effectiveness of STOCSY for the identification of the relevant molecules. STOCSY has been shown to provide both intramolecular and intermolecular correlations, providing the rationale behind cluster analysis statistical spectroscopy (cluster analysis statistical spectroscopy (CLASSY)), an unsupervised approach that uses correlation clustering to deconvolute spectra from complex mixtures according to fold-change [233]. Local clustering is used to identify peaks from the same molecule, and global clustering is used to reveal metabolic networks. The correlation heat maps generated can be used to aid biological interpretation of NMR data sets. The method was illustrated with ^{1}H NMR spectra from rat urine to model the development of toxin-induced prancreatitis, which was shown to cause coordinated changes in compounds with similar pathway connections. Figure 2.4 illustrates the information flow in CLASSY analysis.

2.3.2.16 Validation

Unsupervised methods, such as PCA and cluster analysis, allow multivariate data to be visualized and can be used in exploratory analyses without any assumptions as to what patterns may emerge. For prediction or classification, there are many powerful supervised methods that fit a model to the data. However, it is often possible to obtain a model that provides a good fit to the data, but is in fact over-fitted. When the number of variables, d, exceeds the number of observations, n, the risk of overfitting is high and some form of validation is extremely important. The rule of thumb given by Defernez and Kemsley is that,when $d > (n - g)/3$ (where g is the number of groups), over-fitting may well occur [234]. To assess a model's predictive power, it is necessary to apply the model to data that were not used to obtain the model. The holdout method involves keeping back a proportion of the observations for use as an independent test data set and fitting the model to the remaining training data set. Shao and Li [185] use holdout cross-validation in the analysis of NMR data on sweetcorn kernel heat treatment. They highlight the dangers of over-fitting when cross validation

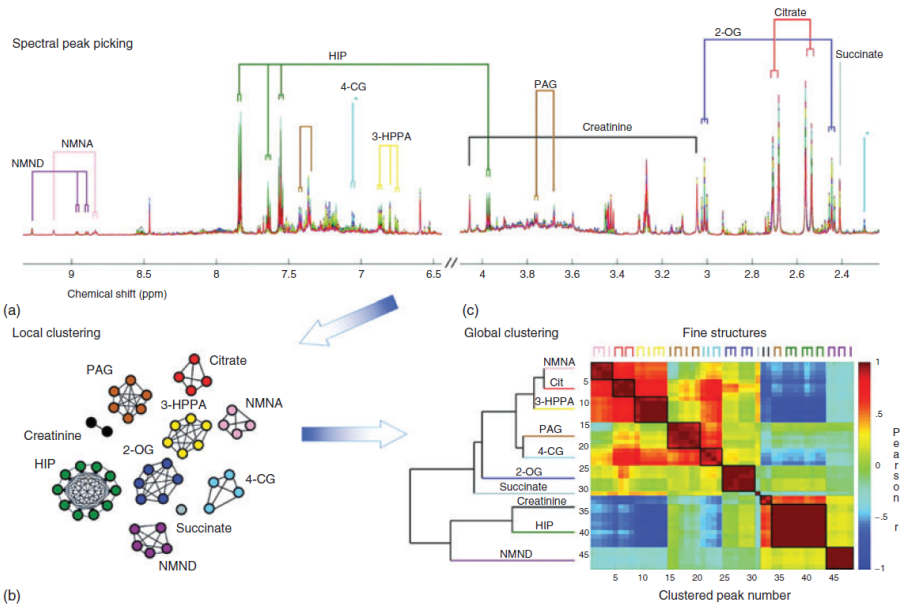


Fig. 2.4: Information flow of CLASSY NMR analysis. (a) shows spectral assignments for 10 metabolites present in control rat urine. (b) shows the local correlation clusters corresponding to each molecular structure. These local clusters are related to each other by global hierarchical clustering (c). Local clusters are differentiated from global clusters by enclosure within diagonal blocks, resulting in a correlation block matrix. Local clusters are assigned to a chemical structure by relating the chemical shifts and fine structure of each cluster to the compound's NMR spectrum. (Reproduced from Ref. [233]. Copyright 2010, American Chemical Society.)

(CV) is not used, as a 100% success rate on training data becomes 85% on the test data. While the use of an independent test data set is desirable, this needs to be large enough to give an accurate estimate of the prediction error. However, the more data used to obtain the model are, the better that model is likely to be, and therefore the holdout method requires sufficient observations to be available. This is often not the case in chemometric studies and various resampling methods are used, at the expense of greater computational cost. In k-fold cross-validation, the data set is split into k equal-sized sections. One of these sections is used as a test set and the other $(k-1)$ sections are used together as a training set. This process is repeated k times, each time leaving out a different section of the data and the average error rate taken as an estimate of the model's predictive ability. Smaller test sets mean more training data are available for training, providing more accurate models, and the extreme case is when $k = n$, known as leave-one-out (LOO) *cross-validation*. Although computationally expensive, LOO has been shown to give a good, if slightly conservative, error estimate, consistent with statistical theory [235].

Internal cross-validation is used during training, for example, to determine the number of latent structures to be used, and can help to distinguish the structure in the data from the noise in order to prevent overfitting. External validation allows a model to be tested on new data and is considered mandatory to assess the predictive ability of a model [236]. Recommendations from the standard metabolic reporting structures (SMRS) working group strongly emphasize the necessity of proper validation for supervised classification, described as an 'inherently biased technique'. Rules of thumb are given for the form of validation to choose when insufficient observations are available for an independent test set: if $n < 10$, LOO is appropriate; for $n \approx 30$, k-fold cross-validation is appropriate and less computationally demanding than LOO.

Hawkins points out that cross-validation involves the comparison of models; a complex model over-fits the data if a simpler model fits equally well [235]. Adding irrelevant variables can make the prediction ability of the model worse and can lead to wrong interpretation of the results. It could be that completely different models provide the same level of discrimination, but it is possible to check stability by comparison of the models obtained for the k training sets during cross-validation [237]/ Tripathi et al. [238] use

the term cross-validation to indicate that results of two or more analytic techniques have been compared, in their case MS and NMR.

2.3.2.17 Future Perspectives

There are many different techniques available for chemometric analysis, and new methods as well as modifications of existing algorithms, more suited to the analysis of mega-variate data sets, are still being developed. The lack of a common benchmark means that new methods are usually only compared with traditional techniques or earlier versions of the same algorithm [239]. Fonville et al. [240] have suggested a nutrigenomic dataset, with two genomic strains and two diet groups, as a benchmark for latent variable methods. In the data, originally presented in Dumas et al. [241], the two diet groups have very different variance structures that dominate PCA and overwhelm genetic differences. The ability to discriminate between genetic strains could provide a criterion for assessment. However, the definitive evaluation of any technique must be its efficacy in providing novel biological insights. Our existing knowledge of biological systems is likely to be biased toward easy-to-identify compounds and networks so that the ability to reproduce known results may not be the best indicator of performance [242]. While it is unlikely that there will ever be a one-size-fits-all solution, a set of standard procedures may help to identify which techniques are favourable in which circumstances. Greater accessibility naturally leads to more evaluation of methods under different conditions and, while there is no common repository for techniques, an increasing number of research groups are now presenting R and Matlab packages.

Multiblock methods, such as hierarchical PCA and PLS models, were designed to improve interpretation when variables could be separated into meaningful blocks [243], but can be used to combine multiple experiments. In addition to methods that use the same multivariate method for analysis of the individual blocks and for the higher level analysis of the scores, for example, PCA-PCA, mixed-mode hierarchical methods, such as PCA-PLS, can be employed. In PCA-PLS, the individual blocks undergo unsupervised analysis by PCA and the scores for a varying number of components can then be analysed by PLS. Various combinations of multivariate methods are possible for such data fusion and the blocks co-analysed may be different NMR experiments or from different technologies. Forshed et al.

[173] evaluated various methods for the integration of data obtained by ^{1}H NMR and LC-MS. The relationship between metabolite concentrations and protein abundances was exploited in a two-stage approach using O-PLSR [244]. Multiple correlations between metabolites and proteins were found and related to the disease profile in a mouse model of prostate cancer. The statistical approach used in methods such as STOCSY has also been extended to combine datasets from different experimental techniques. Statistical heterospectroscopy (SHY) has been used to combine NMR and UPLC-MS data sets and reveal toxin-disrupted metabolic pathways [234].

3. DATASETS

3.1 Medicago dataset: An LC-MS investigation into drought and disease resistance in legumes

3.1.1 Introduction

Legumes are an important agricultural crop, which encompass a wide variety of vegetables including peas, beans and lentils. They act as a good source of fibre and protein and can replace imported soya as a feedstock for animals thereby reducing transportation, its associated costs and environmental impact. In addition, their nitrogen fixing ability reduces the amount of artificial fertiliser required and actively puts nitrogen back into the soil for use by future crops [245].

Legumes are particularly sensitive to the effects of drought, which impacts both the overall yield as well as the nitrogen fixing capability. This is of particular significance in arid regions with little overall rainfall, as well as regions where rainfall is inconsistent if plentiful. Climate change leading to higher temperatures and drier conditions are likely to exacerbate this issue further in the future [246]. Yield can also be affected by the presence of the pathogen *Fusarium oxysporum*, which causes wilt in many crop species worldwide, including legumes, the effects of which become even more severe in the combined presence of other stresses [247]. Once established the Fusarium pathogen is difficult to remove and, with most farmers unable to support the cost of soil sterilisation, the only effective measure is crop rotation, which puts further constraints on the amount of crop which can be viably harvested [248, 249]. However, there is a wide range of tolerance in existing crop varieties and it is hoped that by understanding the genes and processes responsible for traits such as disease and drought resistance it will be possible to select and breed for desirable attributes in future varieties [250].

3.1.1.1 Experimental procedure

Medicago truncatula, a model legume, was subjected to individual biotic and abiotic stresses, and a combination thereof. A total of 150 plants were grown comprising four experimental groups as follows:

$\mathcal{C}$ – Control group

$\mathcal{D}$ – Abiotic stress group – subject to drought

$\mathcal{F}$ – Biotic stress group – infected with the pathogen *Fusarium oxysporum*

$\mathcal{B}$ – Dual stress group – subject to both drought and infection with *Fusarium*

Medicago truncatula (Jemalong A17 genotype) seeds were planted in 350 ml pots containing a 3:1 mixture of perlite to sand by volume. Plants were grown in a greenhouse at a temperature of 28 °C and humidity was maintained using a fog system. *Fusarium* inoculation was carried out by watering the plants with 50 ml of Fusarium inoculate. Drought plants were subject to a 40% drought stress by weight of water, a proportion determined to be effective from a previous pilot study. Three plants (biological replicates) were harvested from each experimental group at daily intervals for 12 days. For the C and F groups 78 plants were harvested from days 1 to 12, whilst for $\mathcal{D}$ and $\mathcal{B}$ harvesting commenced one day later, from days 2 to 12 (72 plants), to allow uniform drying of the growth medium. Each plant was removed carefully from its substrate/gauze to minimise damage. The plant was shaken and gently washed to remove any bound substrate. The plants were carefully dried before leaves were cut directly into beakers of liquid nitrogen. Only healthy mature leaves were cut whilst dead or very young leaves were discarded. After freezing, the material was recovered from the nitrogen and stored in aluminium foil before freeze-drying for approximately 48 hours. Lyophilised samples were then stored and transported for metabolomic analysis at room temperature.

Prior to analysis each dried sample was initially ground carefully into a fine powder using a pestle and mortar to preserve as much material as possible. 5 mg $\pm$ 1 mg of ground sample was accurately weighed into a labelled 2 ml Eppendorf tube. To 5 mg of sample, 1 ml of extraction solvent (1:1 (v/v) methanol:water) was added. Metabolites were extracted into the solvent by shaking for 30 minutes. The solid material was then removed

by centrifugation at 14,000 rpm for 10 minutes and the supernatant liquid split into two 400 µl aliquots, of which one was used for LC-HRMS (Liquid chromatography-high resolution mass spectrometry) analysis. The supernatant to be analysed by LC-HRMS was diluted 4 fold using methanol : water 1:1 In addition to the samples, an in-house reference was extracted daily as a quality control measure. As the amount of material available from experimental samples was very low the material for the QC samples was sourced from a homogenised mixture of control samples collected from a previous experiment following a similar design. This allowed the metabolites likely to be present in the experimental samples to be included in the QC samples without requiring the use of the limited experimental material in order to create the QCs.

3.1.1.2 LC-HRMS parameters

One hundred and forty nine leaf samples were ultimately analysed – the number being slightly lower than anticipated (150) due to plant death. Extractions were subject to both positive (+) and negative (–) mode LC-MS, giving two datasets ($\mathcal{L}$+, $\mathcal{L}$–). LC-MS analysis was conducted in 7 batches to which the samples were assigned randomly to ensure that no particular batch was dominated by any particular experimental group or age-range.

The chromatography column used was an ACE 3Q 150 × 3 mm, 3 µm (Advanced Chromatography Technologies, Aberdeen, UK.). Mobile phases were 0.1% formic acid in water mobile phase A (MPA) and 0.1% formic acid in acetonitrile mobile phase B (MPB). The gradient elution applied was 100% MPA for 5 minutes before increasing to 100% MPB over 15 minutes. This was held for 10 minutes before reverting back to 100% MPA and held for 2 minutes. Injection volume was 10 µl using a full loop injection, flow rate was 0.4 ml/min and column temperature was 25 °C.

The MS used was a Thermo Exactive (Thermo Fisher Scientific, MA, USA.) set at 50,000 resolution full width at half maximum (FWHM) (at 200 m/z) with an acquisition speed of 2 Hz. The column was conditioned before sample analysis using 15 QC injections and then QCs were inserted between every 6 experimental samples.

Leaf (L)
184 observations (149 exp. & 35 QC)
1239 $\mathcal{L}$– peaks 1681 $\mathcal{L}$+ peaks

Tab. 3.1: A summary of the number of observations and peaks for the *Medicago* dataset

3.1.1.3 Data preprocessing

The raw LC-MS data were pre-processed using Progenesis QI [127]. The software aligned all MS spectra in the retention-time domain before applying deconvolution and peak picking algorithms, providing a matrix of potential metabolites against observations. The potential metabolites were initially annotated by accurate mass m/z (between 80 and 1000) and retention time (between 1 and 30 minutes) of their corresponding peak. In reality some of these peaks may be due to erroneous peak detection or several peaks may represent the same compound. Table 3.1 shows the number of observations with the number of metabolites recorded for each dataset.

3.1.2 Initial analysis and discussion

3.1.2.1 Batch correction

PCA was carried out on each of the datasets ($\mathcal{L}$–, $\mathcal{L}$+). It is apparent from the visualisations of the first two PCs that the variance between the experimental observations is dominated by differences between the LC-MS batches. The PCA scores plot for the first two PCs for $\mathcal{L}$– is shown as an example in Figure 3.1.

To include the effects of low-intensity metabolites in the analysis the data were autoscaled. The differences between batches in the PCA plots become even more prominent post-scaling (not shown). Whilst some experimental differences – such as separation of the QC samples can be seen in the plots the batch differences largely obscure the experiential differences. In order to reduce the effects of the batch differences a batch correction method, as outlined in [251] was performed on each of the datasets in turn. A visualisation of this correction on a single variable (LN294) is shown in Figure 3.2. It is important to note that this correction must be applied *pre-scaling*. Since the correction involves dividing by the median line, dividing

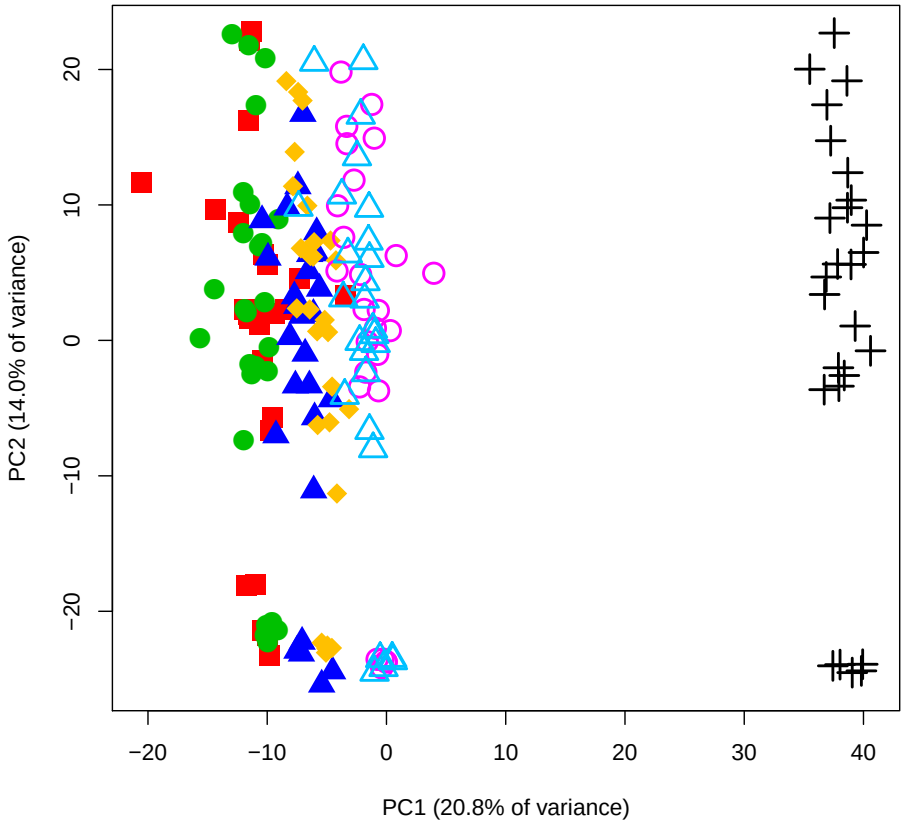


Fig. 3.1: PCA scores plot for the $\mathcal{L}-$ dataset. The data has not been scaled. Icons and colours correspond to LC-MS batch: 1. ■, 2. ●, 3. ▲, 4. ♦, 5. ○, 6. △, 7. +. Peak variance of the data is dominated by batch variance, highlighted by the clustering of peaks by batch across the first principal component.

by values close to zero, such as those typically present *post-scaling*, results in extremely large values in the result, leading to large differences between the within-batch variances, if not the averages.

For the $\mathcal{L}$– dataset this correction was visually successful, with batch differences no longer manifest in the both individual variable and PCA plots and the differences between experimental groups becoming more apparent. The PCA scores plot of the scaled data, post batch correction, for the example dataset ($\mathcal{L}$–) is shown in Figure 3.3.

However, this method was not able to correct for the batch differences in the $\mathcal{L}$+ dataset as shown in Figure 3.4. Several of the batches are "split" along the first principal component (PC1), with part of the batch having low scores for PC1 and the remainder having higher scores. One of the implications of this is that the assumptions of standard statistical tests, such as t-tests or ANOVA may be invalid, since they cannot be applied to a binomial distribution. These problems with the QC correction form the motivation for a the "background correction" batch correction method, which is covered in more detail in Chapter 4.

Post batch correction (QC for the $\mathcal{L}$– dataset and BG for the $\mathcal{L}$+ dataset) the variation due to experimental groups is more apparent. This is shown in Figure 3.5.

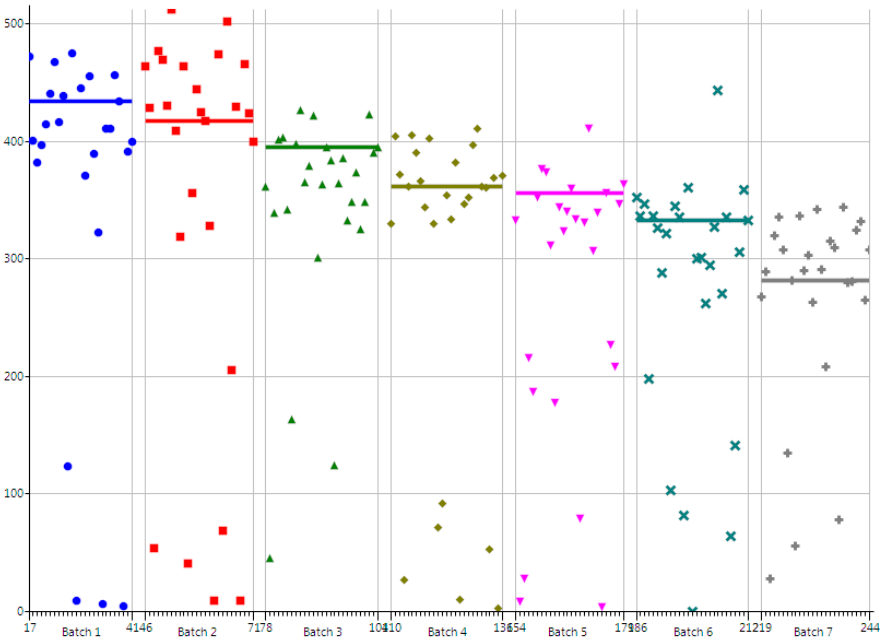
3.1.2.2 Combined datasets

Since the two ion modes share the same set of observations it was possible to combine the $\mathcal{L}$– with the $\mathcal{L}$+ dataset. This produced a larger dataset (leaf-combined ($\mathcal{L}$+/–)) sharing the same set of observations.

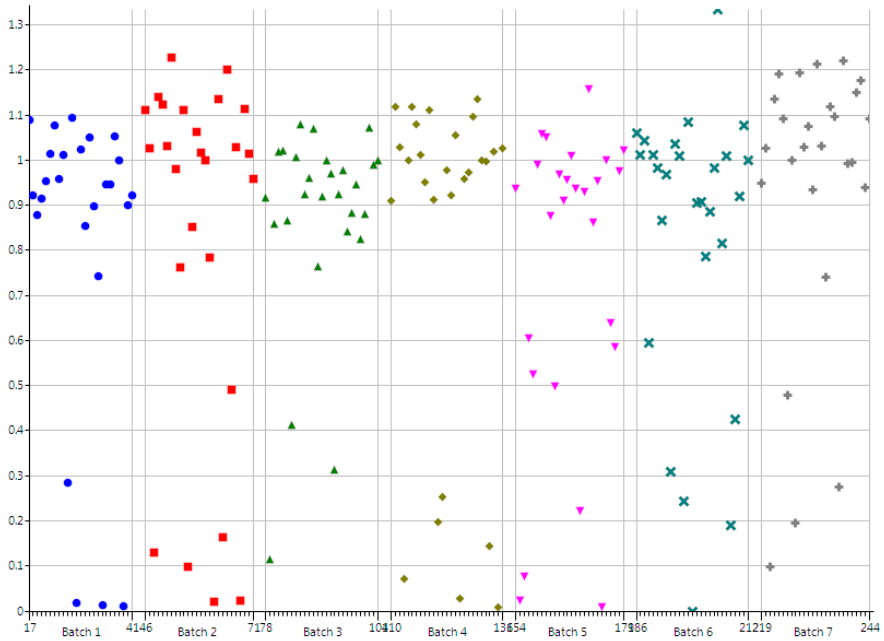
3.1.2.3 Basic analysis

Identifying peaks showing a response to the experimental conditions may allow the identification of pathways displaying stress resistance. In order to identify overtly responsive peaks, t-tests were conducted for each peaks, contrasting the values for each of the experimental groups in turn, against the control group. Several issues are apparent from the results:

- The temporal aspect is ignored. It would be expected that the first few data-points to be similar in all groups, however, the peak showing the



(a) Before



(b) After

Fig. 3.2: Visualisation of a single variable ($\mathcal{L}$– #294) before and after QC correction. The Y axes show the peak intensity and the X axes show the acquisition order. Samples are shown using different symbols and colours across the X axes to denote the different batches (1-7). The correction shows a clear effect, reducing the differences in intensity between samples taken from different batches.

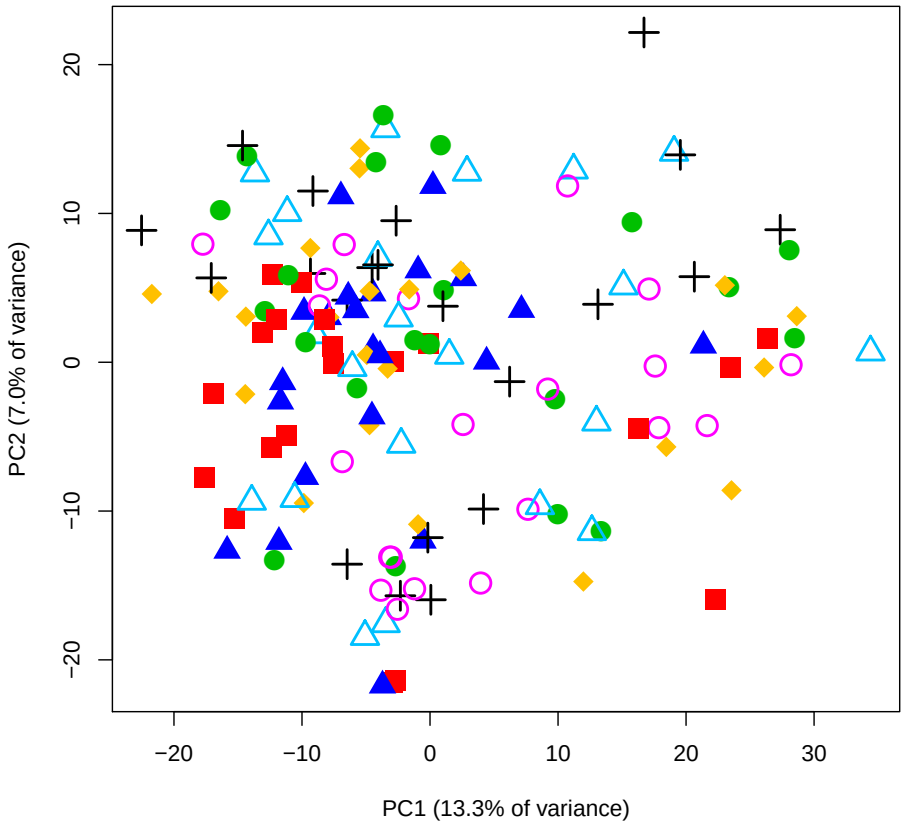


Fig. 3.3: PCA scores plot for the $\mathcal{L}$– dataset, post-batch-correction and autoscaling. Icons and colours correspond to LC-MS batch: 1. ■, 2. ●, 3. ▲, 4. ♦, 5. ○, 6. △, 7. +. Post-correction, the differences due to LC-MS batch are no longer apparent.

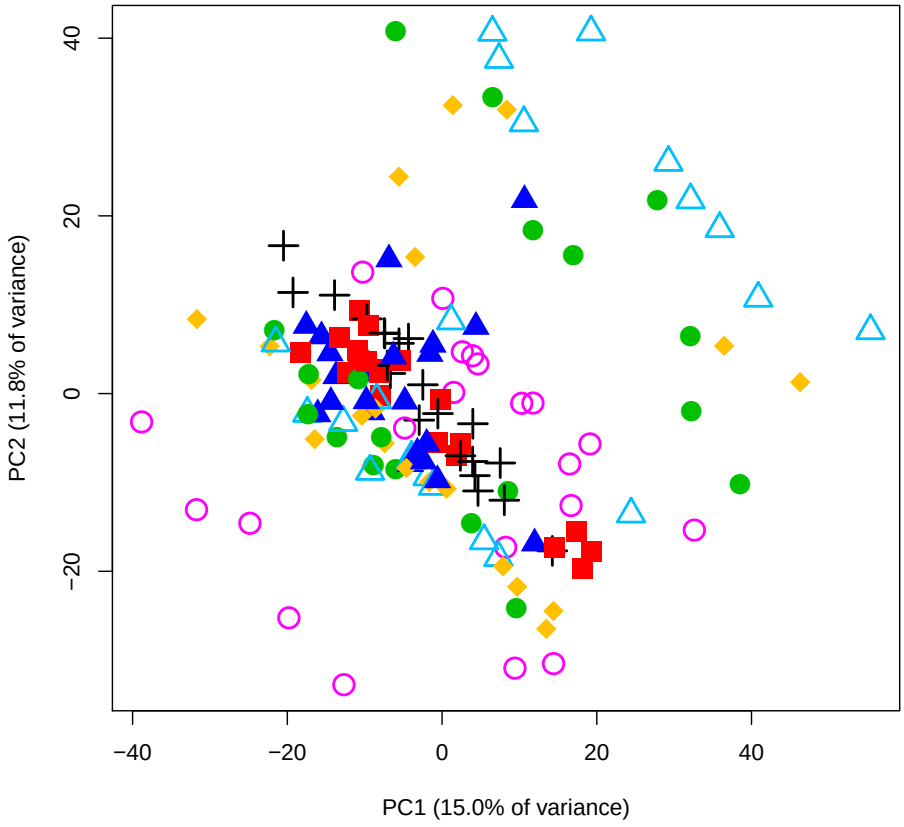


Fig. 3.4: PCA scores plot for the $\mathcal{L}+$ dataset, post-batch-correction and autoscaling. Icons and colours correspond to LC-MS batch: 1. ■, 2. ●, 3. ▲, 4. ♦, 5. ○, 6. △, 7. +. Post-correction using the "mean of the QC" method, the differences due to LC-MS batch are still apparent across PC1 and PC2. Furthermore, several batches show a "splitting" of the samples into two distinct groups across PC1 and PC2.

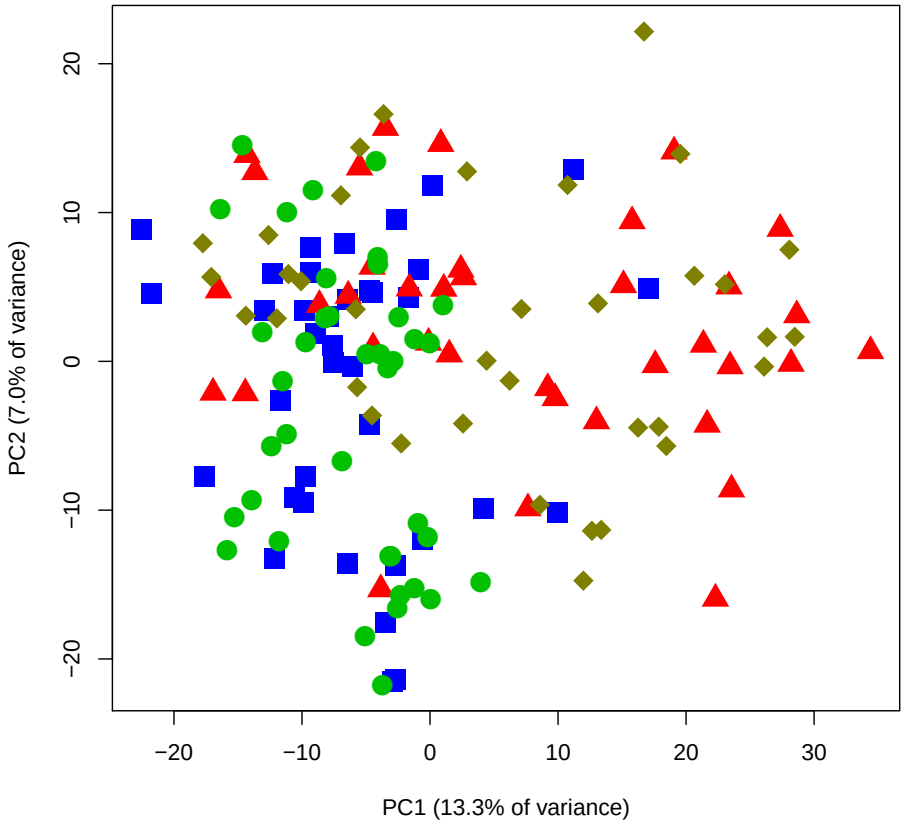


Fig. 3.5: PCA scores for the $\mathcal{L}$– dataset, post batch correction and autoscaling. Icons and colours denote experimental group. ■: Control group, ▲: Drought group, ●: Fusarium group, ♦: Dual-stress group. This figure is the same as Figure 3.3, showing that whilst differences due to batch are no longer apparent, the differences due to experimental group are now manifest.

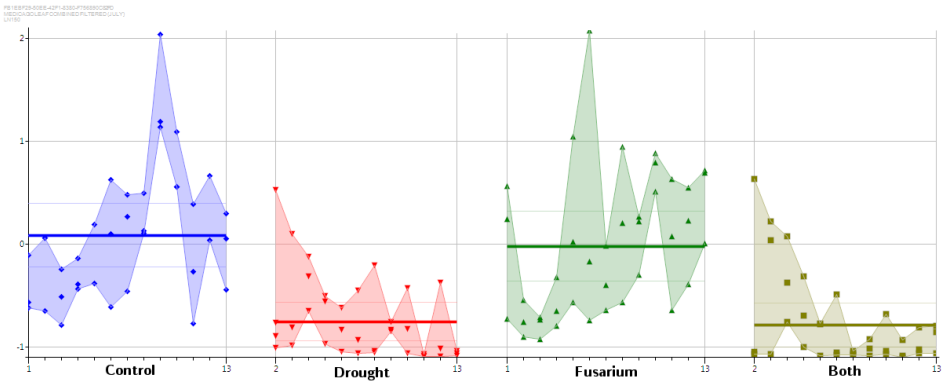


Fig. 3.6: Plot showing the intensities for peak, LN150, which has the lowest t-test p-value for the $\mathcal{C}$ against $\mathcal{D}$ comparison in the $\mathcal{L}+/-$ dataset. Time is displayed along the x axis, for each of the experimental groups in turn. The y axis denotes the ion intensity of for each sample. Whilst the low p-value indicates differences between the samples from the $\mathcal{C}$ and $\mathcal{D}$ groups, samples from the two groups do not noticeably deviate as time (X-axis) increases.

strongest $\mathcal{C}$-$\mathcal{D}$ separation (LN150, Figure 3.6) shows no strong trend in either group, despite having more separation in the first few points.

- There is no fixed set of profiles that are obtained (i.e. the strongest results do not, for example, all show increasing trends over time), making interpretation of the results difficult.

- As a method using the mean and variance, the t-test is particularly sensitive to outliers. This is of particularly significance in our dataset which contains a noticeable degree of noise. This is exemplified in the compound showing the strongest $\mathcal{C}$-$\mathcal{F}$ compound, which possesses a high t score primarily due to the presence of an outlier (not shown).

- In our data, strong correlation is observed between the $\mathcal{C}$ and $\mathcal{F}$ groups, as well as the $\mathcal{B}$ and $\mathcal{D}$ groups. In the case of Drought-Fusarium infection, a high score against control may be due more to drought than combined stress.

Correlation and regression methods are able to account for the temporal nature of the data. The Pearson correlation coefficient (PCC) measures the correlation between two variables and is thus able to identify metabolites

associated with particular patterns over time. Here the experimental observations are compared against "template patterns". These templates are constructed to identify variables showing correlation with time only under the specified experimental conditions.

PCC measures correlation between two vectors (X and Y). For n observations, the first vector ($X = \{X_1, X_2, ...X_n\}$) was set to the intensities of the experimental observations, hence:

$$X_i = I_i \tag{3.1}$$

Where I_i is the intensity of the ith observation. Each value (Y_i) of the second vector was set to the day of extraction (1–12) for observations in the set of experimental conditions of interest (Q), and to 0 for all other observations from the other experimental conditions:

$$Y_i = \begin{cases} T_i & \text{if } G_i \in Q \\ 0 & \text{if } G_i \notin Q \end{cases} \tag{3.2}$$

Where T_i is the age of the ith observation in days, and G_i is its experimental group.

Figure 3.7 demonstrates a template visually, showing the plot of a response variable where $Q = \{\mathcal{D}, \mathcal{B}\}$. Four different values were tested for Q: $\{\mathcal{D}\}$, $\{\mathcal{D}, \mathcal{B}\}$, $\{\mathcal{F}\}$ and $\{\mathcal{F}, \mathcal{B}\}$.

Both Pearson Correlation and PLSR regression were used as a means of identifying variables of interest. In the case of PLSR, VIP scores were used to quantify the role of each peak in the regression. Since VIP scores are calculated across a number of components this was optimised to maximise the predictive ability of the regression whilst avoiding overfitting. This optimisation was performed using LOO cross-validation in order to minimise the root mean squared error of prediction (RMSEP). The plot of RMSEP against number of components (not shown) suggests the ideal number of components is 4, after which overfitting is seen and the predictive accuracy of PLSR on the validation set falls.

Results were initially confounded by significant noise in the data, which can be seen in the examination of individual time-series. Noise can be reduced in the dataset through the use of sample replicates, either biological (i.e. different plants) or technical (i.e. multiple analysis of the same samples).

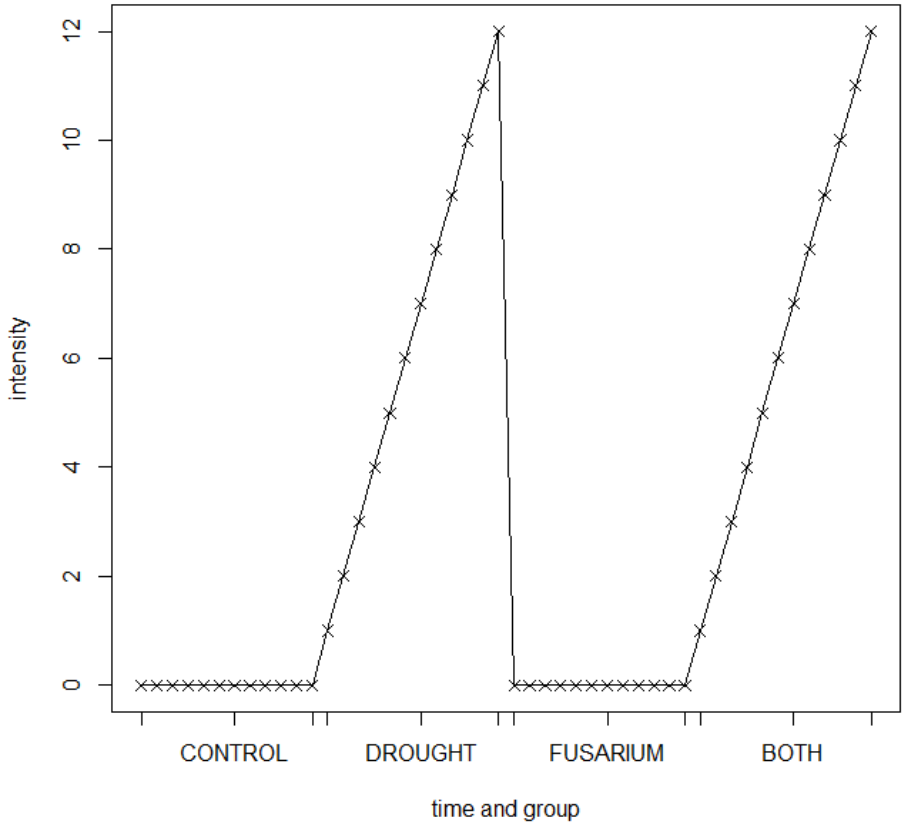


Fig. 3.7: Plot showing a template profile to identify trends with linear correlations with time in the $\mathcal{D}$ and $\mathcal{B}$ groups. The X-axis is denotes time, for each experimental group in turn, whilst the Y-axis denotes an arbitrary intensity value. The $\mathcal{C}$ and $\mathcal{F}$ values are fixed at 0, whilst the $\mathcal{D}$ and $\mathcal{B}$ values mimic the time-value (X-axis) for their group.

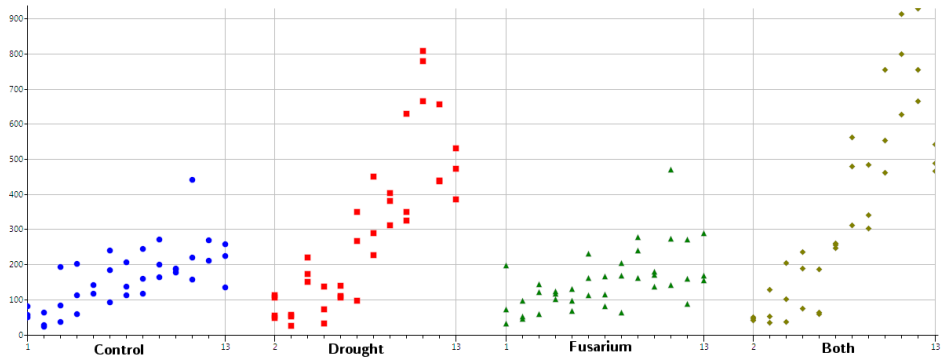


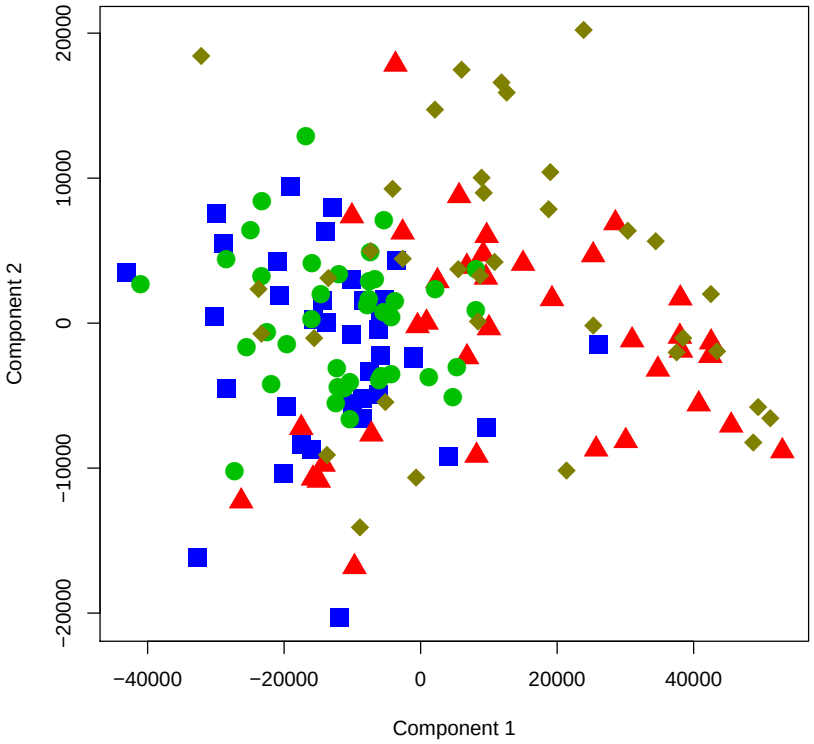
Fig. 3.8: Plot of the intensities for peak LN244, the variable with the highest correlation with the $\mathcal{D}/\mathcal{B}$ template. The X-axis denotes time, for each experimental group in turn, whilst the Y axis denotes intensity (not unscaled). Clear trends over time can be seen for the $\mathcal{D}$ and $\mathcal{B}$ groups, with less noticable trends for the $\mathcal{C}$ and $\mathcal{F}$ groups. Contrast this with the template itself, shown in Figure 3.7.

.

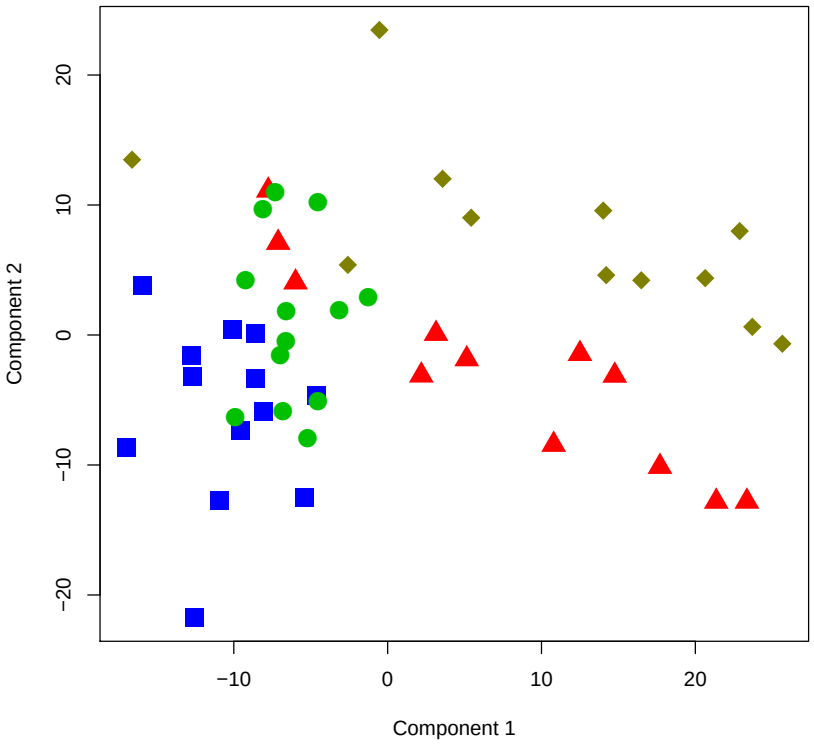
Simply averaging over the replicates however produced a noticeably erratic time-profile and trend-based smoothing methods were employed to counter this. These will be discussed in more detail in Chapter 5.

Variables in each dataset were ranked by their absolute correlation scores of the above response patterns. As an example, Figure 3.8 shows the variable with the strongest correlation (in both PLSR and Pearson regression) with the $\mathcal{D}/\mathcal{B}$ response template in the $\mathcal{L}+/-$ dataset. It is important to note however, that correlation and regression only help to identify those variables matching the pattern crafted in the response variables, which in this case is a linear one. Non-linear or quadratic patterns may appear as less significant or may be missed. Correlation nonetheless serves as a simple but useful starting point to visualise the sort of patterns present in the data. The set of top results for the correlation analysis is shown in Appendix A.

These parameters were found to offer the best smoothing without significantly degrading the pattern seen in the individual variables. Figure 3.9 shows the scores plot for first two PLSR components, demonstrating how averaging can clarify the results by contrasting the averaged and non-averaged ("complete") datasets.



(a) Full $\mathcal{L}+/-$ dataset



(b) Averaged $\mathcal{L}+/-$ dataset (method = moving median, window width = 5)

Fig. 3.9: PLSR scores plots for the dataset for (a) all replicates and (b) averaged across replicates. Icons and colours denote experimental group. ■: Control group, ▲: Drought group, ●: Fusarium group, ♦: Dual-stress group. The average dataset provides a notably clearer set of results.

3.2 Conclusions

There are a clear number of drought responsive peaks within the *Medicago* dataset, with a large number of dual-stress responses following the same pattern as the drought response. The standard analyses performed were moderately effective at identifying a number of these peaks, they do however, suffer from a number of drawbacks. Primarily, the intensity profile of the peak over time must be provided in, and in this case only a linear response was tested. Whilst testing a set of linear correlations for the conditions of interest did reveal a number of candidate biomarkers displaying non-linear profiles, it is evident from manual examination of the dataset that a number of others may have been missed. Furthermore, a high number of drought-responsive compounds are present in the data, which complicates the task of finding biomarkers to target for further analysis. For instance 14% (412 out of 2920) of the peaks in the combined leaf dataset show "significant" correlation with a linear trend over time for the drought group only ($Q = \mathcal{D}$), where $p \leq 1.71 \times 10^{-5}$, which is equivalent to $p \leq 0.05$ after applying the Bonferroni correction to account for multiple-testing, where $n = 2920$. Despite applying this correction the issue remains somewhat of a "fishing expedition" and was performed largely by a manual inspection of the data that confirmed that peaks showing these linear trends, did in-fact exist. Chapter 5 attempts to redress some of these issues through the use of unsupervised cluster analysis in order to determine common "patterns" of metabolites in the data. Issues with noise and in accounting for "growth" related variation seen in the control groups will also be addressed.

3.3 Beef dataset: Using ^1H NMR to identify the storage conditions of matured beef

3.3.1 Introduction - The case for Beef

In 2015 the EU produced 7.6 million tonnes of beef and veal set for human consumption and the UK beef market alone was valued at £2.7 billion [252]. Post-slaughter, beef is aged to produce the desired flavour and tenderness, with increased ageing time producing a stronger "beef" flavour. However, longer-aged beef also commands a higher price; as beef is aged, weight loss is to be expected from evaporation and additional costs are incurred as

valuable refrigerated space is used [253]. This presents potential scope for abuse, whereby short-aged beef produce is deliberately mislabelled as long-aged and then sold at increased price.

The effects of ageing in beef samples have been previously studied in [254] and [255]. Using H^1 NMR the authors were able to identify 12 amino acids and confirm earlier studies, showing increases in the concentrations of all identified amino acids over the ageing period. The authors also compared meat from carcasses hung using two methods (Achilles tendon and Pelvic suspension) the latter of which has been reported to increase meat tenderness [256]. However, they reported that they did not detect any significant difference in the metabolic composition in the meat obtained from each of the methods.

The increase in amino acid composition over time is a potential indicator of the duration of ageing and therefore presents an avenue by which mislabelling could be identified. However, since chemical reactions generally progress faster at higher temperatures a reasonable hypothesis is that in by increasing the storage temperature protein breakdown can be accelerated to the point where short-aged warm-stored meat becomes less dissimilar from long-aged meat under proper (cool) storage conditions. The first goal of this study is to investigate this hypothesis and attempt to identify a mechanism whereby meat stored under two different storage conditions could be identified. This would allow premium "21 day matured beef" to be distinguished from beef stored for a much shorter period under less than ideal conditions.

3.3.2 Materials and methods

The dataset consists of 361 NMR spectra for meat aged in three different storage conditions between $t = 1$ and 28 days:

- Frozen ($\mathcal{F}$) – stored at -20°C
- Ambient, warm ($\mathcal{W}$) – stored at room temperature 20°C
- Ideal, cold ($\mathcal{C}$) – stored in a refrigerated environment at 4°C

Day zero ($\mathcal{Z}$) samples are also present for meat at $t = 0$ which has not yet been stored.

3.3.2.1 Preprocessing

Data was split into bins using the adaptive binning procedure proposed in [111] using the Metabolab tool-kit for Matlab. Using a 3 level wavelet transform this resulted in a total of 1234 bins not categorised as noise.

3.3.3 Initial analysis and discussion

3.3.3.1 Scaled vs. unscaled data

Scaling has the potential to inadvertently amplify the effect of noise in the dataset by bringing low-intensity noise peaks in line with the rest of the data. However, the adaptive binning algorithm used has the advantage of automatically detecting noisy areas of the NMR spectrum and discarding them from the resultant binned data. Initial tests with scaled and unscaled versions of the dataset, including PCA and PLS, indicate better separation and discrimination between experimental groups for the scaled data. With the exception of the early outlier detection noted below, all analysis was carried out on data that was mean centred and scaled to unit variance.

3.3.3.2 PCA

The PCA analysis of the dataset, indicates three outliers emphasised in the second principal component, observations #222, #223 and #224. These outliers are all replicates from the same beef sample (`S08-029333`), representing the warm storage group and with an age of 21 days. In a room temperature environment it is possible that this beef sample became infected to a degree where its chemistry was significantly altered. These three observations were discarded from further analyses in order that the study concentrate on more subtle differences between storage groups. A fourth outlier, observation #353 was identified in the frozen group after auto-scaling the data as outlined above and re-performing the PCA. Manual inspection of the NMR spectrum for this sample indicates a significant difference in the amplitude of many of its peaks, the reason for which is unknown and could be due to a variety of factors, including biological and analytical. This outlier was also discarded from the dataset so as not to confound further analysis.

A PCA plot of the auto-scaled data with outliers removed is shown in fig-

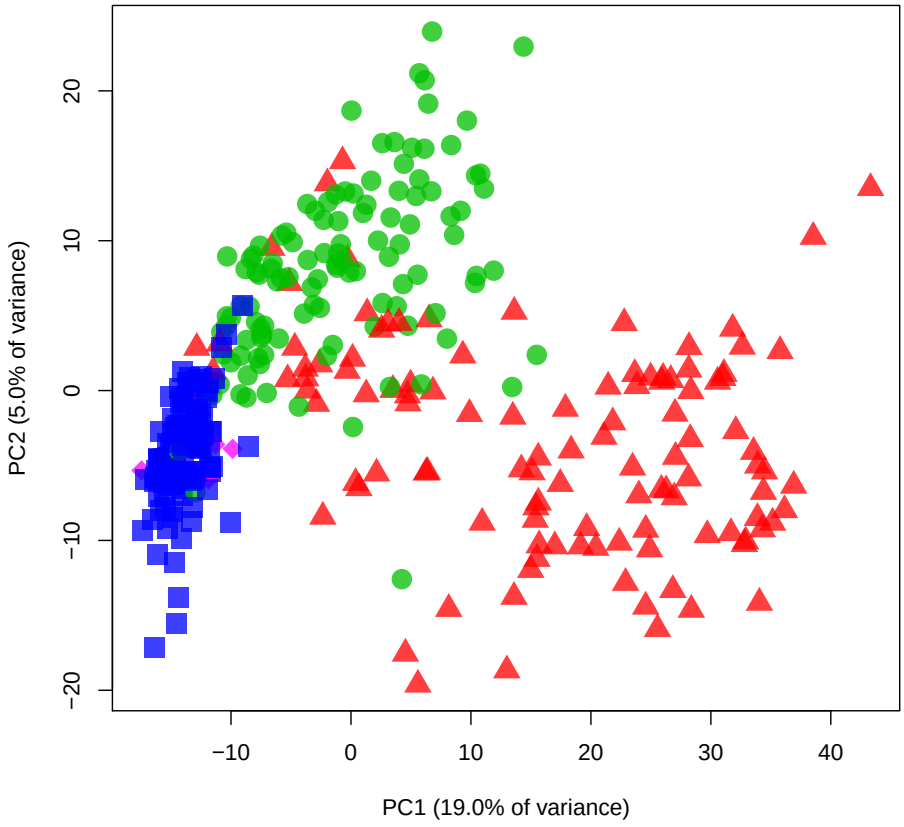


Fig. 3.10: Plot showing the scores along the first two components resulting from PCA of the dataset. The four outliers were removed and the data scaled to unit variance and mean centred prior to performing PCA. Separation between the three experimental groups is apparent along both components. Icons and colours denote experimental group. ♦: Day zero samples, ■: Frozen samples, ●: Cold samples, ▲: Warm samples.

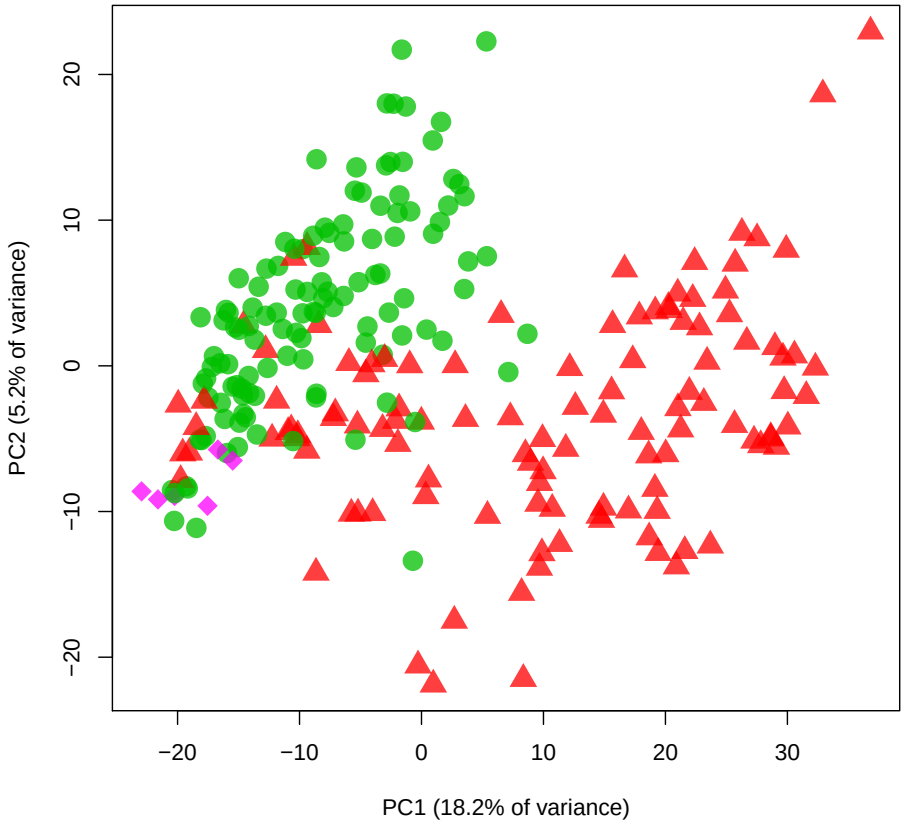


Fig. 3.11: Plot showing the scores along the first two components resulting from PCA of the dataset. The four outliers and all observations from the frozen group were removed prior to PCA. The data was scaled to unit variance and mean centred. Icons and colours denote experimental group. ♦: Day zero samples, ●: Cold samples, ▲: Warm samples.

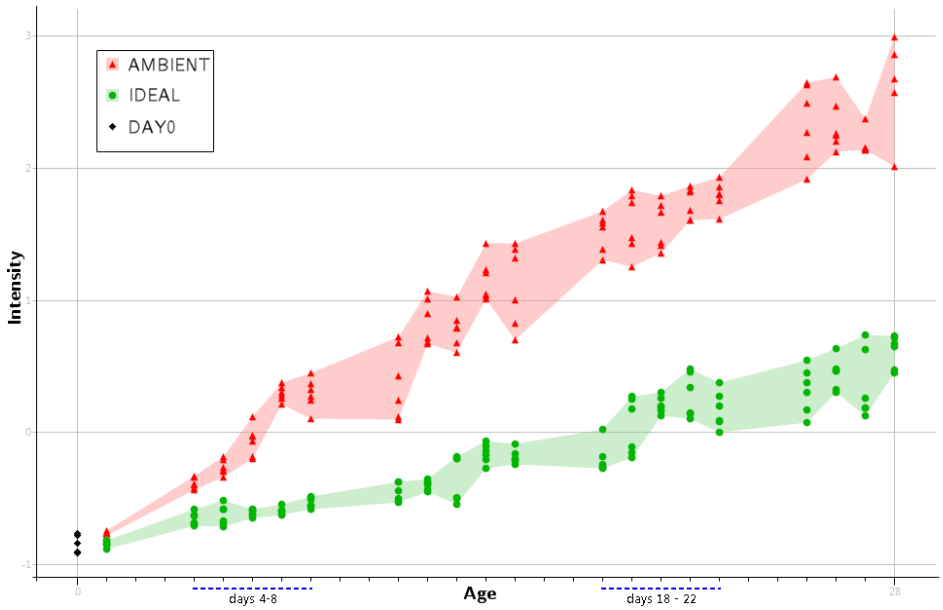


Fig. 3.12: Plots of intensities against age for a single variable (bin) of the beef dataset. The bin is centred at 1.72ppm. The points denote the replicate observations, with icons and colours signifying experimental group. ▲: Warm, ●: Cold, ♦: Day zero. The shaded regions highlight the intensity range for each set (time and age) of replicates. It can be seen that there is a region where the intensities of the peaks from the two experimental groups overlap. For instance, the early warm-storage samples, and the late cold-storage samples, highlighted by the two blue dashed lines.

ure 3.10. It can be seen that the storage method is a dominant factor leading to variation between observations. The difference between warm samples, in contrast to the day zero samples, is obvious along the first principal component. Separation of the cold-group samples is apparent along the second principal component, and to a lesser degree along the first principal component. The frozen storage group meanwhile shows heavy overlap with the day zero samples. This pattern of changes fits well with prior expectations: metabolic activity will increase with temperature. The greatest source of variance along PC1 is due to the difference between the warm samples and the day zero samples, whilst the greatest source of variance along PC2 is due to the differences in the cold group. At $-20\,^{\circ}\mathrm{C}$ metabolic activity will have slowed down significantly and little change from the starting position occurs in the frozen group. After excluding the frozen group from the PCA analysis the plot shown in figure 3.11 is obtained. Whilst the warm and cold groups show differences, there remains some overlap between them, which is most evident between the older (age 20-28) cold-group samples and the younger (age 4-10) warm ones. Younger samples are naturally harder to separate since they have deviated less from the day zero ones. These behaviours appear typical when individual variables are scrutinised. Figure 3.12, shows the intensity of bin #666 as an example, with the intensity of the the samples plotted against age. Overlap between the intensities of the two experimental groups can be easily be seen, for instance between ages 4-8 of the warm group and 18-22 of the cold group, among others. This behaviour is somewhat unsurprising given the general increase in reaction rates with temperature. A single biomarker representative of the storage method could offer a fast chemical test to identify storage. However, unfortunately the existence of the overlapping region compromises the ability to easily discern group from a single variable as almost all variables show some region of overlap. This is emphasised further in the loadings plot, which indicates that the separation which does exist between groups in the plot is the compound effect of many variables, with no small group of bins dominating the loadings.

3.3.3.3 SPCA

In order to determine if a smaller set of bins could be used to achieve a separation between groups similar to the one seen for PCA, sparse principal

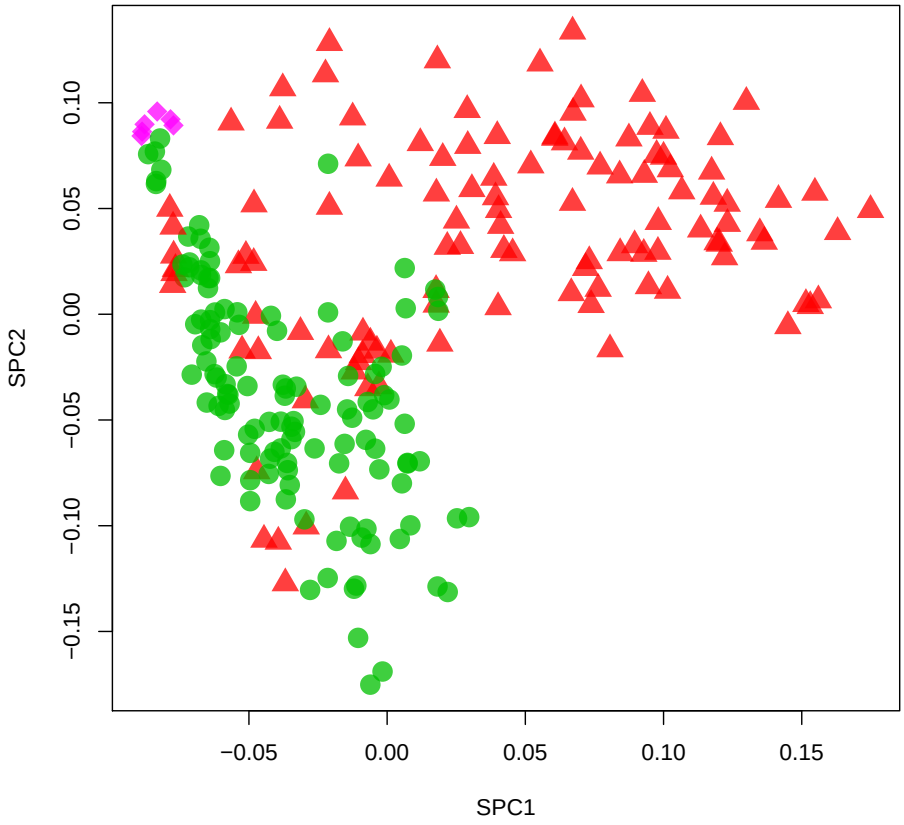
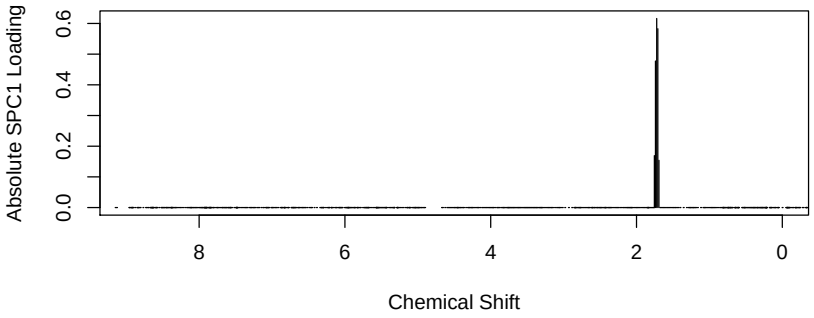
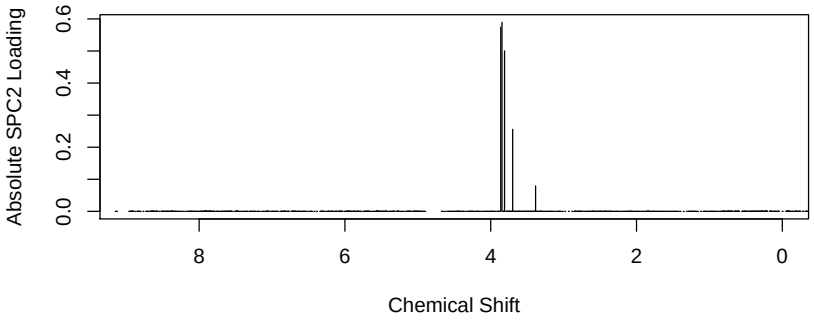


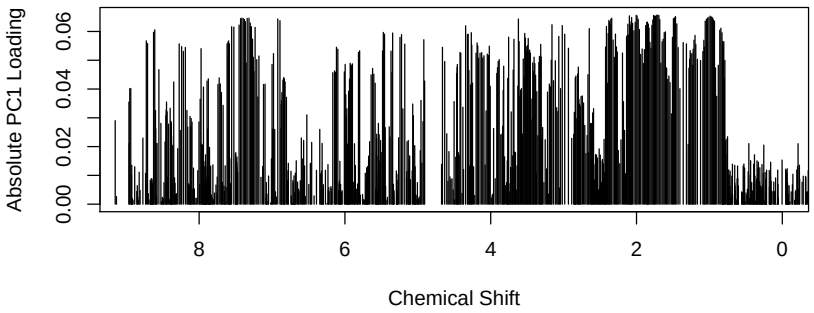
Fig. 3.13: Sparse principal components analysis of the beef data, excluding the frozen group and 3 outliers. Icons and colours denote experimental group. ♦: Day zero samples, ●: Cold samples, ▲: Warm samples. SPC1 accounts for 0.32% of the total variance and SPC2 for 0.30%. Despite the sparse constraint on the analysis, separation of the experimental groups is still notable.



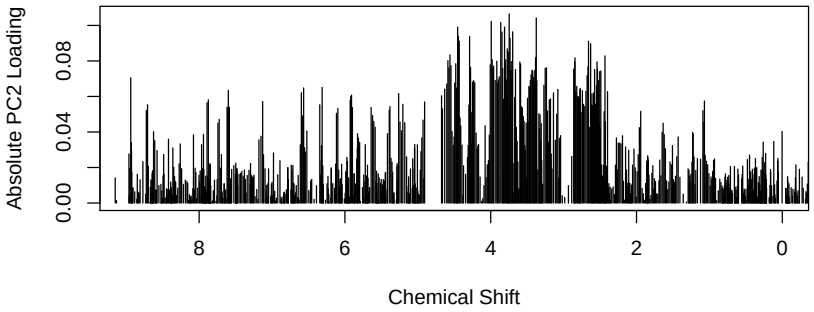
(a) SPC1



(b) SPC2



(c) PC1



(d) PC2

Fig. 3.14: Plots showing how the two PCA methods differ in their loadings (rotations) plots. The absolute values of the loadings have been taken for readability. The top figures show the absolute loadings of SPCA, and the bottom two of PCA. The SPCA loadings have are noticeably cleaner than those of PCA. The variables with the highest loadings in both cases remain similar.

components analysis (SPCA) was performed. SPCA, previously described in chapter 2, follows the general pattern of standard PCA but applies a penalty to non-zero loadings. This has the effect of producing a result with fewer variables contributing towards the model, which is arguably simpler to understand. The number of variables per principal component is regulated by a sparseness parameter, $k = \sum_i(|v_i|)$ where v_i denotes the ith variable used. In this case the parameter was chosen empirically to produce visually good results. A very sparse constraint ($k = 2$) resulted in reasonable separation between groups, the scores plot for which can be seen in figure 3.13. This resembles the results of standard PCA, with the warm group deviating along the first principal component, and the cold group along the second. The loadings plots from the PCA and SPCA analyses are shown in Figure 3.14. It can be seen that, unlike the standard PCA loadings however, SPCA produces a concise set of variables complementing the scores plot, with a set of loadings dominated by a single spike, centred around $V_{666} = 1.72$*ppm (2-Hydroxybutyrate = 1.7)*. The second principal component reveals a similar pattern, dominated by a spike centred around $V_{486} = 3.85$*ppm (Inosine = 3.84, Adenosine = 3.8, Glucose = 3.8, Mannose = 3.84)*.

The bin at 1.72ppm was shown earlier in figure 3.12. This bin also shows the highest correlation with time for the warm experimental group samples ($p_{pearson} = 0.949$). The bin at 3.85ppm, shown in figure 3.15, conversely, has an intriguing pattern where older samples show an increase in concentration in the cold group while showing a marginal decrease over time for the warm group. However, these two variables alone are not able to distinguish between the early warm and later cold-group samples.

3.3.3.4 Univariate testing

Performing t-tests on the binned data highlights 442 out of 1234 bins as showing "significant" ($p < 0.05$) differences between samples of all ages in the warm group against samples of all ages in the cold group. However, late samples are easily distinguished and it is the late cold-group and early warm-group samples for which discrimination is difficult. t-tests were performed using different age ranges from these overlapping groups and potential biomarkers such as the bin at 7.60ppm, shown in figure 3.16, were identified. However, although these bins represent chemicals that are present (i.e. non-zero) in one group but not the other, the peaks are of low intensity, which

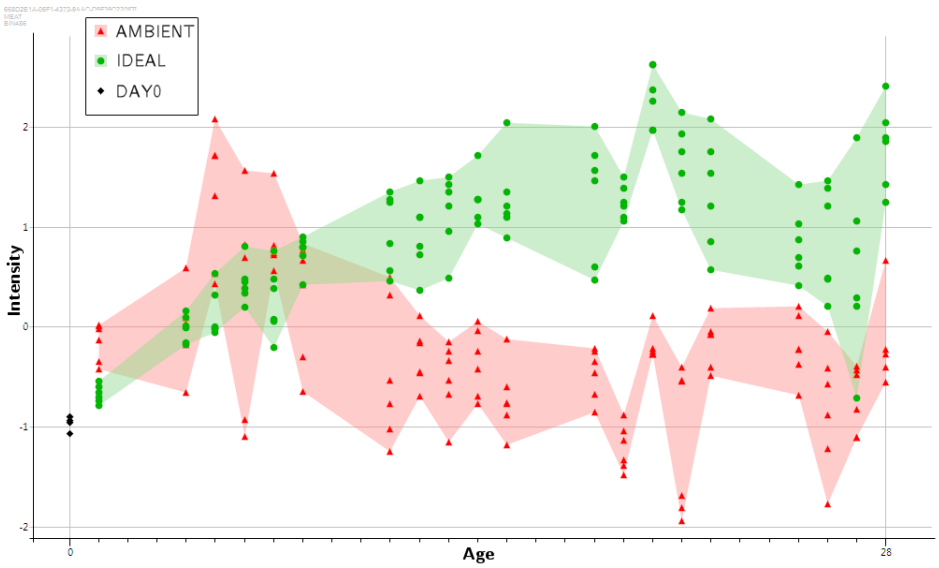


Fig. 3.15: Plots of intensities against age for a single variable (bin) of the beef dataset. The bin is centred at 3.85ppm. The points denote the replicate observations, with icons and colours signifying experimental group. ▲: Warm, ●: Cold, ♦: Day zero. The shaded regions highlight the intensity range for each set (time and age) of replicates. This variable shows an "inverted" profile, where the intensity of the cold group samples tend towards exceeding the warm group samples as time (age) increases.

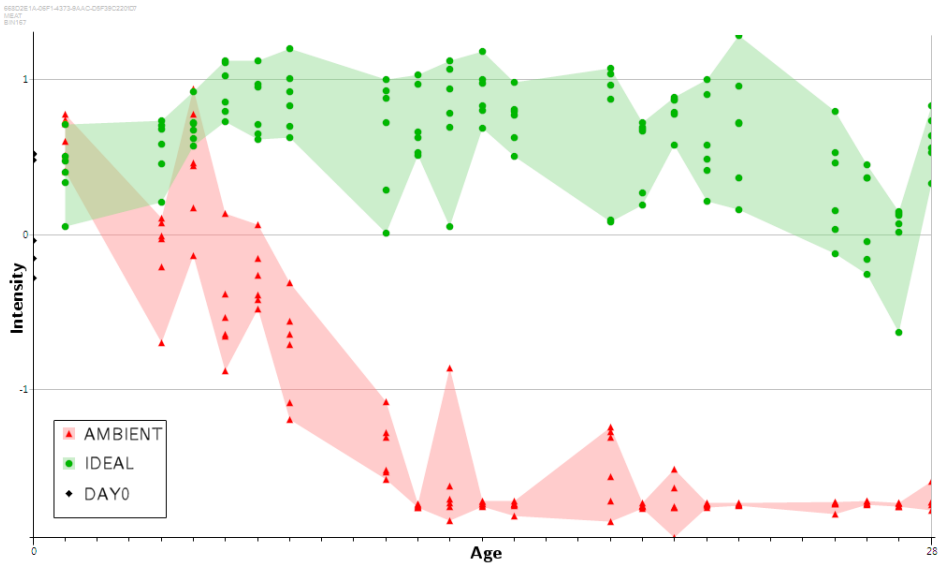


Fig. 3.16: Plots of intensities against age for a single variable (bin) of the beef dataset. The bin is centred at 7.60ppm. The points denote the replicate observations, with icons and colours signifying experimental group. ▲: Warm, ●: Cold, ⧫: Day zero. The shaded regions highlight the intensity range for each set (time and age) of replicates. This variable shows one of the most significant differences between the warm and cold experimental groups, however, the low intensity of the variable suggests that this may be an artefact of the detection method, with the intensities lying on the border of the limit of detection.

may be indicative of low chemical abundance. Thus it may not be possible to identify them using all instrumental platforms. Furthermore, the disappearance of these low intensity markers could be an artefact of the detection method with a zero-intensity value simply meaning that the chemical concentration has fallen below the limit of detection, but not necessarily that the chemical species is truly absent from the sample. In order to fully determine these potential of these species as a biomarker further scrutiny of their true concentrations over time would need to be performed using a more sensitive method.

3.3.3.5 Classification

From the unsupervised PCA and SPCA analyses it can be seen that there are notable distinguishing features between the warm and cold groups. However almost always there exists some degree of overlap between the two groups. It is possible that through a combination of two or more variables the groups could be distinguished.

Due to the ratio of variables to observations in the dataset (i.e. $n_{vars} >> n_{obs}$ it is expected that supervised multivariate techniques may have the effect of presenting an artificial bias in the results. However the visible separation from PCA and SPCA suggests that there already exists a natural degree of separation between the experimental groups which warrants further attention.

3.3.3.6 PCA-LDA

As a means of gauging the separation of the groups in PCA, LDA can be used on the scores. To avoid overfitting, leave-one-out (LOO) cross-validation was used. In this approach, one observation was left out of the analysis and the PCA scores and LDA classification then made for the missing observation and the classification added to the results. This was performed for all observations in turn. Table 3.2 presents the results as a confusion matrix. Whilst the overall predictive accuracy is reasonable, at just under 87%, of the incorrect predictions 84% are warm group samples incorrectly predicted as cold group ones. It can be seen that the set of warm group samples predicted incorrectly are all 8 days or younger, whilst the cold-group samples predicted incorrectly are all 25 days or older. This gives a more objective weight to the

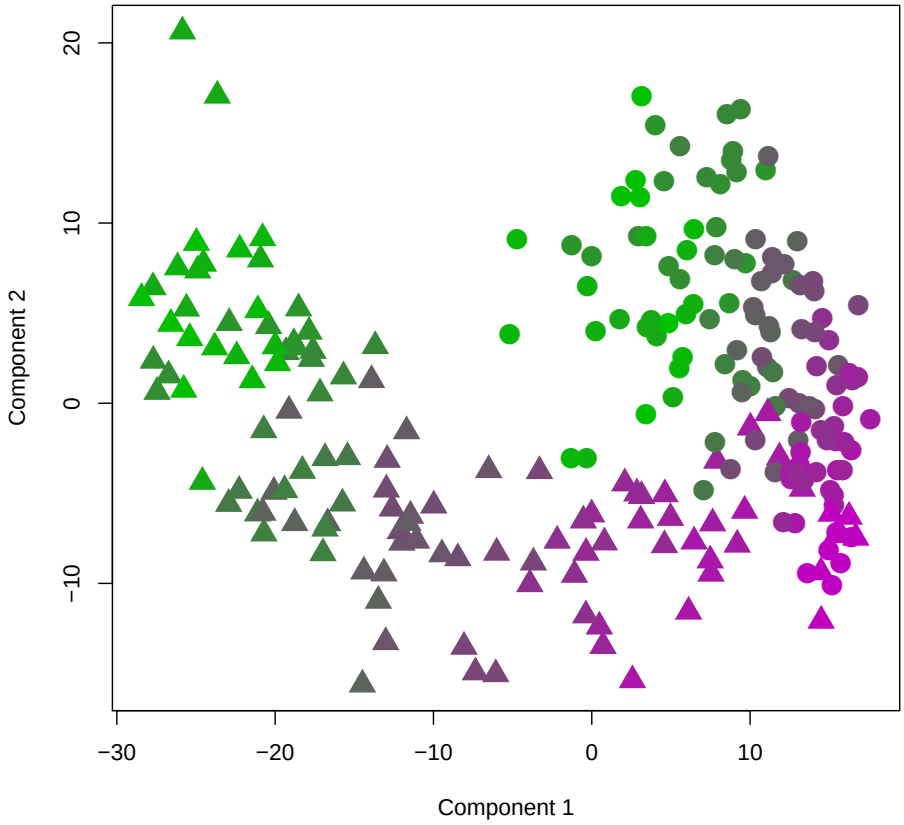


Fig. 3.17: PLSR scores plot for first two components of the Beef dataset. Icons denote experimental group. ●: Cold, ▲: Warm. Colours denote age. ■: Young (day 1), ■: Old (day 28).

argument that short, ambient (warm) storage conditions can emulate longer ideal (cool) storage conditions. This also suggests a more general pattern to the individual inspection of variables seen earlier, in which which indicated a certain "band" at which samples from both groups become difficult to distinguish can be seen.

3.3.4 PLSR

PLSR was performed on the cold and warm group samples using the class as the response matrix, where $I = 0$ and $A = 1$. The first two components

Actual		Predicted		
Class	Day	$\mathcal{W}$	$\mathcal{C}$	(both)
$\mathcal{W}$	1	0	6	6
	4	1	5	6
	5	0	6	6
	6	2	4	6
	7	2	4	6
	8	5	1	6
	11	6	0	6
	12	6	0	6
	13	6	0	6
	14	6	0	6
	15	6	0	6
	18	6	0	6
	19	6	0	6
	20	6	0	6
	21	3	0	3
	22	6	0	6
	25	6	0	6
	26	6	0	6
	27	6	0	6
	28	5	0	5
	(all)	90	26	116
$\mathcal{C}$	1	0	6	6
	4	0	6	6
	5	0	6	6
	6	0	6	6
	7	0	6	6
	8	0	6	6
	11	0	6	6
	12	0	6	6
	13	0	6	6
	14	0	6	6
	15	0	6	6
	18	0	6	6
	19	0	6	6
	20	0	6	6
	21	0	6	6
	22	0	6	6
	25	1	5	6
	26	1	5	6
	27	3	3	6
	28	0	6	6
	(all)	5	115	120
(both)	(all)	95	141	236

Tab. 3.2: Confusion matrix showing the PCA-LDA prediction results using LOO-CV. The results presented are those of the validation set. Whilst most predictions are correct, it can be seen that the predictions of the $\mathcal{W}$ samples are typically incorrect for the early-day sample range (days 1-8), whilst for the $\mathcal{C}$ samples, it is the the late-day samples (days 25-28) that are typically predicted incorrectly.

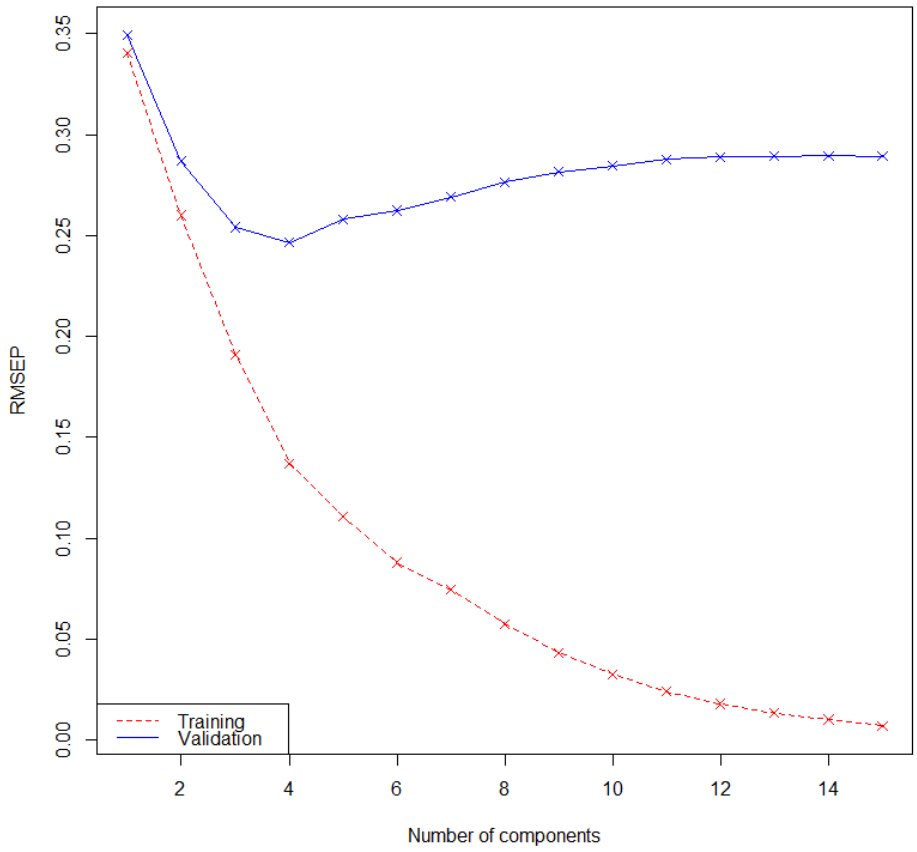


Fig. 3.18: RMSEP for LOO-CV of PLSR. The RMSEP is shown up the Y-axis, with number of components displayed along the X. Lower RMSEP values indicate increased accuracy of prediction. It can be seen that whilst the RMSEP of the training set decreases as the number of components increases, the RMSEP of the validation set reaches a minimum at the fourth component. An increasing number of components thereafter decrease the accuracy of the validation set predictions, indicative of overfitting the PLSR model to the training set.

of the results are shown in figure 3.17. Although the separation between the experimental groups appears to be quite good there remains a degree of overlap between the two groups. A trend with age can also be observed, whereby as the age of the samples increases the separation between groups becomes more evident. The early samples in particular show a large overlap, where up to approximately $t = 5$ for the warm-storage group and $t = 14$ for the cold group the two storage methods follow a similar trend along the first two components. In order to identify the significant variables in the classification VIP (variable importance in projection) scores were calculated.

Leave-one-out (LOO) cross validation was performed to produce a RMSEP (root-mean-squared error of prediction) plot in order to determine the ideal number of components to avoid overfitting the data, in this case 4. The plot is shown in figure 3.18. The 10 highest VIP scores correspond to bins centred around 3 peaks, positioned at 5.91ppm, 7.60ppm and 8.71ppm, in order of importance. The bin at 7.60ppm is the one previously identified using univariate tests and has a mean intensity of less than 1% of the overall mean.

3.3.5 Conclusions

A number of peaks have been identified which offer degree of separation between our experimental groups of interest. However there remains a problematic subset of the observations which, in conjunction with the rest of the observations, cannot be resolved by the standard discrimination methods attempted. This creates a gap in which the use of other methods may be attempted. Chapter 7 will explore the ability of strongly-typed genetic programming in the classification of the data and the production of a succinct set of variables responsible for the classification.

3.4 Alopecurus dataset: An investigation into the herbicide resistance of Black Grass (Alopecurus myosuroides)

3.4.1 Introduction

Alopecurus myosuroides, commonly known as black grass, is a major weed affecting cereal crops in Europe. The treatment for infestations has traditionally been the use of herbicides, primarily for economic reasons [257].

As with many weed species, herbicide resistant varieties are becoming increasingly prevalent and their presence has been reported throughout the continent [258]. A number of different mechanisms of resistance have been documented in the literature, including Acetyl-CoA (ACCase) inhibitors [259], acetolactate-synthase (AS) inhibitors [260] and photosystem II (PSII) inhibitors [261]. Target site resistant varieties such as these contain mutations such that the binding of the herbicide to the target site is prevented, although other mechanisms are also present in some weed varieties, including changes to metabolism and sequestration of the herbicide [262]. Increasingly under scrutiny are multiple herbicide resistant varieties, which utilise more than one mechanism of action [261, 263].

Application of an ineffective herbicide entails unnecessary temporal, economic and environmental costs. Early identification of the phenotype of the invading species is therefore of high priority. The current study monitors three varieties of black grass over a period of two weeks with an aim to locate type-specific biomarkers for herbicide resistance. Such biomarkers can provide a potential means of quickly identifying the resistance capabilities of an invading strain for farmers to use, as well as offer a potential locus for further investigation into the resistance mechanism.

3.4.2 Experimental

The study involves three varieties of black grass (the experimental groups):

- *Susceptible* (non-resistant)
- *Target site resistant* (MHR)
- *Multiple herbicide resistant* (TSR)

Three biological replicates for each experimental group were harvested at 4 different time points: 0, 4, 8 and 13 days, giving a total of $3 \times 3 \times 4 = 36$ biological samples. The samples were then subject to LC-MS analysis. Feature extraction of the resultant spectra was performed using Progenesis QI, giving a total of 3458 variables representing potential metabolites, henceforth referred to as “peaks”.

3.4.2.1 Plant materials and growth conditions

MHR (Essex, UK.), TSR (Nottingham, UK.) and Susceptible (Rothamsted, UK.) seed lines of *A. Myosuoides* were planted into 12 cm diameter terracotta pots in a peat based compost (Petersfield, Cambridge, UK.) using 20 seeds per pot. Plants ($n = 3$ per seed line) were grown under controlled glasshouse conditions at Fera Science Ltd. (Fera, North Yorkshire, UK). There were 24 plants in total which contained 3 replicates for each line over 4 time points - Day 0 (pre-treated samples), Day 4 (post spray), Day 8 and Day 12. After 3 weeks of growth, plants were sprayed at field rates with Topik (an acetyl co A carboxylase inhibitor) and harvested directly into liquid nitrogen cutting from just above the soil line.

3.4.2.2 Metabolite extraction method

Each frozen plant was lyophilised overnight and ground into a fine powder using an A 11 basic analytical mill (IKA, Staufen, Germany). Five mg ± 0.1 mg of ground sample was accurately weighed into a labelled 2 mL eppendorf tube. To 5 mg of sample, 1ml of extraction solvent (1:1 (v/v) methanol: water) was added. Metabolites were extracted into the solvent by shaking for 30 minutes. The solid material was then removed by centrifugation at 14,000rpm for 10 minutes at ambient temperature. To prepare samples for profiling analysis by liquid chromatography–high resolution mass spectrometry (LC-HRMS) the supernatant was diluted 9:1 with 1:1 (v/v) methanol: water. An analytical QC sample was created by pooling 1 ml from each final sample extract.

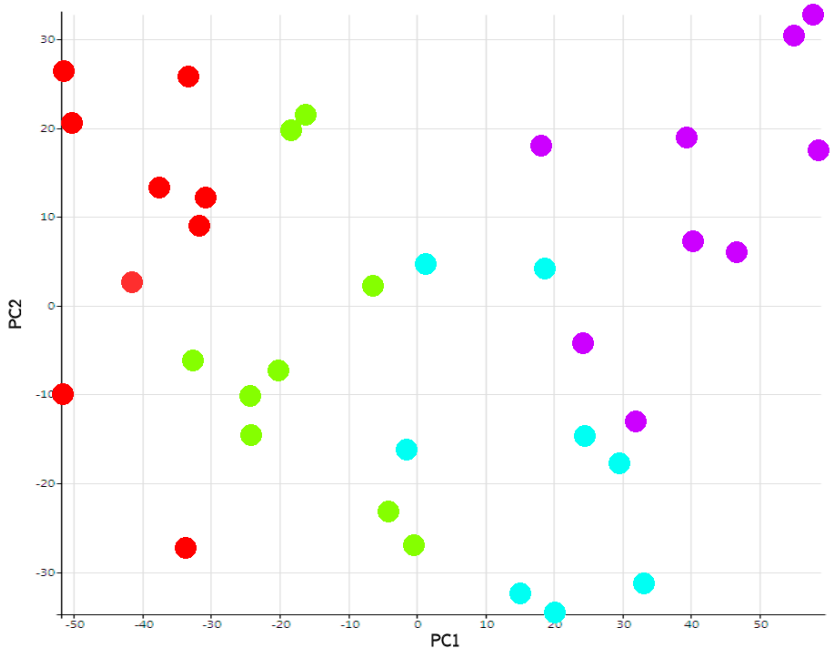
3.4.2.3 LC-HRMS profiling conditions

LC analysis was performed on an Accela 1250 High Speed LC system from Thermo Fisher Scientific (Waltham, Massachusetts, USA). The analytical column used was an ACE Excel AQ (Advanced Chromatography Technologies, UK) 150 mm x 3 mm, 100 Å. MPA was 0.1% formic acid in HPLC water, MPB was 0.1% formic acid in acetonitrile. A linear gradient elution was applied over 10 minutes from 100% MPA to 100% MPB. The gradient was then held for 2 minutes at 100% MPB before re-equilibration with 100% MPA for a further 2 minutes. The LC flow rate was 0.4 ml/min and the column temperature was 30 °C. Sample injection volume was 5 µl. The MS

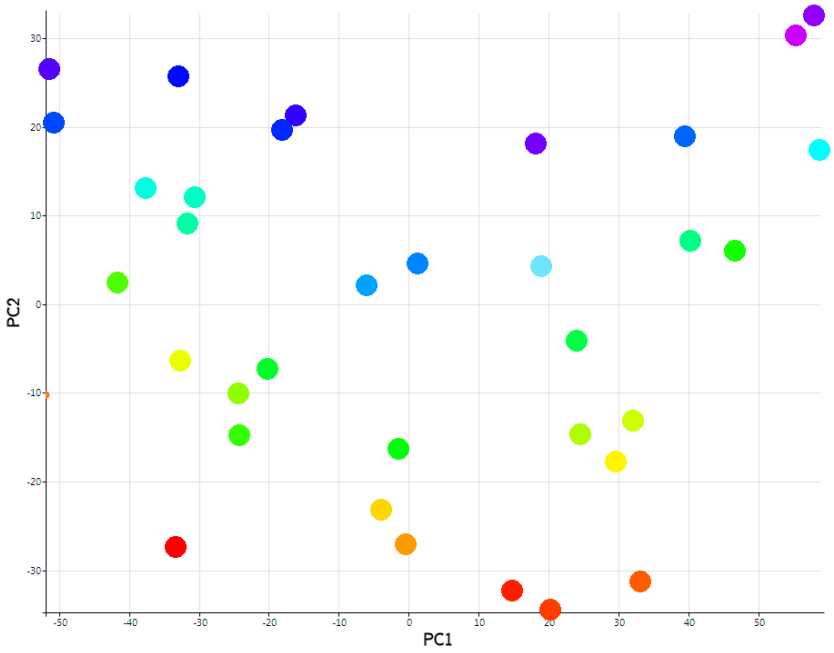
used was an Orbitrap Velos Pro hybrid ion trap high resolution mass spectrometer (Thermo Fisher Scientific, Waltham, Massachusetts, USA) with a mass resolution setting of 60,000 at m/z 200. Maximum injection time was 50 ms. Ionisation was by heated electrospray (HESI) with extracts analysed in both positive and negative mode. The source heater temperature was set to 450 °C with sheath gas set to 51 and aux gas at 16 (au, arbitrary units). The capillary temperature was 370 °C. Sample analysis order was randomised using `www.random.org` and a pooled QC sample was injected every 6 injections to monitor system performance.

3.4.3 Initial analysis

The PCA plot for the first two principal components obtained for the black grass dataset is shown in Figure 3.19, coloured according to age in (a) and acquisition order in (b). Although the major source of variance is age, separation also occurs due to the order in which the samples were acquired by LC-MS, as can be seen along PC2. Differences in experimental group can be seen in the third and fourth principal components (not shown), however the variance due to acquisition order presents a cause for rectification of the data. Performing a batch correction by fitting a linear model to the set of QCs for each metabolite resulted in some, but not complete reduction of the acquisition order trend as can be seen in Figure 3.20a .

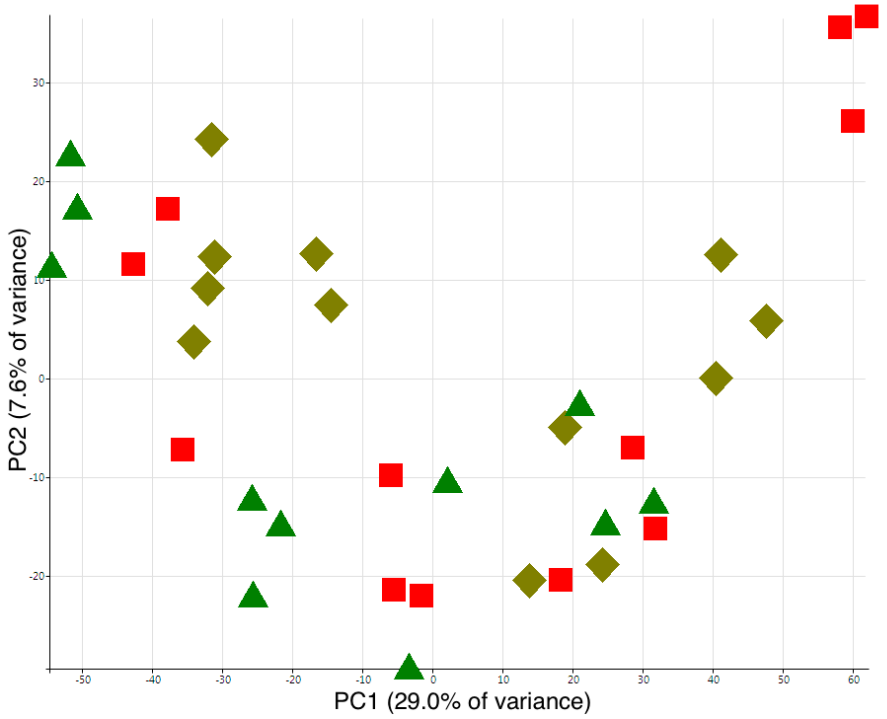


(a) Age. ●: Day 0, ●: Day 4, ●: Day 8, ●: Day 12

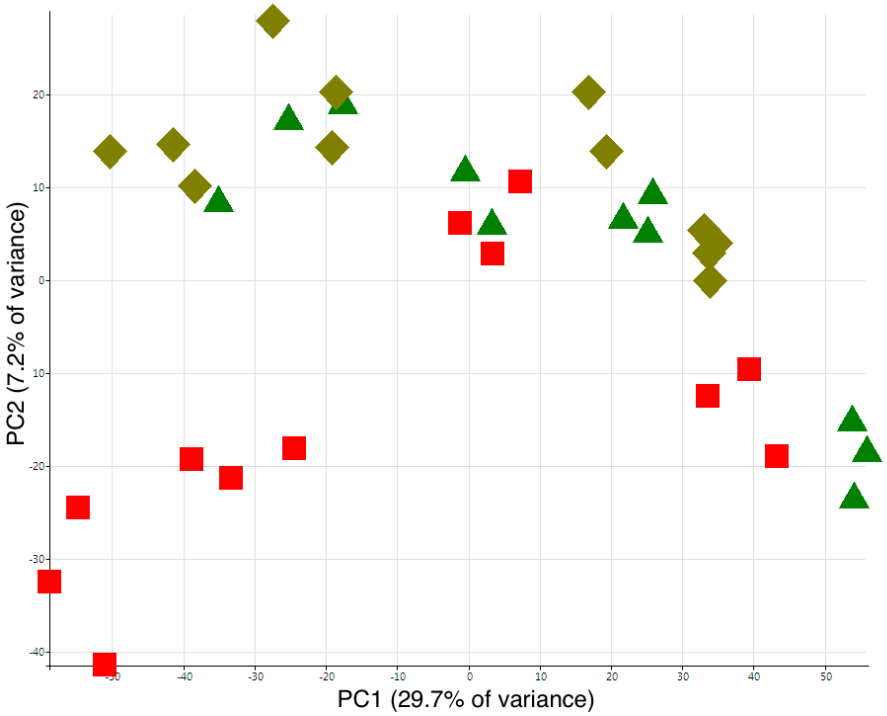


(b) Acquisition. ●: First ... ●, ●, ●... ●: Last

Fig. 3.19: Plot showing the PCA scores plot for the autoscaled *Alopecurus* dataset, highlighted according to (*a*) sample age and (*b*) acquisition order. Although age is the source of the major variance (PC1 = 27.7% of variance), the order in which the samples were acquired via LC-MS produces the second major source of variance (PC2 = 10.4% of variance).



(a) QC



(b) BG

Fig. 3.20: Plot showing the PCA scores plot for the autoscaled *Alopecurus* dataset, corrected using (*a*) mean-of-the-QC and (*b*) background correction methods. Icons and colours represent experimental group. ■: Susceptible, ▲: TSR, ♦: MHR. Experimental groups are notably more grouped for the BG correction method.

4. BATCH CORRECTION

4.1 Introduction

Non-targeted metabolomic studies seek to analyse as wide a range of metabolites as possible. The use of LC-MS for this purpose has found a wide range of applications, including drug discovery [264], disease biomarker discovery [265], pesticide [266] and herbicide [267] analysis in agriculture, waste-water analysis [268] and the discovery of novel metabolites [269]. LC-MS however suffers from lower reproducibility in comparison to other analytical techniques such as NMR spectroscopy [31, 270]. Many non-targeted approaches focus on qualitative results, such as biomarker discovery, and the need for reproducible and comparable results is imperative, especially when differences between experimental groups are small. A number of factors can cause differences in LC-MS response profiles between acquisitions. Many of these relate to chromatographic aspects, such as retention time drift or changes in peak shape [271], but changes in the response of the mass spectrometer can also be seen [272]. Most notable are the changes occurring during the acquisition of a multi-sample experiment due to the gradual contamination of the LC column. Whilst effective cleaning, conditioning and calibration of the instruments can mitigate these problems to a degree, consecutive analysis of large numbers of samples has been shown to present increasingly unacceptable variation [134]. Samples are therefore often run in batches, interspersed with the relevant cleaning and conditioning events. However, this can lead to other sources of technical variation, such as differences in the operating conditions under which the acquisitions of the individual batches are performed.

Further sources of variation may be introduced in the early stages of data analysis. Although advances in methods of spectral alignment can reduce the effects of retention time drift and changes in peak shape, such methods do not always provide a complete solution in non-targeted studies

involving thousands of potential metabolites. Spectral misalignment prior to the peak-picking stage can result in the classic problems seen in spectral binning, with differences between spectra being due to misaligned peaks rather than true changes in intensity. A widely implemented solution to these problems is the inclusion of quality control (QC) samples into the study. During data acquisition the experimental samples are interspersed with a set of identical QC samples, providing a fixed reference point from which any instrumental variation can be tracked and later accounted for. The QC samples should contain the same metabolites as are under scrutiny in the study, being either a mixture of known laboratory grade analytes, or a pooled sample from the experiment itself. The former allows easier identification and quantitative analysis, whilst the latter allows as wide a range of metabolites as is attainable to be evaluated and is naturally more suited for non-targeted analysis. Should insufficient experimental samples be available for pooled samples, biologically similar samples may also provide reasonable QC data [273, 274].

At the very least QCs can be used to gauge the reliability of the measurements for the individual metabolites. For example, in a GC-MS (gas chromatography-mass spectrometry) study, Begley et al. (2009) [275] only accept individual metabolites where the relative standard deviation (RSD) of the QCs is less than 30%. In another study involving DIMS (Direct Infusion Mass Spectrometry), Kirwan et al. (2013) [276] use a limit of 20% RSD with the additional criterion that the distribution of the QC samples be similar to that of the experimental ones. Other criteria have been proposed, for example that QC values should lie within 15% of their mean [275, 277].

However, since many sources of variation pertinent to the sample metabolites also apply to the QC metabolites, the function of the QC samples can be extended to correct for variation, rather than just quantify it. To do this a correction factor must be determined, for each metabolite and sample. Van Der Kloet et al. (2009) [274] list several methods to achieve this, although the general form of the correction follows Equation 4.1:

$$X'_{p,b,i} = X_{p,b,i} \frac{R_p}{C_{p,b,i}} \tag{4.1}$$

Here $X_{p,b,i}$ is the intensity of peak p for sample i within batch b, prior to correction, and $X'_{p,b,i}$ is the corrected value. $C_{p,b,i}$ represents the correction

factor and R_p represents a rescaling factor which allows the relative intensity of the peak to be maintained. We refer to the set of correction factors, C, for a particular peak as the *trend* for that peak. The simplest correction is to divide a peak within a sample by the average intensity recorded for that peak in the QC samples in the same batch as the sample, so that

$$C_{p,b,i} = A_{p,b} = \underset{J \text{in} Q(b)}{\text{average}}(X_{p,b,j}) \tag{4.2}$$

Here $Q(b)$ represents the QC samples in batch b, and average represents the averaging measure, which may be either the mean or the median. As the mean is more sensitive, its use may provide benefits when the number of observations is small, whereas the median offers a more robust measure, useful in cases where experimental outliers may affect the mean.

In [274] the peak is rescaled to the average QC value for the first batch, hence the rescaling factor is $R_p = A_{p,1}$, whilst in [251] it is suggested that the average peak intensity across all samples and batches be used and thus $R_p = A_(p, 1..N_b)$ where N_b is the number of batches. Since changes in instrumental drift can be observed over time, per batch linear regression allows a degree of within-batch dynamics to be accounted for. A linear regression of QCs provides the correction factors:

$$C_{p,b,i} = \beta_b i + \alpha_b \tag{4.3}$$

where α_b and β_b are the regression coefficients for batch b. Here, the integer i, relates to the ith sample for which data were acquired. Other, more advanced regression models including linear smoothers have also been used [274, 278]. Dunn et al. (2011) [273] apply the LOESS (Local Regression) algorithm to generate the trend-line for the QC samples in a method they term QC-RLSC (QC robust LOESS signal correction). LOESS is advantageous in that the data is modelled by a set of local polynomials, which avoids the constraint that the data follow any one global model and is less sensitive to errant data points [279]. The method requires optimisation of a smoothing parameter α. Whilst QCs have been shown to provide an effective method for monitoring and correcting drift there has also been some success involving non-QC correction methods. It has been demonstrated that replicate measurements can be used to track experimental drift in lieu of periodic QC samples in a study involving ICP-OES (Inductively Coupled

Plasma Atomic Emission Spectroscopy) [280]. This naturally allows more time to be dedicated to real sample analysis. The use of QC samples from pooled replicates has also been questioned because of observed inconsistencies between samples and pooled QCs [281].

Checking the performance of any model can however be difficult, and it has been recognised that each dataset should be considered individually in order to determine which methods should be applied [281]. Kirwan et al. (2013) [276] demonstrate success using a variation of the QC-RLSC that substitutes LOESS with a smoothing spline. Here the authors use RSD of technical replicates to determine the algorithm's effectiveness, as did Ranjbar et al. (2012) [281]. Other methods have been proposed which avoid the need for technical replicates. Where QC samples are only used to determine variation, rather than correct for it, the total distance between the QC samples, or the RSD of the QC samples, can be used as a measure of instrumental variation. The distance between QC samples in PCA has been used to justify the idea that instrumental variation is not significant enough to be of concern [282]. The predictive accuracy of PLS-DA on experimental groups has also been utilised to determine the effectiveness of correction [283]. One-way repeated measures ANOVA has been used to calculate unexplained variation to determine the number of peaks for which the variance is reduced on the QCs [281].

Here we explore data that is not amenable to QC correction due to the nature of the drift. The effects and performance of QC and non-QC correction methods are contrasted using these data. Previous studies have focussed on reducing batch or acquisition order differences, using the RSD of replicate samples as a method of gauging correction performance. Since we form the trends used to correct the data from experimental samples in addition to the QC samples, use of this measure could result in real differences between data points being erroneously removed. PLS classification has also been used as a measure of performance, however changes in the data that do not affect the classification rate cannot be detected. Here two evaluation methods are employed, both of which provide a metric of performance on a continuous scale. In addition to the mean relative standard deviation (RSD) to measure the similarity of biological replicates we use PCA-MANOVA, a combination of PCA and Multivariate Analysis of Variance (MANOVA), as a second measure of performance.

PCA-MANOVA allows us to ascertain whether experimental conditions or LC-MS batch order are major sources of variation in our datasets and subsequently whether our improved "background correction" method facilitates a more robust determination of biological trends in our datasets.

4.1.1 Data analysis - Medicago $\mathcal{L}$+ dataset

It can be necessary to discard certain data points, for instance to remove noise peaks which present no useful information. Variables were removed from the dataset where the median of the QC values was zero (i.e. when 50% or more of the QCs fail to show a value) to ensure that an accurate trend could be obtained. Similarly, when determining the trend using non-QC techniques, variables for which the median of all values was zero were removed. All data analyses were carried out in R [284].

4.1.2 Assessment of performance

Performance was assessed using the mean relative standard deviation (RSD) across all metabolites and replicates. For simplicity only replicate sets containing at least 3 observations were used, and values approaching zero (identified by at least one of the 3 or more values being zero in the original data, or containing all zeroes in the corrected data) were discounted. RSDs were calculated using the equation for the RSD of a subset [285]:

$$RSD = \frac{\sigma}{\bar{\bar{x}}} \tag{4.4}$$

where σ is the standard deviation of the 3 replicates and $\bar{\bar{x}}$ is the grand mean for the metabolite. Our RSDs were calculated from the sets of biological replicates from plants exposed to the same experimental conditions for the same timepoints. It should be noted that in comparison to technical replicates, some differences are still to be expected even if a perfect batch correction were to be performed due to natural biological variation between the samples.

A combination of PCA and MANOVA was also used to judge the correction in terms of group separation. Data were mean centred and variables scaled to unit variance (divided by the standard deviation of the variable) prior to PCA to prevent metabolites with larger intensities dominating

the scores. MANOVA was used to provide an F statistic which shows the between group to within group variance ratio:

$$F = \frac{\text{variance between groups}}{\text{variance within groups}} \tag{4.5}$$

Comparison of the F value with the appropriate F distribution gives a p-value for the significance of any difference between experimental groups. We used MANOVA on the PCA scores (coordinates of the rotated variables) for the first two principal components to quantify differences between experimental groups. This allowed the most apparent variations in the data to be considered in the MANOVA test. With an ideal correction the highest source of variation should be due to experimental groups rather than batch differences. The groups considered in each test set are:

- control and drought groups
- drought and dual-stress groups
- grouping due to LC-MS batch

We compared the control and drought groups as differences were already apparent in the uncorrected data and these should be retained by any correction method applied. Initial analysis showed little difference between the drought and dual stress groups and a correction method that could reveal these differences would be advantageous.

4.1.3 Correction Methods

The correction procedure involved the determination of the correction factors $C_{p,b,i}$ shown in Equation 1. This process was split into three stages. In the first stage the observations used to calculate the trend were selected: this could be based solely on the QCs, sets of replicates, or on all observations. The second stage involved selecting the method to be used to calculate the trend and in the third stage the observations to which the correction was applied are selected, i.e. individual batches or the full dataset.

In this analysis, correction methods were tested using only the QCs, but also using all observations (including QCs) to generate the trend, which we refer to as background correction. Both methods were tested on batches individually (batch-wise), and with the full dataset considered as one.

4.1.4 Trend Functions

The different methods used to determine the trend in the second stage were as follows:

Mean – The trend is set to the average of the samples, as in Equation 2.

Linear Regression – The trend is modelled via a linear regression of the samples.

Moving Median – The trend is generated from the data using a simple moving average for smoothing. We used the median as analysis revealed the moving mean resulted in unfavourable responses to individual high or low values (including genuine experimental values and not just outliers). For the moving median the correction factor C_i is calculated as the median of a moving window:

$$C_{p,b,i} = \text{median}(X_{p,b,i-w}..Xp, b, i + w) \tag{4.6}$$

where the $X_{p,b,i}$ values used in the calculation are as defined for equation 1 and w is the window width.

Polynomial regression – Polynomial regression allows the data to be modelled as a simple nth degree polynomial and requires the degree of the polynomial n to be specified.

Smoothing Spline – The smoothing spline method fits a set of intersecting polynomials to the data. The function is controlled by a smoothing parameter λ, with larger values of λ leading to smoother functions [286]. The `smooth.spline` algorithm from the R package `stats` [287] was used to generate the smoothed spline.

LOESS – Combines multiple regression models and has previously been used to determine the correction factors both on QCs and on the full data set for DI-MS and LC-MS data [276, 288]. Like the smoothing spline, LOESS is also controlled by a smoothing parameter.

4.1.5 Method parameters

Several methods used to account for non-linear drift require parameters to be optimised. The window width w for the moving median, the degree n

Method	Parameter	Value
LOESS	Neighbourhood (α)	0.45
Batchwise LOESS	Neighbourhood (α)	0.5
Moving median	Window width (w)	5
Batchwise moving median	Window width (w)	5
Polynomial	Degree (n)	6
Batchwise polynomial	Degree (n)	1

Tab. 4.1: Table showing correction method parameter values optimised in terms of RSD of biological replicates

of the polynomial and the neighbourhood α that determines the smoothing parameter in LOESS were optimised to give the lowest mean RSD for biological replicates. The optimised parameters are listed in Table 4.1. Note that the correction using the batch-wise polynomial performed best with a polynomial degree of 1, effectively making it a linear correction. The smoothing spline was calculated using the R function smooth.spline with the default parameter set, which optimises the parameter λ via generalised cross validation in order to best fit the curve to the data [287].

4.2 Results and discussion – Medicago datasets

For each of the *Medicago* datasets, it is clear from the Principal Components Analysis (PCA) of the scaled data that the majority of the variance is due to batch differences rather than experimental groups. Figure 4.1a shows the scores plot for the first two principal components for the $\mathcal{L}+$ dataset. After batch correction using the traditional "mean of the QCs" method, PCA plots reveal that batch differences in the $\mathcal{L}–$, datasets are clearly reduced, with differences between the experimental groups becoming more apparent. However, this method was not able to correct for the batch differences in the L+ dataset as shown in Figure 4.1b. It can be seen that several of the batches are "split" along the first principal component (PC1), with part of the batch having low scores for PC1 and the rest having higher scores. One of the implications of this is that the assumptions of standard statistical tests, such as t-tests or ANOVA may be invalid. Closer inspection of the L+ dataset reveals that a large degree of within-batch drift can be observed for many metabolites, such as the example shown in Figure 4.2a. Initial analyses of correction methods were also confounded by the presence of an

outlier (drought, day 6, replicate 3), which was removed and the analysis repeated.

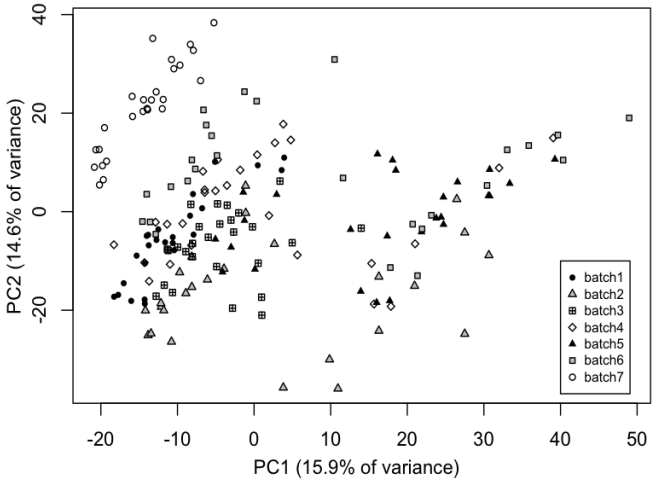
The use of linear regression modelling of the QCs in each batch to determine the trend appears to give improved results, as batch differences are no longer the greatest source of variance in the PCA. However batch differences are not eliminated and are now apparent along PC3. Furthermore, the method creates a number of outliers due to intensities being divided by very small numbers. This happens, for example, with metabolite #1283, which is responsible for the majority of variance along PC2 in unscaled PCA, and so is not restricted to peaks of low intensity. Patterns in the data when viewed in order of acquisition also remain, with sudden changes in the reported intensities within an individual batch that are not accounted for by a linear model. For example in batch 6, metabolite #1459 shows a drift in the experimental values different to that of the QCs (Figure 4.2a). Such changes, which could have instrumental or analytical origins, lead to a poor fit of the linear regression model. The average RSD of the biological replicates, calculated across all variables and metabolites, shows that linear regression of the QCs leads to a huge increase in variation (Figure 4.3). In fact the greatest source of variance seen in PCA is now due to artefacts introduced by the QC correction rather than to genuine differences between experimental groups.

Figure 4.3 shows that methods which use all observations reduce the batch variation more than methods based on the QCs alone. The comparatively poor performance of the QC based methods may be due to several factors:

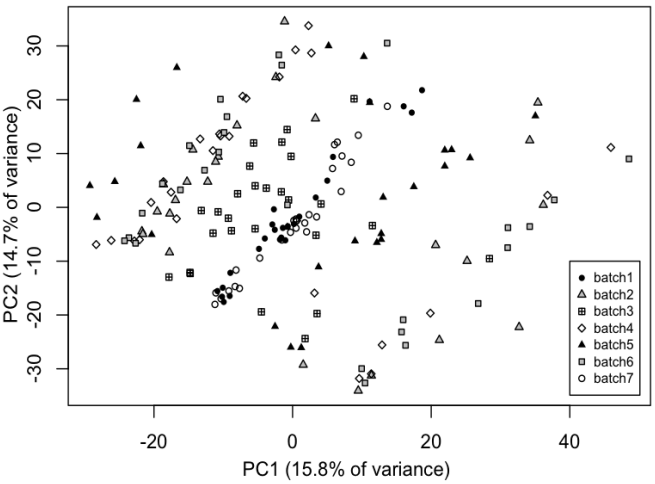
- It can be problematic to determine an accurate trend due to the variation in the recorded intensities of the QCs.
- Since the QCs are placed intermittently they are unable to account for changes occurring at points between their placement
- The number of QCs is low in comparison to the total number of observations, providing less information from which an accurate set of correction

factors may be determined.

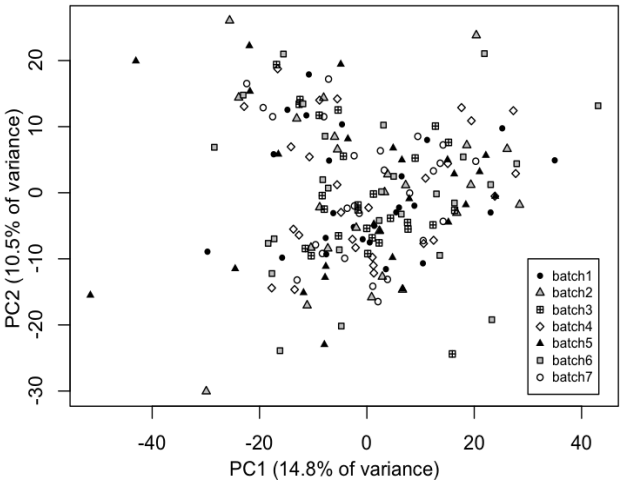
Background correction methods, i.e. techniques based on all observations (not just QCs), can follow the drift seen in the actual experimental samples



(a) Original

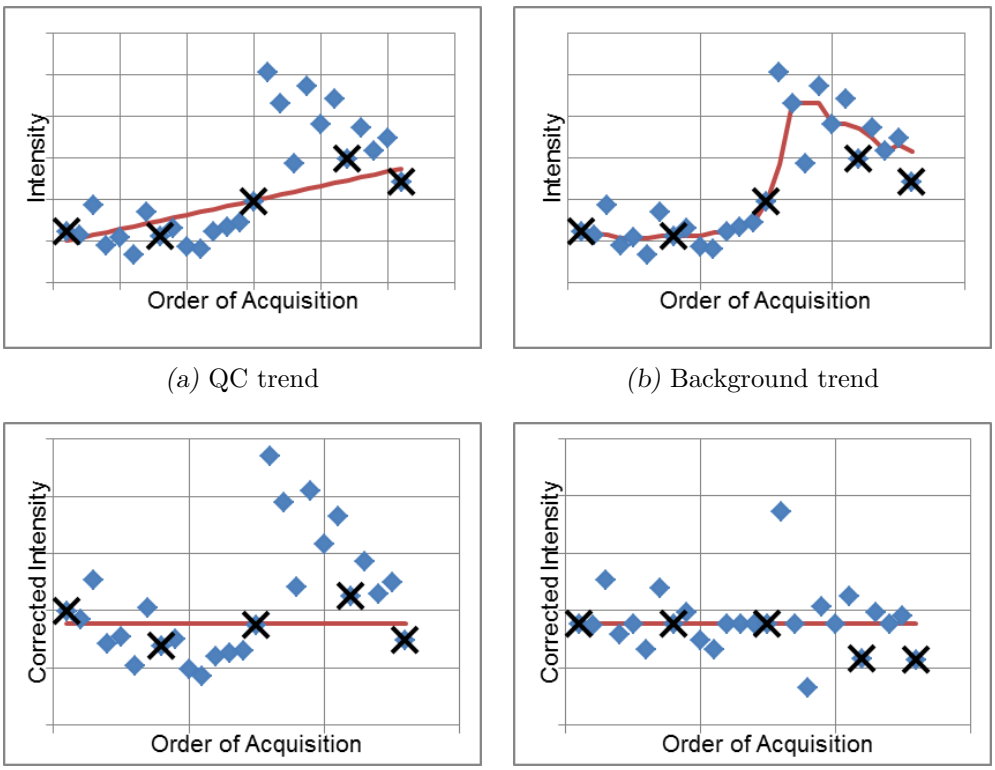


(b) QC-mean corrected



(c) Background corrected

Fig. 4.1: (a) The scores plot for the first two principal components of the scaled $\mathcal{L}+$ dataset showing batch differences as a major source of variation. (b) The scores plot after batch correction using the mean QC value, in which batch differences are made worse. (c) The scores plot after batch correction using the background correction method, in which batch differences are no longer apparent.



(a) QC trend

(b) Background trend

(c) QC corrected

(d) Background corrected

Fig. 4.2: Plots showing how two methods of correction affect a metabolite ($\mathcal{L}$+ #3280) and batch showing strong within-batch drift. Plots (a) and (b) show the values prior to correction, with the trend used for the two different correction methods shown by the bold line. Figures (c) and (d) show the values post-correction, with the bold trend-line at 1.0. The linear correction (c) shows a notable pattern in the results when compared with the moving median correction in (d).
Diamonds indicate observations, with QCs highlighted by crosses. The line indicates the correction factors forming the trend on which the corrections are based.

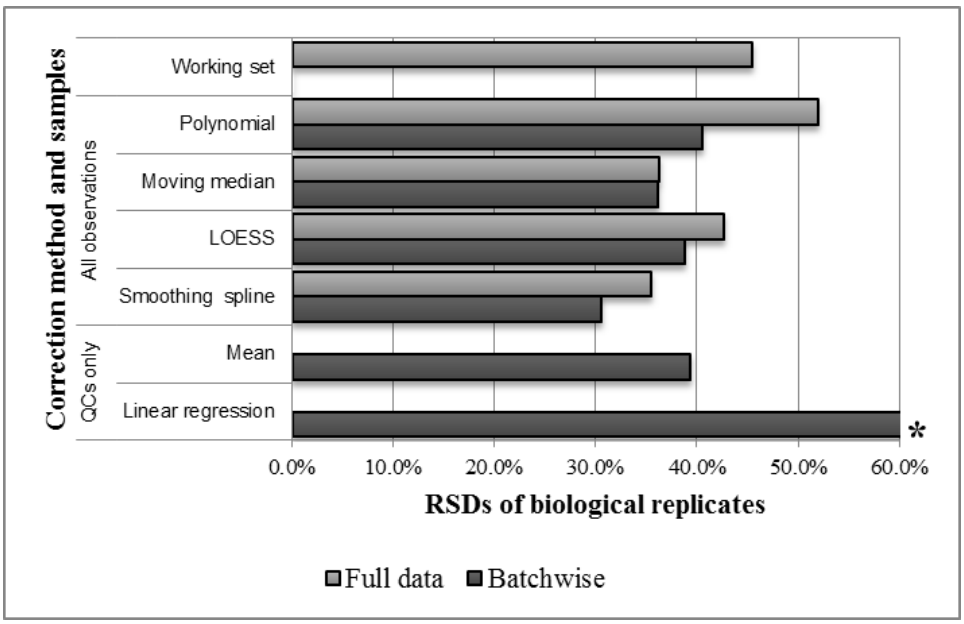


Fig. 4.3: The Mean Relative Standard Deviation (RSD) using various correction methods. Lower values are indicative of greater coherence between sample replicates and suggest improved correction. The working set represents the original data with an outlier observation and metabolites approaching the limit of detection removed, as described in the methods section. For each method the results are shown for the optimised parameters. *Note that the results using the linear regression of the QCs have been truncated and the RSD is actually 193%. Calculation of QC-only based techniques using the full dataset is not appropriate and is not shown. The working set is not corrected and hence only one value is displayed in the graph.

of interest, allowing the correction of metabolites where the concentration is sufficiently different between QC and experimental samples. Figure 4.3 also shows that performing a background correction separately on each batch is more effective than ignoring batching and using all observations in a single background correction step. The average reduction in RSD achieved using batch-wise correction is 5.4%. The difference is most apparent in polynomial correction, with the moving median being the least affected, possibly due to the moving median's ability to rapidly track abrupt changes in the general flow of the data.

The best results, in terms of RSD between replicates, is achieved with the batch-wise smoothing spline with a 14.4% reduction in RSD in comparison to the working set (the original data with variables classified as "noise" removed). The LOESS and the moving median correction methods both gave an improvement of $\tilde{9}$% in comparison with the original data.

The optimal parameters determined by RSD analysis are shown in Table 4.1. The correction methods were then evaluated using PCA-MANOVA. Figure 4.4 shows the PCA-MANOVA F statistics for control-drought discrimination are actually decreased by some batch correction methods in comparison to uncorrected data. In particular, the moving median, which gave good results in terms of RSD between replicates, gives a lower F statistic for the between group to within group variance ratio than for the working set. However the control-drought groups separate well prior to batch correction, with a p-value of 0.001 for the F-test. The p value of 0.003 for the moving median shows the separation is still significant. The smoothing spline methods, which also showed good separation based on RSDs, show little difference in comparison to the uncorrected data, suggesting that, at the very least, we can apply these corrections without significantly damaging existing variations of interest.

Figure 4.5 shows the PCA-MANOVA results for the drought and dual-stress groups. It can be seen that all correction methods give improved separation of experimental groups in comparison to uncorrected data. Interestingly, the moving median methods provide the best separation, performing considerably better than the smoothing spline methods. Figure 4.6 shows PCA scores plots before and after correction with the moving median. The correction highlights a trend with plant age across PC1 with the older plants showing increased separation with experimental group along PC2.

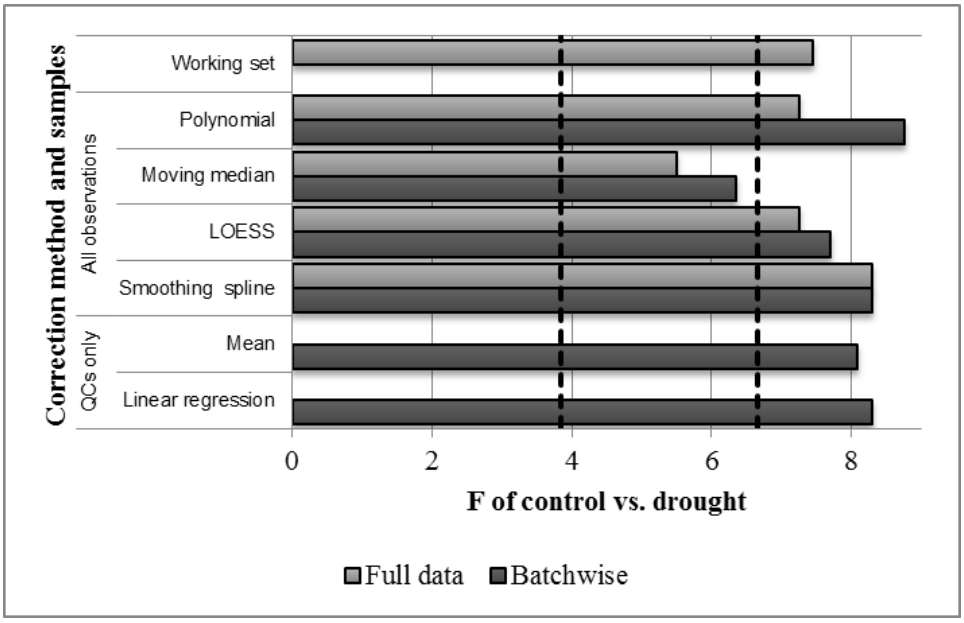


Fig. 4.4: PCA-MANOVA results for the separation of control and drought experimental groups after batch correction using various techniques. A larger F statistic indicates a higher between-group to within-group variance ratio and therefore improved correction. Where applicable the techniques have been optimised to provide the lowest RSD across biological replicates. The working set represents the original data with metabolites approaching the limit of detection removed. The dotted lines show the critical F values of 3.85 for $p = 0.05$ and 6.65 for $p = 0.01$.

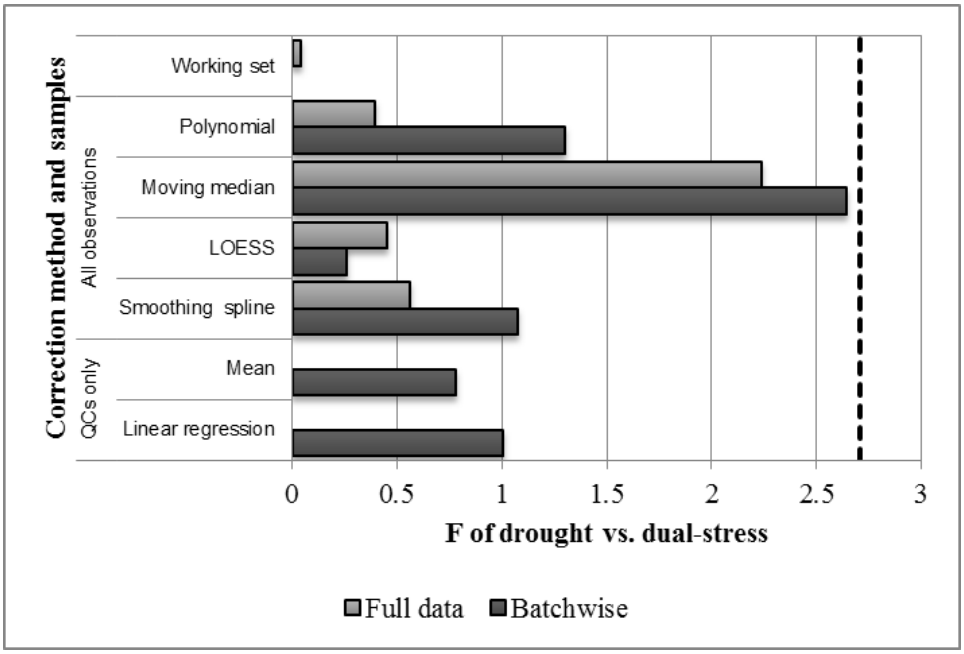


Fig. 4.5: PCA-MANOVA results for the separation of drought and dual-stress experimental groups after batch correction using various techniques. A larger F statistic indicates a higher between-group to within-group variance ratio. Where applicable the techniques have been optimised to provide the lowest RSD across biological replicates. The working set represents the original data with metabolites approaching the limit of detection removed. The dotted line shows the critical F-value of 2.71 for $p = 0.1$.

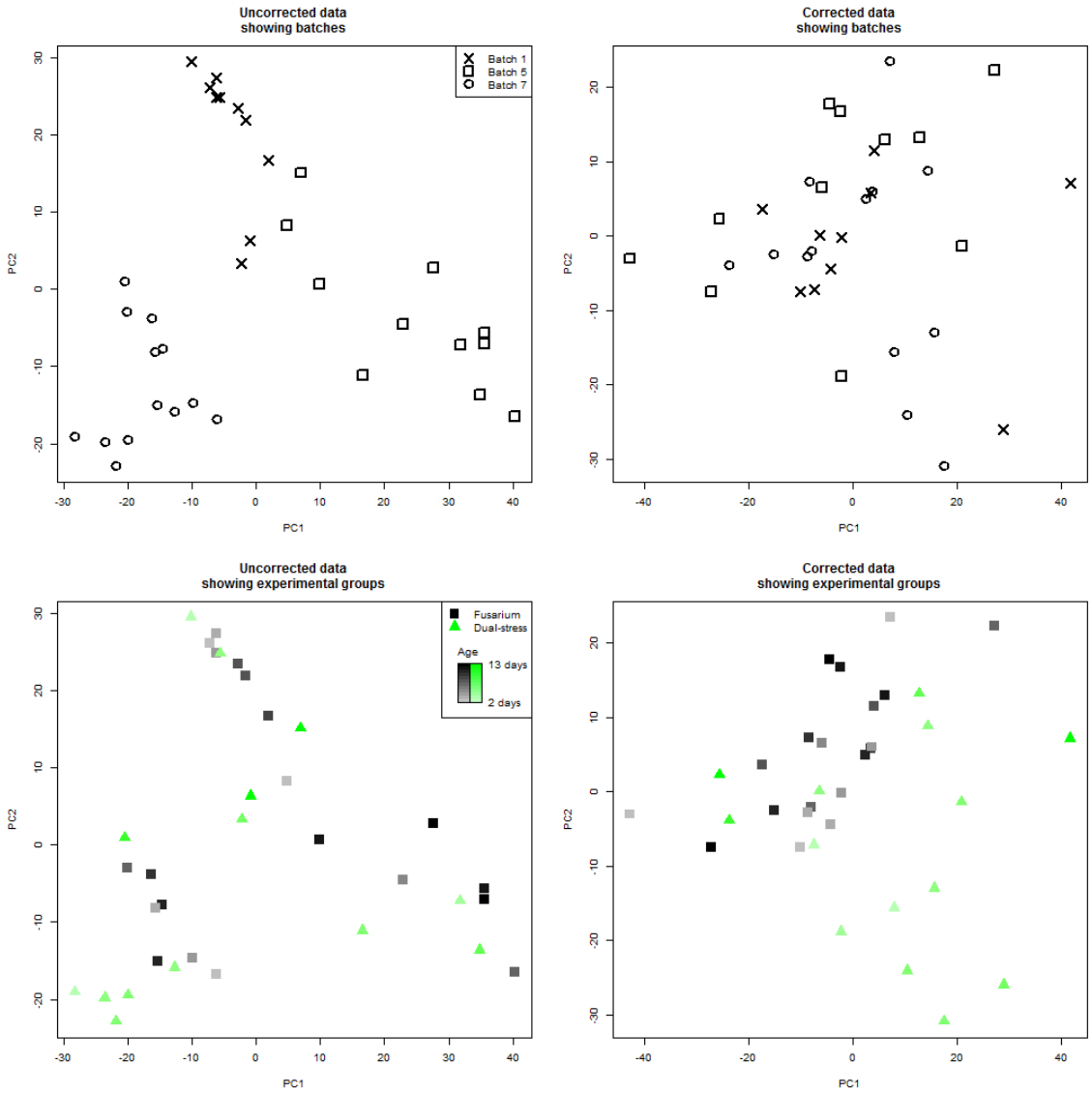


Fig. 4.6: PCA scores plots of *Fusarium* and dual-stress samples for three batches, before and after background correction. The top plots show that obvious batch differences in uncorrected data are not evident after correction. The lower plots show the same data coloured according to experimental group with darker colours indicating samples from later in the time series. For the uncorrected data, PC1 accounts for 23.0% of the variance, and PC2 for 15.7%, whilst, for the corrected data, PC1 accounts for 10.2% and PC2 for 17.7%.

PCA-MANOVA analysis of batch separation shows all correction methods provide a drastic reduction in batch differences, with only the uncorrected data having a significant F statistic. However, in some cases the F statistic may be reduced by the splitting of batches into two clusters, as shown in the PCA scores plot in Figure 4.1. Since the different metrics of success yield different results this suggests that different correction techniques have their own merits and some may be more suited to certain situations than others.

4.3 Concluding remarks

Where experimental drift occurs steadily throughout data collection, the overall trend may be identified using QC samples. However, jumps between batches require each batch to be treated individually and may result in insufficient QC samples to characterize the within-batch drift. In such cases improved correction may be achieved using a smoothed function of all observations within the batch to represent the trend. Background correction can be more effective than standard QC correction and does not necessarily require additional samples. Although the use of a batch-wise smoothing spline to represent the experimental drift was found to reduce the differences between biological replicates, all background correction methods evaluated provided better discrimination between experimental groups than uncorrected data. The use of a simple moving average not only gave good reduction in RSDs between replicates, but gave the highest between-group to within-group variance ratio for the drought and duel-stress groups, so that more complex smoothing methods may not be necessary. However, the moving median was less effective for the drought and control groups, where separation was already apparent in the uncorrected data. Just as scaling improves results in some situations and not others, different correction techniques may be more suited to some situations than others with no single method providing the optimal correction in all cases.

5. CLUSTERING METABOLOMIC TIME-SERIES

5.1 Introduction

The specific roles of individual genes or metabolites are sometimes represented within gene regulatory networks (GRNs) for genes or biochemical pathways for metabolites. The relationships can be expressed as a graph, with the nodes representing genes or metabolites and the edges representing the interactions between them. Such networks were historically calculated manually. For instance, M. Calvin mapped out the set of interactions now known as the Calvin-cycle with the aid of paper chromatography in 1956 [289]. Whilst manual interpretation is feasible for studies involving small numbers of metabolites, the large amount of data pertaining to modern, non-targeted studies necessitates relationships to be identified computationally.

The guilt-by-association (GBA) axiom is frequently noted in -omic analyses [290, 291]. The principle implies that elements, such as genes or metabolites, performing similar functions or sharing the same mediators will naturally exhibit similar behavioural patterns. In its simplest realisation, correlation between metabolic concentrations or gene expressions can be used to infer interaction. Although it has been noted that many elements in a network do not in fact show similar behaviour [292], and that simple correlation may produce false-negatives, or "missed" interactions [293] GBA has nonetheless been demonstrated to extract valuable information from data. In the context of *Medicago*, a study of HPLC-UV data, using a set of labelled compounds used correlation analysis to determine the effect of yeast-based elicitors in alfalfa [294]. A similar but non-targeted study of GC- and LC-MS time-course data for 249 compounds also used correlation analysis to study biotic and abiotic stress responses in *Medicago truncatula* [293]. Compounds in this case were identified via comparison to known samples and existing databases. Both studies empirically demonstrated metabolic interactions

present only in the presence of chemical or environmental elicitors, allowing context-specific pathways to be determined.

5.1.1 Cluster analysis

The *Medicago* dataset introduced in Section 3.1 contains 2920 potential metabolites (peaks). A plot of the network of correlations between peaks, overlaid on top of the PCA scores, is shown in Figure 5.1. This plot was generated from the PCC matrices between each peak and every other. Four matrices were generated in total, each using only the observations from one experimental group at a time – $\mathcal{C}$, $\mathcal{D}$, $\mathcal{F}$ and $\mathcal{B}$. The correlations plotted in the figure are highlighted by the experimental group(s) in which they occur.

The correlation of 1000s of compounds can become unwieldy and will inevitably suffer from the multiple comparisons problem. However, from the plot, several highly inter connected groups of compounds can be seen. Many of these compounds, as noted in the *Medicago* studies noted above, show correlations that are only elicited under certain experimental conditions. This natural grouping sets the precedent for cluster analysis. As a complementary method to the generation of correlation based networks, cluster analysis can be used to analyse metabolomic data. This is a primarily unsupervised approach and so avoids the so-called fishing expedition paradigm of multiple univariate analyses. Clustering is used to divide data into groups or *clusters* in which the association between members of the same cluster is strong and the association between members of different clusters is weak. This serves a variety of purposes. Data dimensionality is reduced, and so assists other analyses either by reducing the search space to a discrete set of clusters, or by acting as a method of data exploration in which the data is effectively summarised to a set of common traits. Furthermore, for -omic datasets, clustering can be used to highlight common relationships or potential functionality between variables (metabolites or genes).

This chapter demonstrates the use of clustering on the *Medicago* dataset, which was previously described in Chapter 3. In the context of this dataset, clustering will be performed with an aim to identify sets of metabolites showing stress responsive profiles. In addition to the identification of potential biomarkers, a secondary objective is to test the hypothesis that a data-driven comparison of the clusters, to databases of known metabolic pathways, can identify specific pathways affected by the stress conditions.

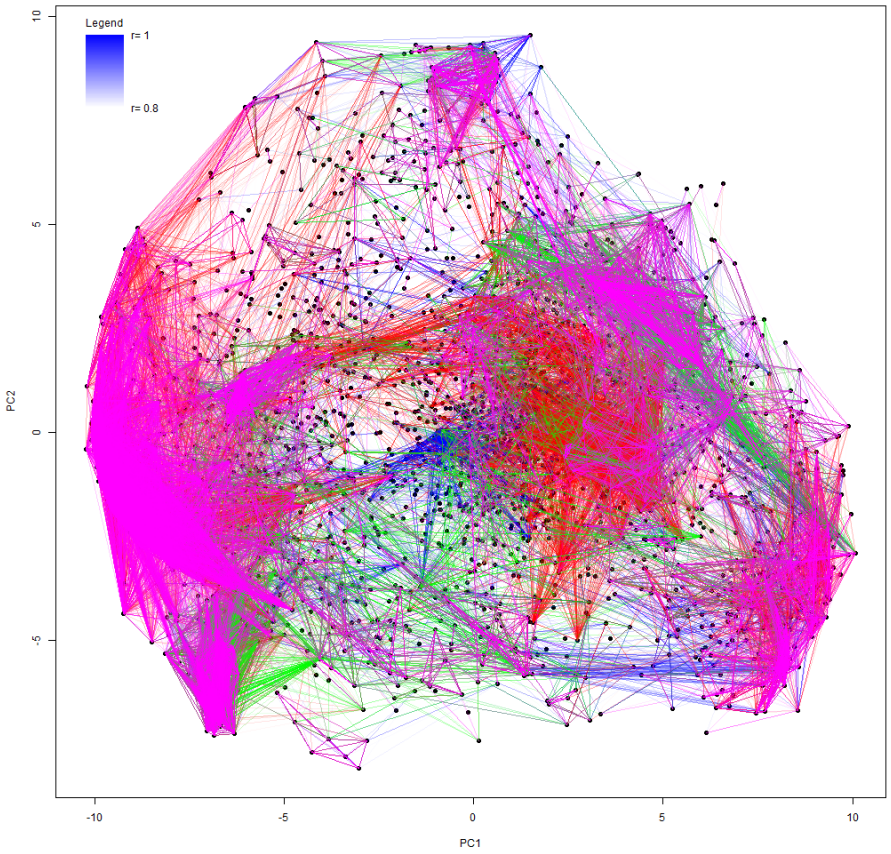


Fig. 5.1: A plot showing strong ($r > 0.95$) correlations between peaks. Blue lines indicate strong correlation in between $\mathcal{C}$ samples, red between $\mathcal{D}$ samples, green between $\mathcal{F}$ samples, and magenta between $\mathcal{B}$ samples. There are noticeable regions of PCA space showing tight correlation of samples for different experimental groups.

For this analysis, it is assumed that that any data submitted this workflow has been suitably pre-processed. In the case of LC-MS data this necessitates that feature selection and batch correction, previously discussed in Chapter 4, have been performed. However, during the course of the clustering analysis a number of other processing stages, prior to the actual clustering algorithm, were found to greatly assist in improving the performance of time-series clustering, which are summarised in the following workflow:

- Formation of the clustering vectors

  Control correction

  Trend-line generation

- Removal of interfering data
- Selection of a distance metric
- Selection of a clustering algorithm
- Selection or optimisation of algorithm parameters

5.2 Methods

5.2.1 Pre-processing

Effective feature selection, noise filtration, batch correction and data-scaling are standard practices in non-targeted metabolomic studies. The exact implementation of data-scaling is often analysis dependent. Clustering methods such as HCA and k-means can be particularly sensitive to scaling. Differences between large variables will contribute much more to distance metrics such as the Euclidean distance and can obscure differences in smaller variables. It has been shown on NMR spectra that, without scaling, spectra may simply cluster according to sample concentration rather than biology [295]. However, other distance metrics, such as the Pearson distance[1] are insensitive to data scale.

For this chapter, the batch-corrected *Medicago* from Chapter 4 was used as the input to this analysis. This data was scaled to unit variance and mean centred (autoscaled).

[1] $1 - r$, where r is the PCC

5.2.2 Input vector selection

Clustering algorithms use a set of n_C input vectors $(X_1, X_2, ..., X_{n_C})$. Either these vectors are directly used by the clustering algorithm (e.g. k-means), or the algorithm uses a distance matrix calculated from these (e.g. k-medians and HCA). The purpose of the clustering algorithm is to assign each vector to a set of one or more clusters (C):

$$X_i \in C \tag{5.1}$$

In the usual case, each input vector represents an experimental observation, and the elements of each individual vector are defined as the set of independent variables for that observation (j), hence:

$$X_j = \{v_{j,1}, v_{j,2}, ..., v_{j,n_P}\} \tag{5.2}$$

Where $v_{j,i}$ is the ith variable of the jth observation and there are n_P variables and n_O observations. The number of cluster vectors is therefore equal to the number of observations, $n_C = n_O$.

In order to further the understanding of the metabolites, the data is transposed such that clustering is performed on the independent variables – the metabolites – rather than the observations. Hence, given the ith metabolite, the cluster vector X_i represents the set of experimental observations made on that metabolite:

$$X_i = \{v_{1,i}, v_{2,i}, ..., v_{n_O,i}\} \tag{5.3}$$

Here the number of cluster vectors is then equal to the number of metabolites and $n_C = n_P$.

5.2.3 Trend identification

For the purposes of clustering intensity data from an experiment containing n_G experimental groups, the observations from each experimental group can be concatenated to form the input vectors as shown in the example in Figure 5.2. The exact order of observations depends on the distance metric but in most cases, such as with the Euclidean or Pearson metrics, the order is unimportant, provided that it is the same for each input vector. In the layout of the example vectors, and also in the *Medicago* dataset, each

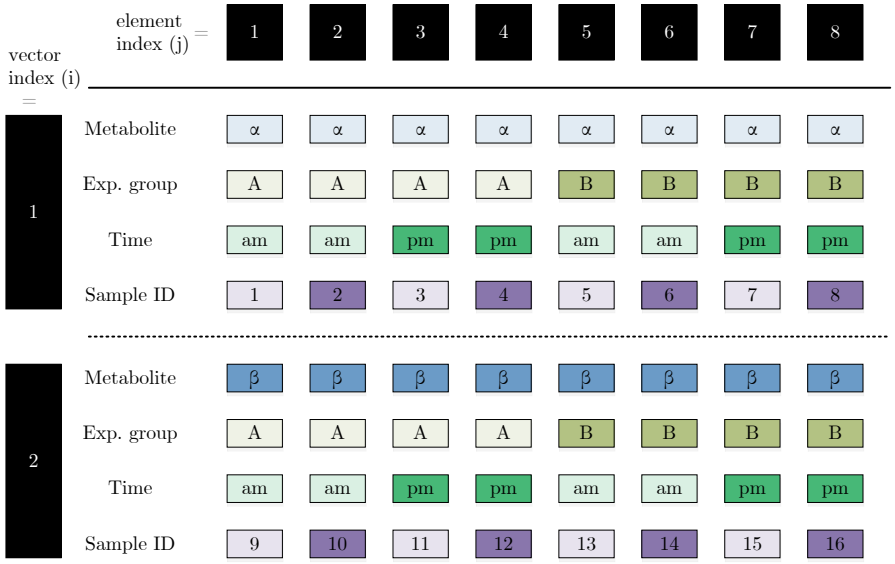


Fig. 5.2: Two sample clustering vectors ($j = 1$ and $j = 2$) from a hypothetical dataset containing two metabolites (α and β), two experimental groups (A and B), two timepoints (am and pm) and two replicates per time-point and experimental group. All samples are identified by a unique ID.

observation comprises two or more biological replicates. The replicates are biologically independent, and each replicate is from a different plant. Since these replicates are independent, the comparison between samples is not clear. In the example shown, the order of the replicates is ambiguous and a change to the positions of any two replicates, for instance by swapping samples #11 and #12 in the example figure, would be make no semantic change. However, different results would be obtained.

A second issue, highlighted in Section 3.1, is that the analysis of the *Medicago* dataset is confounded by the presence of noise in the data. Post batch correction it is expected that this will be true for any dataset, as some biological or technical noise will inevitably remain elusive to known correction methods. Sample replicates, either biological (e.g. different plants) or technical (e.g. multiple analysis of the same samples) serve as a mechanism by which noise can be both measured and reduced. By combining sample replicates, later analyses are also simplified by removing the aforementioned ambiguity in the order of the input vectors.

The simplest method of accounting for replicates is through simple averaging:

$$\tau_{p,g,t} = \underset{r=1..n_R}{\text{average}}(s_{p,g,t,r}) \tag{5.4}$$

Where $s_{p,g,t,r}$ represents the source data (peak p, group g, time t and replicate r) and $\tau_{p,g,t}$ represents the final "trend", with replicates accounted for. The number of replicates is denoted by n_R.

If outlier values are present in the data, the median may provide a more robust measure of central tendency in comparison to the mean.

However, it is possible to apply a more advanced smoothing method to account for, whilst still reducing, noise. Here a trend profile is generated to obtain the set of smoothed points. An example trend is extending the average to a *moving average*. For each time point in the trend $\tau_{p,g,t}$, the moving average considers all replicates within a pre-specified time window:

$$\tau_{p,g,t} = \underset{r=1...n_R, k=(t-\frac{w-1}{2})...t+\frac{w-1}{2})}{\text{average}} s_{p,g,k,r} \tag{5.5}$$

The window width parameter, w, must be optimised to provide a smooth trend without significantly compromising resolution. Other smoothing func-

tions include modelling the data for each p and g and with a smoothing function $f_{p,g}$, and then obtaining the result of this function for each individual time point.

$$\tau_{p,g,t} = f_{p,g}(t) \tag{5.6}$$

Example smoothing functions include polynomials, smoothing splines [286] and LOESS [279].

From this, it is apparent that the trend profile elements $\tau p, g, t$ can be considered analogous to the batch correction factor, $C_{p,b,i}$ previously discussed in Chapter 4. In this case, observation time (t) substitutes acquisition order (i) and experimental group (g) substitutes batch number (b).

5.2.4 One vector per experimental group

While the input vectors can be sourced from the concatenated observations of each experimental group, it is also possible to consider the observations of each experimental individually. This focusses the study on general metabolite trends as opposed to specific response profiles and has the potential to be useful if the variety of response different profiles are too numerous to cluster individually. In this method a clustering vector is created for each experimental group, providing a n_G different input vectors to the clustering algorithm per peak. The number of input vectors is therefore $n_C = n_G \times n_P$.

5.2.5 Control correction

Changes in metabolite levels over time may not necessarily relate to the experimental conditions of interest. When dealing with plant material over a period of days it can be expected that a number of compounds will be growth related and will show predictable trends with age. Such compounds are clearly present in the *Medicago* dataset, such as the peak shown in Figure 5.3. Other trends also present include fluctuations with environmental conditions such as light or temperature. In the presence of a control group corrections can be applied to account for these changes, providing a set of control-relative changes. To correct the data in this fashion the trend of the control group can be subtracted from the experimental groups:

$$s'_{p,g,t,r} = s_{p,g,t,r} - \tau_{p,c,t} \tag{5.7}$$

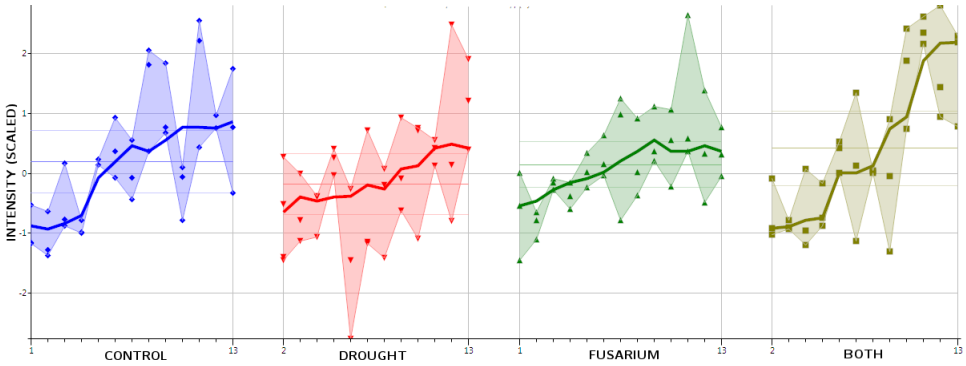


Fig. 5.3: Plot showing $\mathcal{L}+$ peak #984 (m/z = 217.068, retention time = 2.09 min, tentative identification = D-ononitol). Whilst showing a strong upwards trend with time for the experimental groups ($\mathcal{D}$, $\mathcal{F}$ and $\mathcal{B}$) the control group ($\mathcal{C}$) also shows an age related trend making the other trends hard to interpret.

Here $s_{p,g,t,r}$ represents the intensity prior to control correction and $s'_{p,g,t,r}$ represents the same value post correction. $\tau_{p,c,t}$ represents the trend of the control group for the corresponding time point, t.

An advantage of control correction is in that it also allows truncation of the input vectors via exclusion of control group samples, simplifying the search space by reducing the number of possible response profiles.

5.2.6 *Distance measures*

In order to cluster data, a measure of similarity or dissimilarity between data points is required. Euclidean distance and Pearson distance are popular choices in relation to clustering time-course or expression profiles [296]. The Pearson distance between two vectors x and y is defined as $1 - r$, where r is the PCC of x and y.

Qian et al. however note in [296] that taking a direct representation of relationship between gene or metabolite levels could be considered an oversimplification of real biological processes, since other factors are present which are indicative of a relationship between trends over time. In the original paper the authors deal with gene expression and note that, in addition to directly correlated profiles, there are cases of suppression, where one gene inhibits the expression of another, as well as temporal delays in effects, since time is required for transcription to take place. It is a reasonable hypothesis

that these same assumptions hold true for metabolomic time series. Conversion processes will increase the level of one metabolite at the cost of another, and time is required for substrates to be transported and react. A distance measure termed *local clustering* has been proposed to deal with these issues, and accounts for these additional relationships [296]. The process is described below, where a "variable" represents a gene or metabolite:

1. Normalise all variables (UV scale and centre)
2. Construct a "score matrix" M for each pair of variables, x and y

   $M_{i,j} = x_i y_j$

   Where x_i and x_j are the values at time-steps i and j for each variable respectively.
3. Construct sum matrixes E and D:

   $E_{i,j} = max(E_{i-1,j-1} + M_{i,j}, 0)$

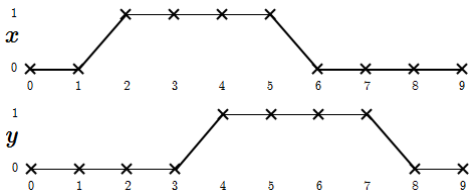
   $D_{i,j} = max(E_{i-1,j-1} - M_{i,j}, 0)$
4. Find the match score, s

   This is the maximal value from the matrixes E and D.

In addition to the match score s, this method allows determination of the kind of relationship present: match scores originating from matrix D are indicative of inverted (inhibitive) relationships, and off-diagonal match scores indicate time-delayed relationships. Figure 5.4 presents an example calculation and visualises how the process is able to account for time delayed relationships.

The algorithm has been shown to be able to determine new relationships between genes when tested using a simple clustering method on expression profiles of the yeast cell cycle [296]. The proposed network topology was in this case validated through a qualitative comparison to known relationships in the literature, as well as the viability of newly discovered relationships.

The Qian metric does make the assumption that the time-course data occurs in meaningful steps. As the frequency of measurement is increased the delay between events will become more apparent and hence, events may not be visible if the time-steps are prohibitively large. This is potentially particularly true for metabolomic studies, where time-scales may be on the scale of



x	0	0	1	1	1	1	0	0	0	0
y	0	0	0	0	1	1	1	1	0	0

(a) Gene expression profiles x and y

0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0
0	1	1	1	1	0	0	0	0	0
0	1	1	1	1	0	0	0	0	0
0	1	1	1	1	0	0	0	0	0
0	1	1	1	1	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0

$M_{i,j} = x_i y_y$

(b) Score matrix M

0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0
0	1	1	1	1	0	0	0	0	0
0	1	2	2	2	1	0	0	0	0
0	1	2	3	3	2	1	0	0	0
0	1	2	3	4	3	2	1	0	0
0	0	1	2	3	4	3	2	1	0
0	0	0	1	2	3	4	3	2	1
0	0	0	0	1	2	3	**4**	3	2

$E_{i,j} = max(M_{i,j} E_{i-1,j-1}, 0)$

(c) Sum matrix E

Fig. 5.4: Calculation of the match score $s = 4$ using the local clustering method for two hypothetical variable profiles. Here it can be seen that the time delay is accounted for by the procedure.

2 minutes or less, in comparison to the time-scale of transcriptional events, with the lower end of the time scales being around 10 minutes [297, 298]. A second potential issue is of genes or metabolites operating on different scales, for instance whereby a sharp increase in one metabolite results in a gradual increase or sustained generation of another due to the creation of a non-consumed catalyst. Some problems with false positives are however already evident, and it is likely that the more factors which are considered the more the false positive rate will grow. Later research comparing true to false positive rates at various s threshold settings for known genes in yeast and *Arabidopsis* does suggest however that whilst the method identifies more interactions, the false positive rates are nonetheless better than direct similarity measures such as the PCC [299].

Like Qian et al., Kiddle et al. notes that this process requires high quality data and high temporal resolution. They also note however, that more complex similarity measures are not actually suitable for use with simple clustering methods such as k-means, due to a mean not being calculable from the results. On the other hand more complex methods such as Hidden Markov Models are computationally intensive and therefore unsuitable for large datasets. They thus suggest an approach which combines the *local-clustering* scoring method from Qian et al. with the *affinity propagation* clustering method devised by Frey and Dueck, which will be discussed later.

This method was found to produce results with a significantly lower cost-function when compared with k-centres [299]. In addition to indicating coherence with genes identified in existing studies this combined method indicated new potential links. One interesting discovery from this is of several genes related to the circadian clock, which, from the time-course data is highlighted by a particular 24 hour time cycle. Unfortunately this research, like many others, suffers from a lack of real-world knowledge to compare the results of their analysis to, making a true comparison of methods difficult.

5.2.7 Peak filtering

Since the clustering algorithm deals with only the trend, post-scaling, a flat trend will in isolation be seen as an erratic line. These trends have the potential to negatively impact the ability of the clustering algorithm to identify genuine clusters. Early identification of such trends and exclusion from from the clustering algorithm is a potential method of avoiding this

issue.

A potential method to identify these trends is through a univariate t-test, which requires the presence of control-corrected data. Since, post-correction, the control group can be used as a example of a "flat" profile, the comparing the data from each of the experimental groups can be compared in turn against the control group in order to identify any changes of interest:

- The p value of a t-test comparing all data (time-points and replicates) for each experimental condition in turn, against the same data for the control group.
- Using the most significant (lowest p) of these values as the final *significance* value p_{min}.
- If $p > \alpha$, where α is the chosen confidence limit then the peak is marked as "insignificant" and is excluded in the clustering algorithm.

5.2.8 Clustering methods

5.2.8.1 Hierarchical cluster analysis

For n observations, hierarchical cluster analysis (HCA) begins with n clusters, each consisting of a single observation. Clusters are merged sequentially according to a distance metric until a single cluster of all observations is obtained. A dendrogram (tree diagram) obtained from the distances at which the mergers take place allows the relationship between observations to be visualized. HCA has been used in various metabolomic studies, for example, to investigate similarities in the effects of different toxins on rats [109]. Here, HCA not only identified organ-toxicity-specific differences but also showed similarities between treatment groups that helped explain misclassifications in kNN analysis. Hierarchical clustering has been used to characterize ^{1}H NMR spectra obtained from urine specimens from population samples across the world [301] and, in an analysis of cocaine seizures, the clustering of ^{1}H NMR spectra due to minor components provided important information about the origin of trafficked consignments [302].

5.2.8.2 k-means

The *k-means* is another commonly used clustering method, which assumes that there are k clusters. k-means follows the principle that each observation

is assigned to the cluster having the nearest centroid – i.e. that each cluster contains the set of observations within its Voronoi space, where "Voronoi space" partitions space into "cells" containing the subset of space closest to the nearest centroid. From an initial random partition, the observations are reassigned in an iterative procedure, recalculating the centroids as clusters are changed until no more reassignments take place. This attempts to minimise the following cost function:

$$\arg\min_{A} \sum_{C \in A} \sum_{X_i \in C} ||X_i - \bar{C}||^2 \tag{5.8}$$

Here, A is the set of all clusters, C is a single cluster (set of observations), X_i is a single observation and $\bar{C}$ is the centre of cluster C. Whilst the algorithm has seen optimisations in terms of computational performance since its inauguration, the general premise of determining the configuration of A is defined by Lloyd's algorithm [303]:

1. Assign each observation to its nearest cluster centre
2. Redefine cluster centres as the centroid of their assigned observations
3. Repeat 1, 2 until the centres stabilise

k-means is parametrised in several ways. Notably the number of clusters, k must be specified. The distance metric, $||X_i - \bar{C}||$ used to determine dissimilarity between any two vectors must also be specified, and is typically Euclidean. $\bar{C}$ represents the cluster centre and, in k-means is typically the mean of all samples within that cluster. However, it is not always possible or logical to calculate the mean of a set of samples, for instance with categorical data, and *k-centres* variations exist using other measures, such as the median.

5.2.8.3 Message passing

Frey and Dueck present an alternative clustering algorithm which they term *affinity propagation (AP)* [300]. Like median-based k-centres this algorithm does not require the variables underpinning the observations to be directly observable and only requires the similarities between individual observations to have been established. Individual observations, termed *exemplars*, act as

cluster centres . In AP each node is connected to every other node in a graph. Two "messages" values are attached to each edge of the graph:

- **Responsibility** $r(i,k)$ – sent from node $i \to k$ – can be considered as evidence supporting k as an exemplar for i
- **Availability** $a(i,k)$ – sent from node $k \to i$ – can be considered as evidence supporting whether i should select k as its exemplar

The process follows an iterative procedure which update the responsibilities and availabilities. As incoming r values accumulate indicating a node is being selected by other nodes as an exemplar, the outgoing a values of that node increase, making it more likely to be selected as an exemplar by undecided nodes. After a number of iterations the system reaches a stable state and the final set of exemplars are extracted from the edges.

This algorithm is tested by the authors on clustering faces, DNA segments and other methods demonstrating that AP performs faster and with a higher success rate than k-centres, based on true positive to false positive ratios. Another advantage of AP is that it is able to handle asymmetric similarities, when $s(i,k) \neq s(k,i)$. While AP does not take a fixed number of clusters as k-centres does it must be tuned to produce varying numbers of clusters and more information on the sensitivity and ease of selecting this tuning may be useful.

AP has been used to cluster NMR spectra in order to gain more insight into the way fibre modifies the lipid profile of different subjects [154] and also with LC-MS spectra in the identification of systemic response to ischemia [304].

Whilst the methods mentioned so far assign each observation to a single cluster, fuzzy clustering methods allow observations to belong to more than one cluster. Proteins, metabolites and transcripts may be associated with more than one function [305] so that discrete clusters may not give the best representation. Fuzzy k-means produces a "membership matrix" containing a set of weights for each observation showing the strength of association to each cluster [306]. Fuzzy clustering has been used for the classification of ^{1}H NMR spectra obtained from cancer cell line extracts and from urine samples of type 2 diabetes patients and animal models [307]. In comparison to HCA, k-means, and PCA, the authors note fuzzy k-means clustering gave

improved results and, unlike the other methods tested, allowed distinctions to be made between subtypes in the cell line data set by exploring the samples' affinity with different groups. Similar results have been noted for LC-MS studies. For instance on the use of temporal data in the analysis of signalling networks [308].

5.2.9 Performance analysis

Clustered trend data can be rapidly plotted and visualised, offering a quick indication into clustering performance. However in terms of more subtle changes, a statistical metric of performance offers a quantifiable and objective measure which can be used to optimise the clustering procedure. Potential metrics include the within cluster sum of squares (*WCSS*), the silhouette width (SW) and the Bayesian information criterion (BIC).

5.2.9.1 Within cluster sum of squares

k-means clustering aims to reduce the within cluster sum of squares, i.e. it attempts to minimise WCSS with regards to equation 5.9:

$$\text{WCSS} = \Sigma_{j \in C} \Sigma_{i \in j} |x_i - c_j|^2 \tag{5.9}$$

Where C represents the set of k clusters, x_i represents an observation i in cluster j, and c_j represents the centre of cluster j. The distance between a point and its corresponding cluster centre is here defined by a distance metric $|x - y|$.

A quick measure of clustering performance is to measure WCSS, with lower values indicating a better fit of the clustering model to the data. This value can be used to acquire a rough estimate as to an optimal value of k. With increasing k a coinciding decrease in WCSS is to be expected. However, at some point the decrease in WCSS "drops off", with increasing values of k offering little benefit in terms of reducing this value. This *elbow method* can be used to estimate a reasonable value of k based on the location of the drop-off.

5.2.9.2 Silhouette width

The determination of k using the elbow method is largely subjective. The silhouette width is an alternate measure that can be used to more precisely

gauge clustering performance. The silhouette width defines how well a point fits into its assigned cluster in relation to how well it might fit into a different cluster. This is described by equation 5.10:

$$s_i = \frac{b_i - a_i}{\max\{a_i, b_i\}} \tag{5.10}$$

Here, a_i is a measure of how well i fits into its assigned cluster, and b_i is a measure of how well i fits into the next-best possible cluster. a_i and b_i are given by the following equations:

$$a_i = d(i, g(i)) \tag{5.11}$$

$$b_i = \min_{g \neq g(i)} d(i, g) \tag{5.12}$$

$d(i, g)$ represents the fundamental measure of how well an observation fits into a given cluster. This is defined as the average distance, or dissimilarity, of that observation from all other points in the same cluster:

$$d(i, g) = \operatorname*{average}_{(j \in g)} |x_i - x_j| \tag{5.13}$$

The silhouette width for any particular observation (s_i) lies in the range $-1..1$, where values closer to 1 indicate better assignments and values below 0 indicate that the observation would be better suited to a different cluster. The values of s_i can be averaged over a group of observations, such as an individual cluster or the entire dataset, in order to determine how well that group is clustered.

5.2.9.3 Bayesian information criterion

It is natural that a model with more clusters would be able to provide a better fit to the data. An alternative performance metric is the Bayesian Information Criterion (BIC). Unlike the silhouette width, BIC applies a penalty for results with more clusters:

$$\mathrm{BIC} = -2\ln(L) + k\ln(n) \tag{5.14}$$

Here k is the number of free parameters – i.e. the number of clusters –, n is the number of observations, and L is the maximum likelihood function

value.

5.3 Results – *Medicago* dataset

5.3.1 Trend identification

Several problems were encountered when generating trends over time using an average of the replicates for each time-point with the *Medicago* data. Notably, taking the mean average of the biological replicates suffers from the presence of outliers in the dataset, whilst using the median is affected by the low number of replicates (3 or less for *Medicago*) and produced an excessively "jagged" profile.

In the case of the *Medicago* data a moving median was found to produce a smooth trend based on visual inspection. More complex smoothing methods, including polynomials and LOESS produced a trend which, whilst smooth, resulted in a deviation of the trend from the data which was readily apparent from visual inspection alone. The effectiveness of the moving median is partly unsurprising since the same smoothing function also produced a good trend during batch correction of the same data. This trend gave optimal performance as measured by the RSD of replicates and the PCA-MANOVA analysis of experimental groups cohesion given in Chapter 4, albeit for a different function of time, using $t = \textit{acquisition order}$ instead of $t = \textit{sample age}$).

5.3.2 Control correction

The *Medicago* dataset contains a large number of age-related compounds, with 1239 peaks showing significant linear correlation with age in the control group alone. Initial control correction carried out using the average of the control-group replicates resulted in noise from the control group being transferred into the other experimental groups.

Whilst the method used to obtain the trend for the control group may be different to that used to account for the replicate samples the same method (moving median, window width = 5) was found to be effective on the *Medicago* data. Figure 5.5 shows the peak shown in Figure 5.3 after control correction is applied. The relative increase of the $\mathcal{B}$ group is now apparent. Based on a comparison of m/z to the MedicCyc database the peak

was potentially identified as D-ononitol, a compound previously implicated in the stress response [309, 310].

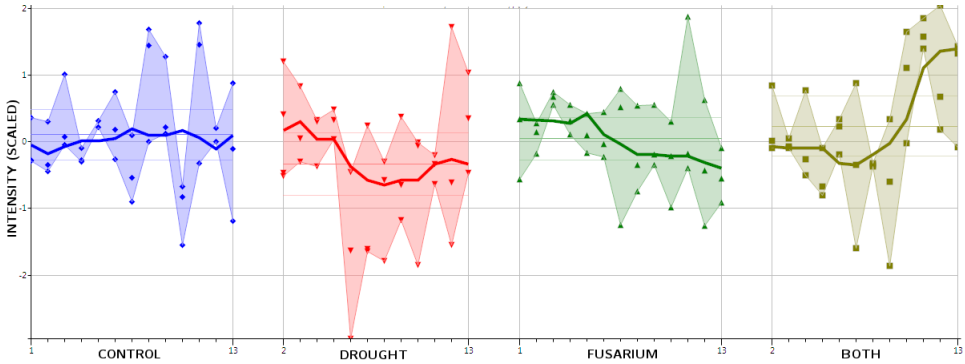


Fig. 5.5: Plot showing $\mathcal{L}+$ peak #984 (m/z = 217.068, retention time = 2.09 min, tentative identification = D-ononitol). After control correction an upwards trend in the $\mathcal{B}$ group is now apparent.

5.3.3 *Distance metric*

The plant stress response shows the presence of time-delays in the data, with noticeably fewer differences being present between the control and experimental groups in the first few days post-introduction of the stress conditions. Relationships between metabolites are also likely to show delays. The prompted the application of the Qian distance in the analysis of the the *Medicago* dataset.

The distance measure is, however, difficult to select by inspection of individual distances alone, and issues only become apparent once the the distance measure is used as a means to cluster or form a correlation network. Post-clustering it was found that a large number of Qian-indicated correlations existed due to the presence of outlier values. Figure 5.6 shows the results of k-means clustering combined with the Qian distance metric. While time-insensitive measures such as Pearson or Euclidean will be artificially low in the cases where two metabolites possess an outlier at the same-time point, this is a relatively rare scenario. Due to the nature of the Qian distance metric however, two outlier values do not need to be at the same time-point and thus a large number of outlier-based correlations are presented. As such, several clusters are present, and can be seen in the figure, which possess erratic profiles with no distinct pattern in the peaks. In

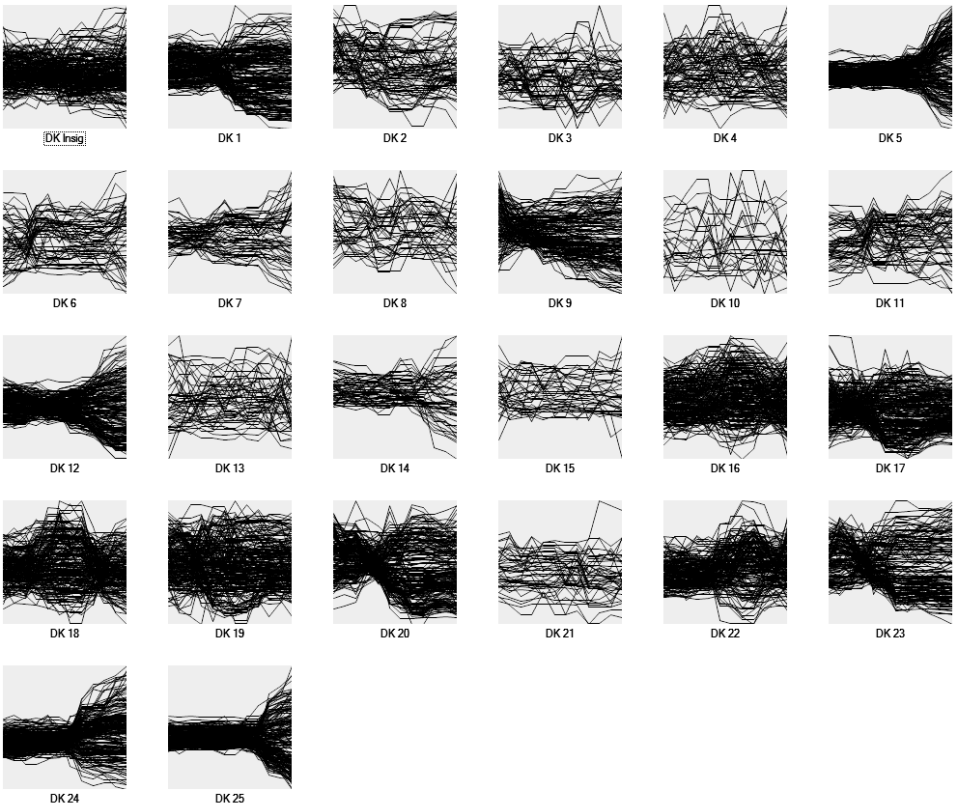


Fig. 5.6: Plots showing vectors clustered using k-means combined with the Qian distance metric. The x-axes of the plots denote time, whilst the y-axes denote intensity. Whilst several of the clusters display noticeable trends over time, a number of clusters display noisy profiles due to the presence of samples grouped together by spurious data-points.

light of this, the standard Euclidean distance metric was favoured in further analyses of the *Medicago* data. The Pearson metric meanwhile, is highly trend-sensitive, and less able to distinguish profiles following similar trends but at different magnitudes. Metabolites consistently above, or consistently below the midpoint, for example, are presented by the Pearson distance as similar.

5.3.4 Peak filtering

Initial clustering using all peaks of the $\mathcal{L}+/-$ dataset revealed a large number of "noisy" clusters with no distinct pattern over time. Inspection of the peaks within these clusters revealed a large number of "flat" trends. As noted in the introduction, these trends are therefore not themselves noisy, but appear to be noisy when taking their scaled values in isolation or using metrics insensitive to scale.

To reduce the effect of adverse trends on the clustering, peaks were filtered out of the *Medicago* data. Rather than discarding these peaks they were placed into a separate group for later analysis, as they all share the same "flat" profile.

Selecting a peak filtering method is a largely unintuitive problem. As an ad-hoc method of measuring the effectiveness of various filters the peaks were manually classified into three categories: *interesting* (those showing noticeable trends over time), *uninteresting* (those appearing to have flat trends over time) and *unknown* (those somewhere in-between).

A number of methods were tested, including measuring correlations over time, the values of piecewise regression, PCA-MANOVA and Shannon Entropy. However using the t-test method described in the methods section of this chapter produced reasonable results, achieving a 92% classification rate of the interesting and uninteresting samples. This required a cut-off point at which to classify peaks as "flat", with $p < 0.82$ giving the best performance. This value is problematic since it is unrelated to intuitively interpretable values such as *statistical significance* ($p < 0.05$). The number of time-points before which any change is seen in any experimental group is likely to increase the p value beyond those normally expected. Using $p < 0.05$ for instance, resulted in a large number of clearly flat profiles entering the clustering algorithm. The effects of selecting only a certain time-range of observations were also considered, however the performance

increase in classification rate was not deemed to justify the added complexity of the method.

5.3.5 Clustering method

The use of AP as a clustering method was hindered by the selection of the initial parameters. Whilst other clustering methods also necessitate input parameters, the time taken for the AP algorithm to converge made selection of the parameter impossible given limited time. Due to its accessibility and simplicity of configuration, the k-means algorithm was therefore used to cluster the data. The selection of k in this algorithm was selected by considering the silhouette width and BIC statistics of the resultant clusters. In total, 25 clusters of peaks were generated, comprising 1311 peaks.

5.4 Conclusions

This chapter has presented a clustering workflow and described a method of generating suitable input vectors for the clustering of time-course data. Artefacts such as age-related profiles and flat-trends, which interfere with cluster analysis have been identified and methods suggested to reduce their impact upon the analysis. Peaks from a number of the clusters obtained in this analysis have been proposed as the targets of further analyses. A number of problems have however been identified. Firstly, there are a large number of stages between data acquisition and performing the cluster analysis itself. In this chapter it has been noted that problems with the configuration of several of these stages, including the selection of the distance metric, and the requirement of control correction and peak filtering, only became apparent once the actual cluster analysis had been performed. A second issue is that the clustering algorithm itself is computationally expensive. k-means itself not only requires optimisation of k, but as the starting configuration is random, multiple runs of the algorithm must be performed in order to obtain adequate results. This compounds the first problem, limiting exploration of the analysis parameters. The next chapter will present a software-mediated dynamic workflow, whereby changes to early-stage pipeline parameters can be explored in the latter stages. A less computationally expensive variety of the k-means++ algorithm will also be presented, more suitable for rapid, exploratory analysis.

6. METABOCLUST: SOFTWARE FOR TIME SERIES ANALYSIS

6.1 Introduction

The previous chapter introduced the use of clustering in the analysis of metabolomic time series data and a number of stages of analysis prior to clustering were addressed. The entirety of this workflow is presented below:

- Data acquisition – i.e. from LC-MS
- Peak picking – identify peaks and calculate intensities
- Outlier removal - removal of both bad peaks and bad observations
- Batch correction – correct for batch differences and instrumental drift
- Trend identification – Calculate the trends over time and consolidate replicates
- Control correction – Account for changes in the control group
- Other corrections – e.g. scaling and centring
- Peak filtering – removal of "unclusterable" peaks
- Clustering – application of the clustering algorithm
- Optimisation – optimisation of the clustering algorithm
- Comparison to databases – comparison of clustered peaks to database

Given an arbitrary dataset, it is not immediately apparent which algorithms and parameters should be used at each stage of analysis, or even which stages needed to be performed. The "no free lunch" theorem [311] suggests that it is unlikely that any one technique is suited to the exploration of any and all datasets and a number of software tools have been developed that permit a more exploratory analysis of metabolomic data, for example

XCMS [124], XCMS Online [312] and MetaboAnalyst [313, 314]. Of these programs some provide a graphical user interface (GUI), which can reduce the learning curve required in order to obtain meaningful information from metabolomic datasets.

Several existing tools provide a web based "thin-client" interface, which are advantageous in that they facilitate access to users and are easier to maintain. Disadvantages include the fact that they are technologically constrained with regards to the size of the dataset permitted and the level of interaction they can provide, given the time taken to move the initial dataset across the network. As such these applications typically favour a direct input-to-report workflow, which can limit the ability to explore certain methods and parameters.

In this chapter a graphically interactive software package for metabolomic time series analysis is presented, and two case studies are used to demonstrate its use. The software, named "MetaboClust", takes into account some of the limitations enforced by web-based interfaces and operates as a standalone application, allowing fast, highly visual, interactive data exploration and making use of clustering methods to investigate patterns in biological time series data. This novel workflow is ideally suited to large untargeted studies where the patterns of interest may not be known at the outset and therefore the software provides visualisations at all stages of analysis and allows the user to navigate quickly between clusters, features, metabolites and pathways. The use of dynamic workflows allow the user to explore the effects of potential data manipulations at stages further along their set of planned changes. The overall objective of MetaboClust is to support the user in creating and exploring time series trends, locating similar trends and identifying the potential metabolic pathways they relate to.

During the development of this software it became apparent that the speed or optimisation requirements of clustering algorithms such as k-means of affinity propagation were not amenable to the rapid exploration of data. A deterministic modification of the k-means starting configuration is therefore also presented in this chapter, with a specific goal to facilitate rapid data exploration.

The peak-picked data from the *Medicago* and *Alopecurus* datasets are used as case studies to demonstrate the software. These were previously discussed in Chapter 3.

6.2 The Software

6.2.1 Implementation

The software was developed using Microsoft C# and the .NET framework, which provides a library of reusable elements (the *Framework Class Library*) and tools. In particular the Windows Forms library provides an out-of-the-box means to interactively and iteratively develop a familiar user interface, which has been ported to a number of operating systems and platforms. The *R.NET* library is used to provide an in-process interoperability bridge to the R script interpreter [315]. The inclusion of *R* [284] allows complex statistical analysis to be performed, allowing users to incorporate their own methods and providing easy access to the wide variety of algorithms developed for R. A number of other mathematical functions, particularly the distance metrics, make use of the system native Math.NET Numerics package [316].

The major components of a data-driven metabolomic analysis are presented in the UML diagram shown in Figure 6.2. These fall into four broad categories:

- The experimental data – Intensity matrices, peaks and observations
- The configuration – Which algorithms have been applied and to where
- The annotation data – Compounds and pathways, and the annotations placed upon the peaks
- The algorithms – The actual algorithms available, of which four types have been identified

Of note here is the high degree of interdependency of various components. Clusters can, for instance, be summarised as their constituent peaks, peaks by their compounds and compounds by their pathways. Several features can be also be reused, in particular trend algorithms are used in batch correction, control correction and in the condensation of replicates. The design of the software takes these features into account, allowing exploration along the network of related elements, for instance, allowing the user to summarising the relationship between an individual cluster, and the pathways represented by the compounds annotated on its peaks.

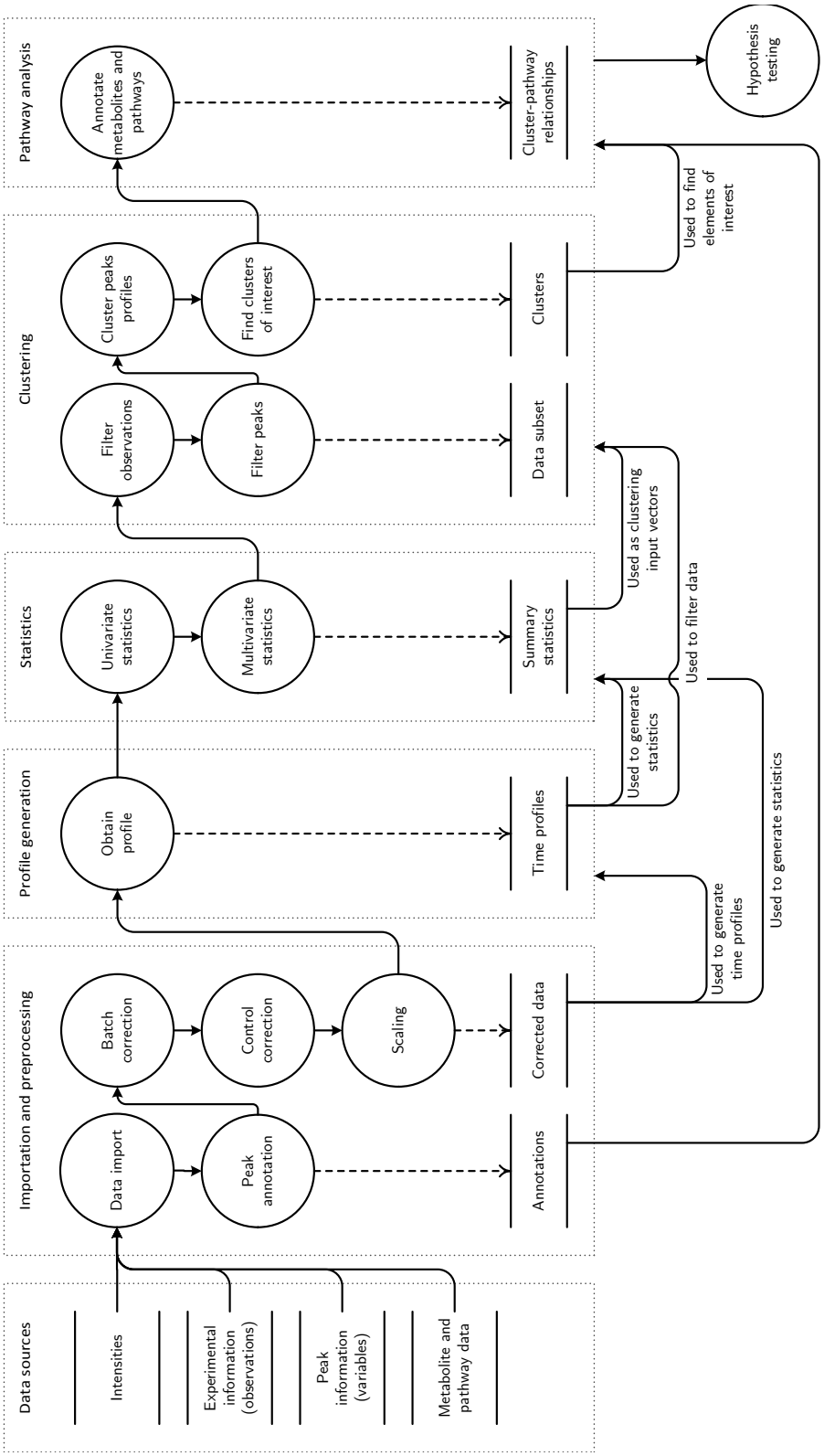


Fig. 6.1: Diagram describing the workflow implemented by MetaboClust.

The flow of information between the stages of the clustering workflow is presented in the flow chart shown in Figure 6.1. Whilst this is a largely feed-forward process, poor results at one stage can necessitate revisiting earlier analyses. The MetaboClust workflow is therefore designed to be as fluid as possible, allowing the impact of various parameter selections on one aspect to be immediately reflected in the others, allowing the user to prepare the analysis in the order they see fit.

6.2.2 Workflow

6.2.2.1 Data import

Data is initially imported into a software "session" via a guided importation wizard, which prompts the user to specify the dataset to be used in their analysis. The session, shown to the bottom left of Figure 6.2, includes the *experimental data* and the *annotation data*, comprising:

The intensity matrix – A data matrix with rows corresponding to observations and columns corresponding to integrated peak intensities.

Observation information – Details on experimental observations are required if certain statistical analyses are to be conducted. These include the the *experimental group*, *time-point* and *replicate number* of each observation.

Peak information – Details on peaks are optionally taken for later reference, peak m/zs may be imported to allow to allow peak annotations to be made based on m/z values.

Metabolite database – Information on metabolites is required for automated annotation. Details of relevant metabolic pathways are required for pathway analysis. The software is able to import databases in the BioPAX pathway exchange format [317], in addition to providing the data as a spreadsheet (CSV). Manual identifications (obtained for example via XCMS or Progenesis QI) can be loaded to replace or augment the automated annotations.

Adduct database – For automated identification of LC-MS data a list of potential ion adducts is required.

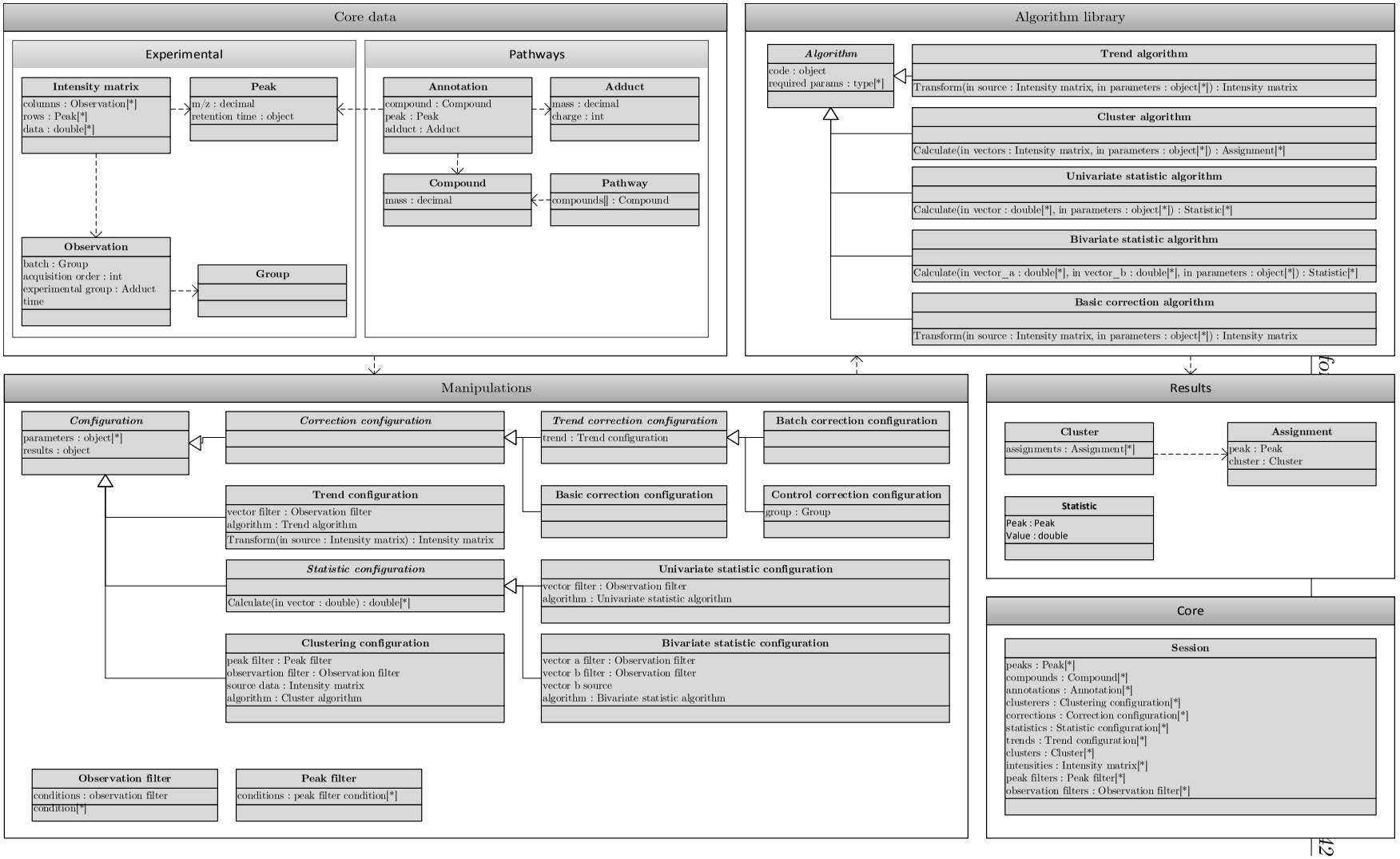


Fig. 6.2: UML diagram depicting the major components of a data-driven metabolomic analyses. Some connections have been hidden for clarity.

The software accepts the data in CSV format due to wide support for the format in existing conversion utilities and spreadsheet programs. Once created this it is saved back to disk by the software into a native binary format (MS-NRBF [318]) for speed of access.

When the session is loaded *the algorithms* are automatically read from disk and the session *configuration* is populated with a few frequently used statistics, such as t-tests.

6.2.2.2 Data exploration and pre-processing

PCA and PLSR can be performed in software in order to aid decisions on the pre-processing methods selected.

6.2.2.3 Data correction

Following the workflow outlined earlier, MetaboClust provides four sets of correction functions C:

- Scaling – $C_S(M, P_S)$
- Control – $C_C(M, C_T)$
- Batch – $C_B(M, C_T)$
- Trend – $C_T(P_T, M, F)$

Here, P_S and P_T are algorithm-specific parameters and M is the input matrix. The trend algorithm, C_T also accepts a filter, F, defining an observation subset over which to obtain the trend. Whilst the algorithms have different purposes in each case, these corrections share common requirements – M and C_T – and such can be visualised in similar manners by plotting the input matrix M, with optional trend C_T, adjacent to the output matrix M'.

Batch correction techniques have been discussed in Chapter 4 and follow the general formula presented in equation 4.1. Control correction and trend generation techniques were mentioned previously in Chapter 5. Scaling correction formulae are algorithm specific, and several were outlined in Chapter 2. As an example, in UV scaling a vector x is scaled as $x' = x/\delta$, where δ is the standard deviation.

6.2.2.4 Univariate analysis

MetaboClust incorporates a number of univariate analysis including univariate statistics (e.g. the mean) and bivariate statistics (e.g. t-test, Euclidean distance). These can be combined with the aforementioned observation filters on specific observations to provide a range of possible investigation points, for instance to t-test an experimental group against control.

6.2.2.5 Clustering

The software offers several inbuilt clustering methods including HCA, affinity propagation [300] and k-means.

As in all stages external R scripts can be implemented or existing ones modified where more esoteric solutions are desired. In the previous chapter the necessity of isolating specific peaks from the clustering algorithm was noted. Filters on peaks can be specified during clustering to sequester peaks that either interfere with the algorithm or are known to not be of interest.

6.2.2.6 Pathway analysis

Post-clustering the software permits the user to browse the potential compounds and pathways implicated by the clustered data. This is accomplished by backtracking from the clusters, to their assigned peaks, to their potential metabolites, to their implicated pathways. It is possible to obtain a scores relating the degree of overlap between a cluster and a pathway, based on the number of compounds in the pathway, the number of those potentially represented by peaks in the data, and the number of those peaks occuring in the cluster. Clustering results can therefore be sorted by the degree of overlap between clusters and pathways and the results visualised through plots.

6.2.2.7 d-k-means++

Due to the random nature of the initialization step, k-means is non-deterministic and can be highly sensitive to the initial starting conditions, resulting in multiple runs of the algorithm yielding different results [319]. When large numbers of observations and clusters are present this becomes increasingly computationally expensive and may not be possible to achieve in a satisfactory time. Another disadvantage of k-means-like procedures is that it requires

the number of clusters to be stated up-front, in turn this may require yet more runs of the algorithm to be performed if the number of clusters present is initially known. In the case of clustering metabolites or genes this is often the case.

For the purposes of rapid-clustering in software the *d*-*k*-means++ initialisation method is presented. Rather than selecting a random set of starting centres the initial centres are selected following an iterative procedure:

1. Select a centre at the edge of the search space, for instance one showing high Pearson correlation with time
2. Compute the squared distances D between each observation, X_i, and the nearest cluster centre.
3. Select the furthest observation from any centre (D_{max}) and assign as a new centre
4. Repeat 2, 3 until k exemplars have been chosen or until D_{max} falls below a threshold D_{stop}

As this method gives more weight to the selection of observations furthest from existing centres it therefore potentially results in better initial coverage of the search space. A second advantage of this method is that it does not require a cluster count, k, to be specified in advance. Instead cluster generation can be ended when the search space is adequately covered, as determined by the distance of the observation selected in the third step ($\arg\max_i D_i$) from any existing cluster centre being below the threshold parameter D_{stop}.

This procedure is fundamentally similar to the k-means++ algorithm, with differences being that k-means++ assigns the first centre based on a random, uniform distribution, and selects subsequent observations using a probability distribution, where the chance of an observation being selected proportional to $D_i^2/\Sigma_j D_j^2$. k-means++ is advantageous in that the first "seed" point is intelligently selected based on existing information, and the iteration steps maximise search space coverage. It is limited by its deterministic nature however, in that repeating the algorithm can not yield better results.

Adduct	Charge	Mass Difference
-H	−1	−1.00728
-2H	−2	−1.00728
$-H_2O-H$	−1	−19.0184
+H	+1	1.007276
+Na	+1	22.98922
+K	+1	38.96316
$+NH_4$	+1	18.03382
+2H	+2	1.007276

Tab. 6.1: Table of adducts used for m/z based peak annotation for our case study. These represent a subset of the data found at [321].

6.3 Case study 1: Analysis of drought and disease in the model plant Medicago truncatula

The purpose of this study is to identify metabolites in *Medicago truncatula* responsive to the experimental conditions: $\mathcal{D}$, $\mathcal{F}$ and $\mathcal{B}$ in relation to $\mathcal{C}$. In particular, the aim is to highlight potential key biological pathways associated with metabolites that are being elicited or suppressed under the stress conditions.

6.3.1 Data import

The MetaboClust import wizard was used to input CSV files containing the *Medicago* data and a list of 1847 compounds known to be present in *Medicago truncatula*. This compound list was downloaded from the MedicCyc database [320] and includes information for 407 pathways as well as mono-isotopic masses for the metabolites. A further CSV file containing mass and charge information for the eight possible adducts shown in Table 6.1 was also imported into the software. These 8 adducts represent a subset of the data published by Kind [321], filtered to exclude infrequently occurring adducts.

6.3.2 Exploration and pre-processing

It is already known that batch differences are present in the *Medicago* dataset. PCA, conducted within the software, revealed notable differences between LC-MS batches, with the between-batch variance overriding the

variance between experimental groups. Signal correction using QC ion intensities for each batch successfully resolved the batch differences in the positive ion mode dataset. It was also immediately apparent from the MetaboClust data correction visualization (Figure 6.3) that the negative ion mode data was not amenable to this correction. PCA scores plots (not shown) further demonstrate that batch differences are made worse by this method. For the negative data, the "background correction" (Chapter 4 was therefore applied [322]. Figure 6.4 shows the software's preview window, demonstrating the effectiveness of this correction method. After batch correction, the data were scaled to unit variance and mean-centred, in order that all features be given equal weight in further analysis. Invalid floating-point values (NaN, resulting from division by zero errors during batch correction) were set to 0 following the recommendations outlined in [251].

Viewing the data also revealed that a large number of peak profiles showed a trend over time in the control-group, likely to represent age-related compounds. As the changes induced by the experimental conditions ($\mathcal{D}$, $\mathcal{F}$, and $\mathcal{B}$) are of primary interest, age-related trends were accounted for by control correction. Whilst the simplest method to account for a trend in the control replicates would be to average over the replicates at each time point, visualisation in MetaboClust shows that this method transfers noise present in the control group to the other experimental data. Using the moving average (as applied for batch correction) with a window width of 5 days was effective in accounting for the general trend of the control group without transferring noise. Again, the median, rather than the mean, was used in order to reduce the effect of outlier values present in the data.

The trend lines (clustering input vectors) were generated using the same smoothing method applied to batch and control correction – a moving median across t for each experimental group, with a window width of $w = 5$.

6.3.3 Univariate statistics

As outlined in Section 5.3.4 "flat" profiles were identified via comparison of the experimental group observations to the control group observations using the rule $p < \alpha$, where α was set to 0.82. This yielded a data subset comprising 1577 peaks flat across all experimental groups and 1311 peaks designated as inputs to the clustering algorithm.

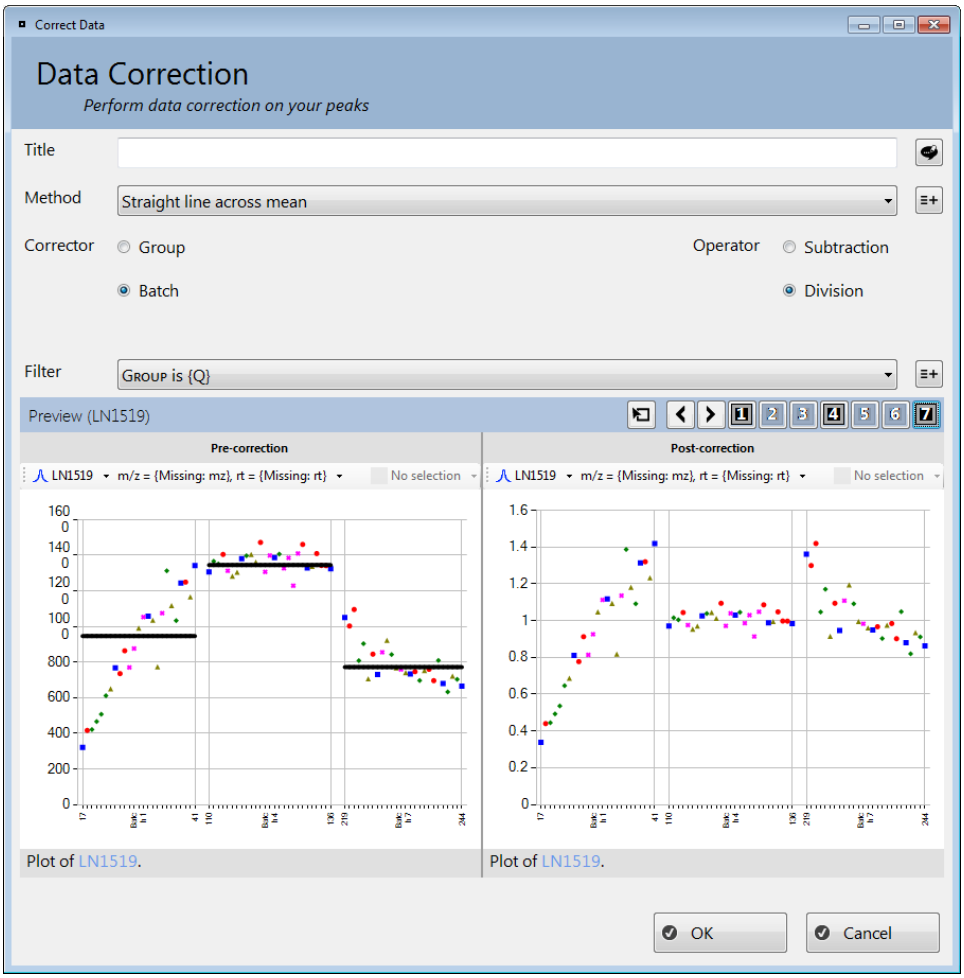


Fig. 6.3: Image showing the in-software preview displayed for a mean-of-the-QCs batch correction. Variations in intensity (Y) between batches and along the acquisition order axis (X) can be seen post-correction.

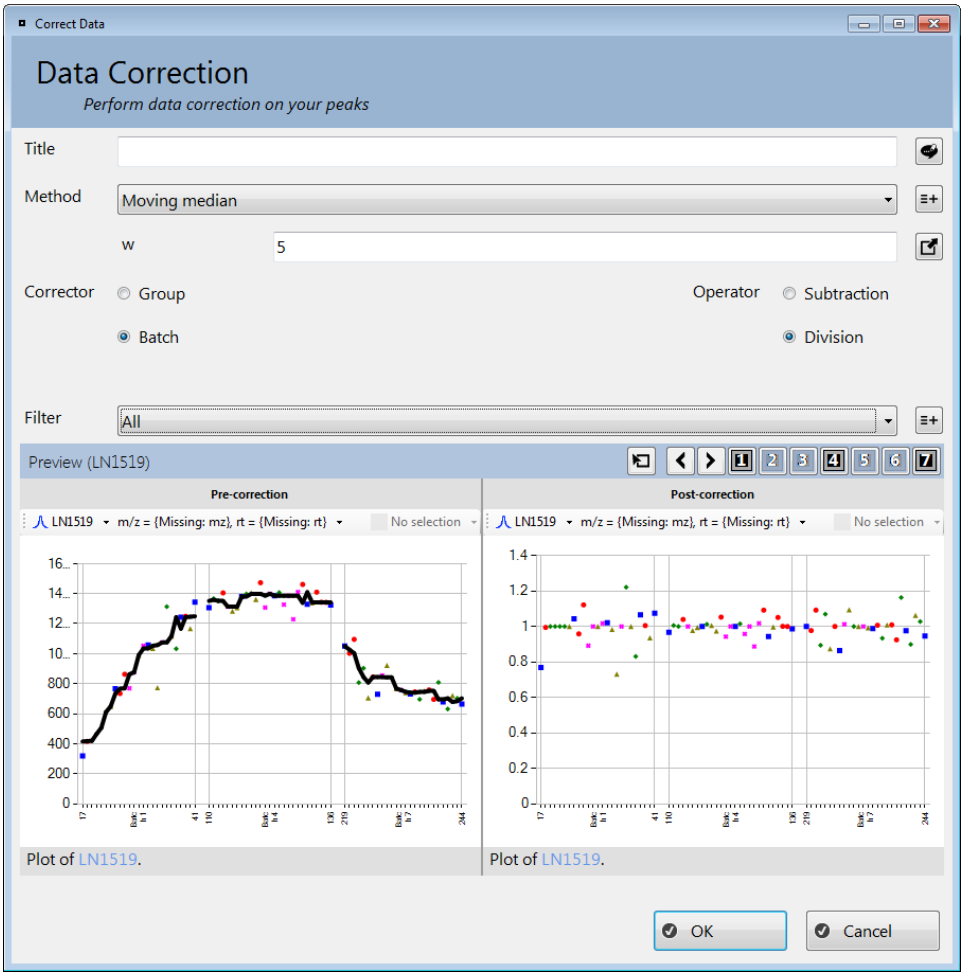


Fig. 6.4: Image showing the in-software preview displayed for a moving median batch correction (window width = 5) of a particular peak. The post-data correction can be seen to be free of the changes in intensity (Y) with acquisition order and batch (X) that are encountered if a linear correction model is used.

6.3.4 Clustering

Exploratory analysis revealed the differences in time profiles across the dataset to have a continuous nature rather than forming discrete clusters, as can be seen in the PCA plot of the input vectors shown in Figure 6.5. This is confirmed by metrics for clustering performance, whose values are plotted in Figure 6.6. As the number of clusters (k) increases, the silhouette width performance statistic shows a rapid decrease in performance, with the best clustering being performed for $k = 2$. The BIC performance statistic reveals similar results. This makes the number of clusters largely subjective. Whilst too many clusters makes the identification of common patterns difficult, too few increase the complexity of individual clusters and thus fail to provide usable information in terms of a coherent set of profiles.

We found the k-means++ algorithm with $k = 25$ produced good similarity between the time profiles within clusters without having clusters that were too similar to each other. The d-k-means++ algorithm produced similar results, again with $k = 25$ (available in Appendix B. The average deviation from the cluster centre for each metabolite was $D = 2.15$ for d-k-means++ in comparison to $D = 2.11$ for k-means, optimised over 1000 runs. The closest 10% of metabolites have an average distance of $D_{closest10} = 1.11$ with k-means and $D_{closest10} = 1.07$ for d-k-means++. The metrics show that differences between the two methods are small and we therefore favour the d-k-means++ algorithm over the much slower k-means.

Whilst differences between the control and *Fusarium* groups were not apparent for individual time-points, the cluster analysis revealed time profiles that differed between the two groups, for example clusters `DK7` and `DK8`, shown in Figure 6.7. Furthermore substantial differences between the profiles of drought and dual stressed plants were highlighted (e.g. clusters `DK24` and `DK25`).

6.3.5 Pathway analysis

In the overview previously referenced (Figure 6.7), different response profiles to the experimental conditions can be seen in the 25 clusters generated. Cluster `DK18` shows a group of compounds that increase in intensity over time for the dual-stress group, whereas the profiles for both the drought and *Fusarium* groups show a different behaviour, dropping in intensity beyond

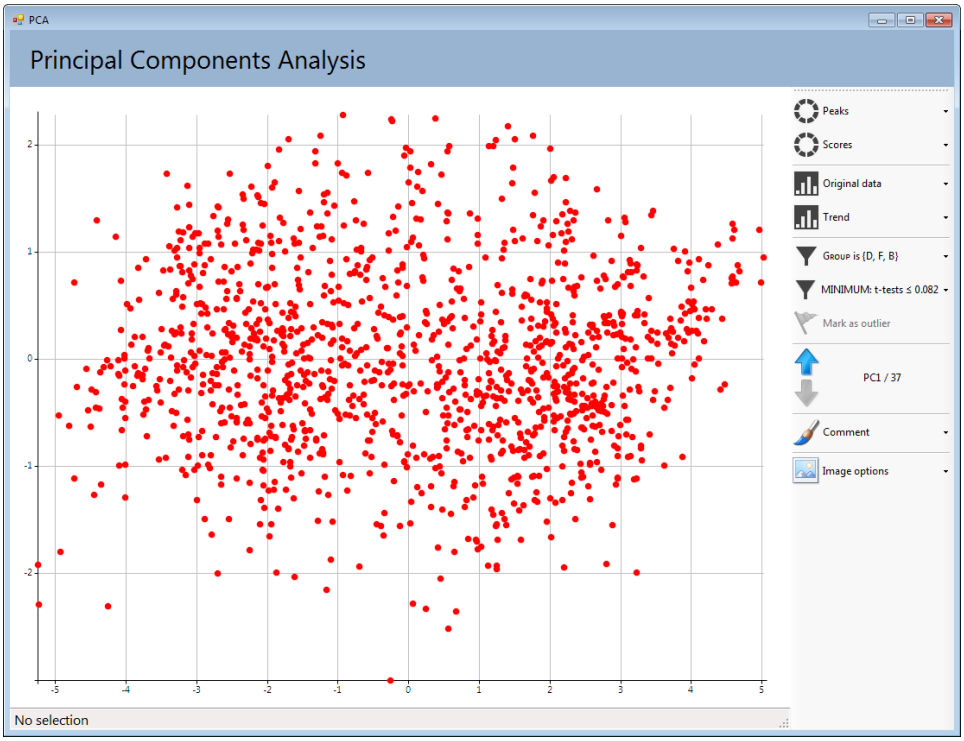


Fig. 6.5: PCA plot of the input vectors used in the clustering model. No distinct clustering is visually apparent. The trends were generated from the Drought, *Fusarium* and Dual-stress group samples for each peak. Peaks were filtered to exclude those not showing significant deviation from the control group for any of the profiles based on $min(t) < 0.082$.

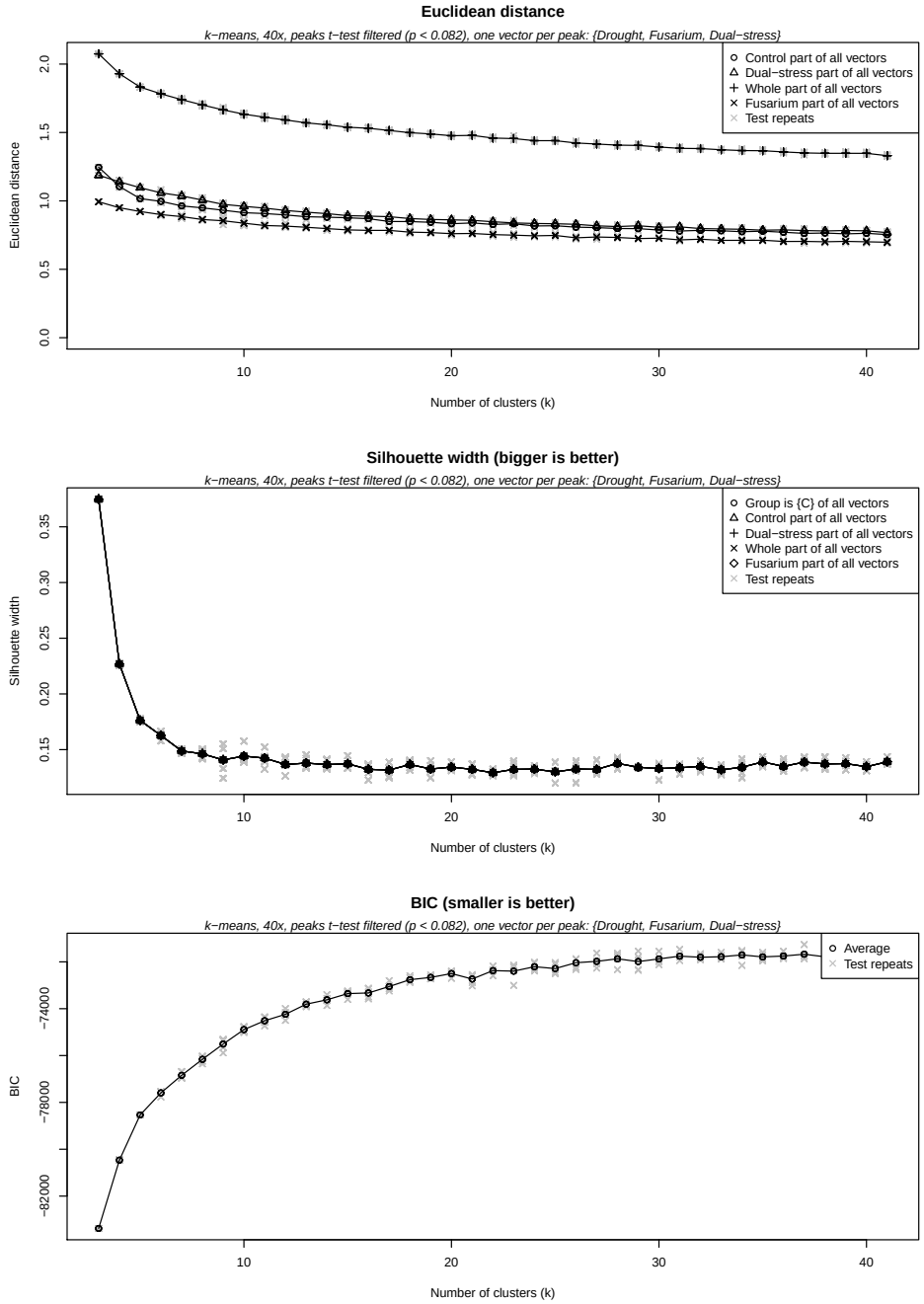


Fig. 6.6: Results of various runs of the k-means clustering algorithm, displayed in terms of two measures of performance: Silhouette width and BIC. The average euclidean distance from the cluster centres is also shown for comparison. These results show decreasing performance with increasing k.

$t \approx 8$. DK18 contains 36 input vectors (i.e. 36 peaks), of which 15 possess tentative compound identifications.

In-software analysis shows that the compounds potentially represented by the peaks within this cluster are present in a number of different biological pathways, with protein biosynthesis in the *tRNA charging* pathway having the highest number of distinct compounds associated with this cluster. Figure 6.8 shows the overlap between cluster and pathway as displayed in MetaboClust. The compounds in the database for this pathway comprise the set of 20 standard amino acids. The accumulation of amino acids in drought stressed plants has been known for some time in the literature [323, 324] and a recent study has suggested the application of free amino acid to wheat enhances drought performance [325], implicating the role of this increase as a tolerance mechanism, rather than as an artefact of injury.

Cluster DK19 also shows a group of features with very similar time profiles for the drought and dual-stress groups. This cluster comprises 87 features and shows good overlap with the *TCA Cycle* pathway[1]. The database contains 29 compounds for this pathway, of which 9 have tentative annotations against peaks in the *Medicago* dataset. Some of these metabolites have been tentatively assigned to more than one peak, giving a total of 16 profiles that could be related to compounds in this pathway. Figure 6.10 shows how the time profiles within a cluster that can be associated with a particular pathway are highlighted in-software. Figure 6.11 shows that the converse can also be visualised, i.e. that of all time profiles that could potentially be associated with a particular pathway, those within the same cluster (and therefore having a similar trend) can be highlighted.

6.4 Case study 2: Comparison of phenotypes of *Alopecurus myosuroides*

6.4.1 Data importation

As no database specific to *Alopecurus* could be found, databases for several different plant species were downloaded from the PMN database collection [326] in addition to the *Medicago* database used in the first case study (available from MedicCyc [320]) to cover as many metabolites as possible. These are shown in Table 6.2. All are available in the PathwayTools database

[1] PWY-5913: TCA cycle VI (obligate autotrophs)

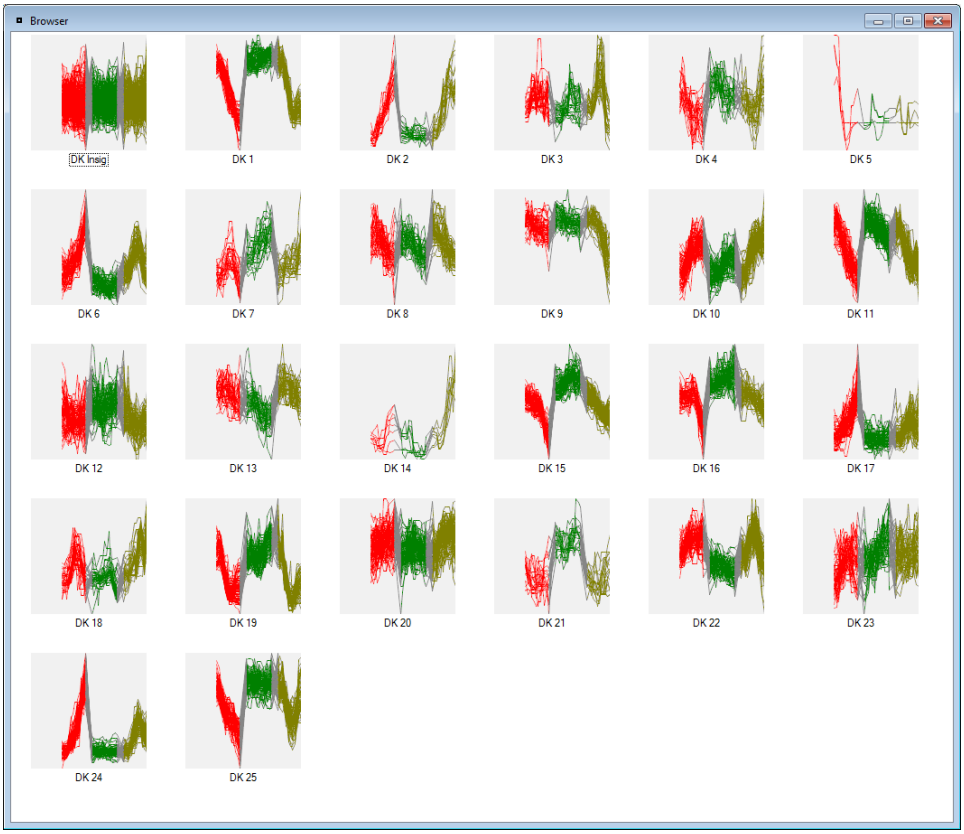


Fig. 6.7: Screen-shot of the software showing the cluster explorer post-clustering in thumbnail view. The 25 clusters are visually displayed by the plot of the input vectors assigned to them. The input vectors are coloured according to from which experimental group each element was selected from and are sorted by experimental group and time. From left to right for each group the coloured portions are: Left, red, drought group, days 2-13. Centre, green, *Fusarium* group, days 1-13. Right, ochre, dual-stress group, days 2-13. The cluster centre is represented by the bold dashed line (purple).

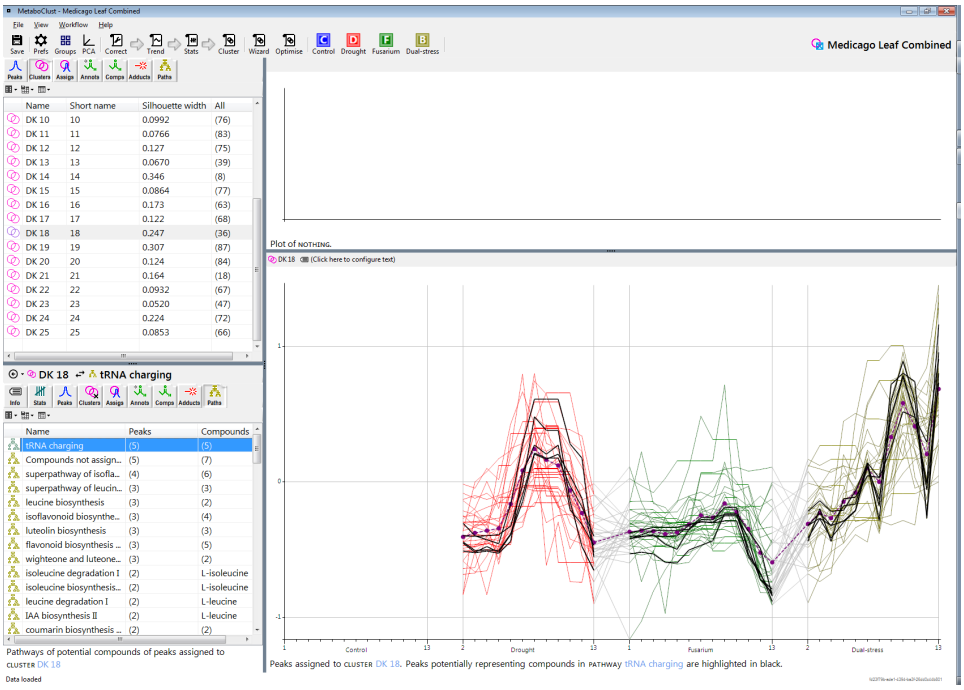


Fig. 6.8: Screen-shot of the software showing the overlap between the tRNA charging pathway and cluster "18". The plot in the bottom right shows the trends of peaks assigned to cluster DK18, with peaks potentially representing compounds in the tRNA Charging pathway based on their tentative annotations highlighted in bold-black.

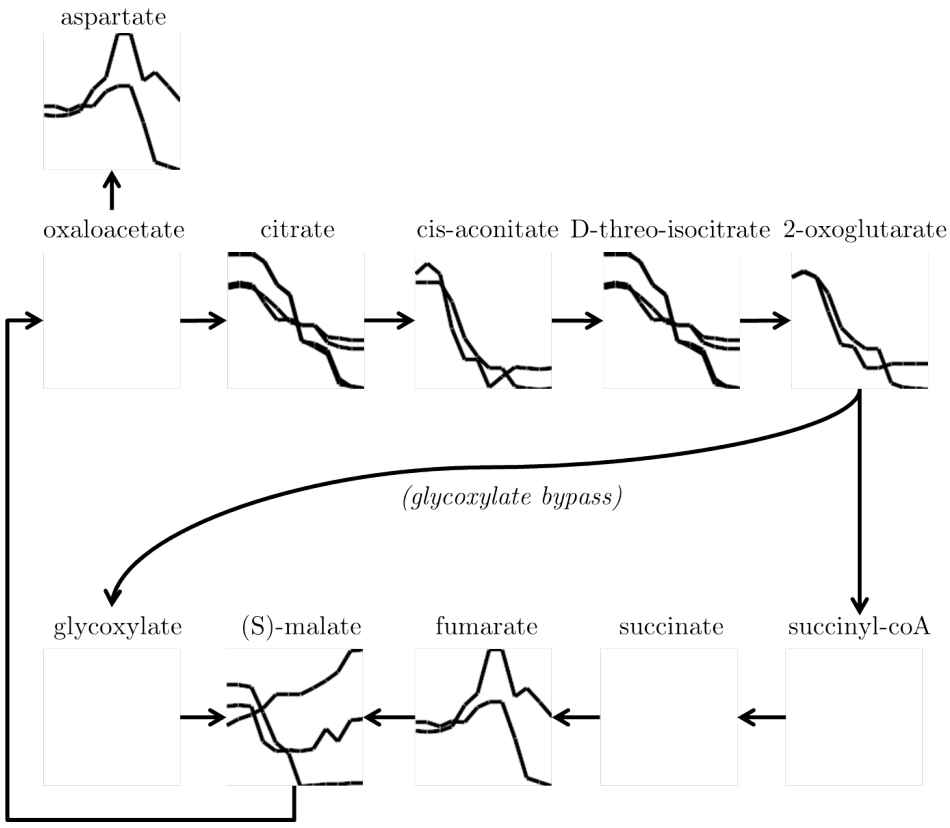


Fig. 6.9: Depiction of the citric acid cycle. The plots show the intensities of peaks coincident to the named compounds based on their m/z. Intensities are shown using their defined trend (a moving median) for drought-stressed ($\mathcal{D}$) plants between days 2 and 13, relative to control ($\mathcal{C}$). Empty plots signify that no matching peaks were found based on their m/z values. Note that two pairs of compounds, citrate and D-threo-isocitrate, as well as fumarate and aspartate, are isomers and therefore share the same plots due to having the same m/z.

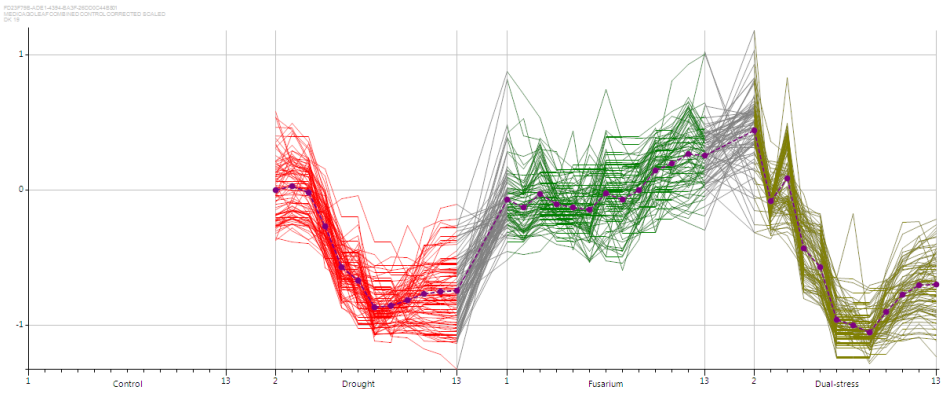


Fig. 6.10: Figure showing the plot shown in the software for cluster #19. This cluster comprises 87 input vectors for 87 peaks. The input vectors each contain concatenated data from the three experimental groups and the portions of each vector are coloured according to which group each datapoint comes from. The data for each vector are sorted in experimental group and time order. From left to right for each group the coloured portions are: Left, red, drought group, days 2-13. Centre, green, *Fusarium* group, days 1-13. Right, ochre, dual-stress group, days 2-13. The cluster centre is represented by the bold dashed line (purple).

format, which can be imported into MetaboClust. The *Alopecurus* data itself, along with the adducts list were imported as CSV files.

6.4.2 Data pre-processing and exploration

Peak intensities were UV scaled and mean centred in software. Batch correction was not required as all data were acquired in one batch per ionisation mode and, as no control group was available, no correction for age-related effects was performed.

To explore the response profiles, t-tests were calculated for each peak to compare the final two time-points between each pair of experimental groups ($\mathcal{M}$-$\mathcal{T}$, $\mathcal{T}$-$\mathcal{S}$, $\mathcal{S}$-$\mathcal{M}$). Time profiles for two features shown to be highly significant for the comparison of plants tolerant to multiple herbicides with those tolerant to specific herbicides ($\mathcal{M}$-$\mathcal{T}$) are shown in Figure 6.12 ($p = 6.3e-8$ and $p = 3.5e-6$). It is clear that the trends in the two profiles are very different and in fact, several different trends result in a significant difference between the final two time-points for these groups.

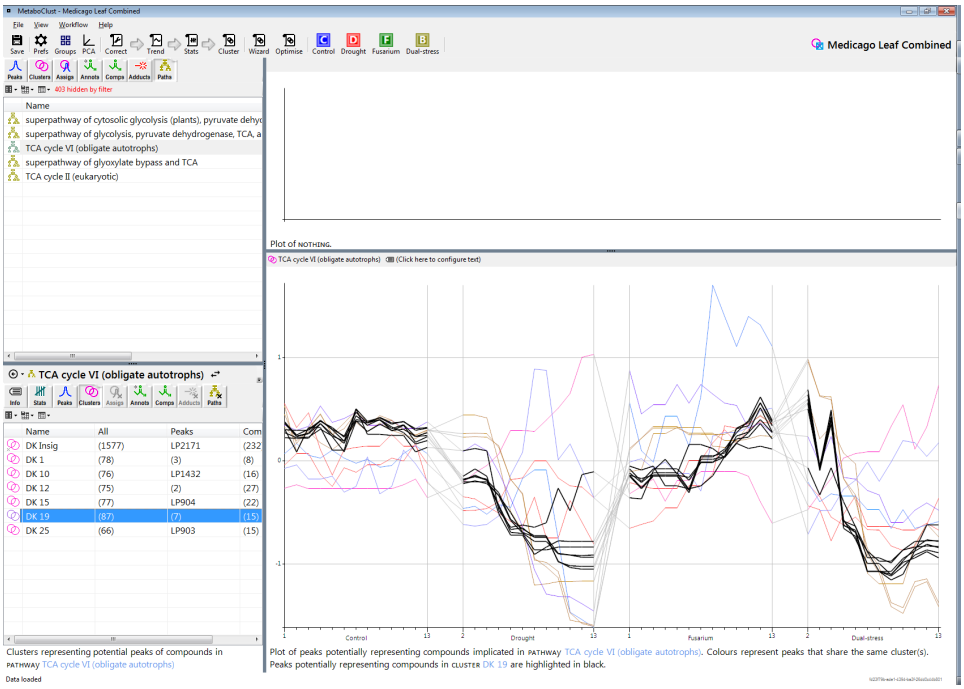
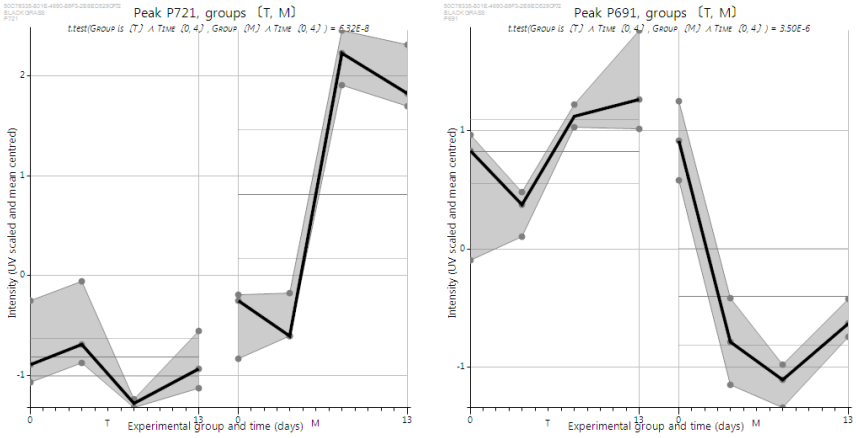


Fig. 6.11: Screen-shot of the software showing the overlap between the "superpathway of glyoxylate bypass and TCA" pathway and cluster "19". The plot in the bottom right shows the trends of peaks potentially representing compounds in the glyoxylate bypass/TCA pathway based on their tentative annotations. Peaks assigned to cluster "19" are shown in bold-black.

AraCyc 13.0	BarleyCyc 3.0	BrachypodiumCyc 3.0
CassavaCyc 5.0	ChineseCabbageCyc 3.0	ChlamyCyc 5.0
CornCyc 6.0	GrapeCyc 5.0	MossCyc 4.0
OryzaCyc 3.0	PapayaCyc 4.0	PoplarCyc 8.0
PotatoCyc 2.0	SelaginellaCyc 4.0	SetariaCyc 3.0
SorghumBicolorCyc 3.0	SoyCyc 6.0	SpirodelaCyc 1.0
SwitchgrassCyc 3.0	TomatoCyc 1.0	WheatACyc 1.0
WheatDCyc 1.0	MedicCyc*	

Tab. 6.2: List of the databases used in our case study on *Alopecurus*. *With the exception of the MedicCyc *Medicago* database, taken from [320] these were downloaded from the PMN database collection available at [326]

.



(a) Positive ion mode peak #721, $p = 6.3E-8$

(b) Positive ion mode peak #691, $p = 3.5E-6$

Fig. 6.12: Plot of time and experimental group versus intensity for two peaks (Positive ion mode peaks #721 and #691). Both of these peaks show significant differences ($p\langle 0.01$) in a t-test comparing the intensities of the the observations for first two time points ($t \in \{1, 4\}$) between the $\mathcal{T}$ and $\mathcal{M}$ experimental groups. Whilst both peaks test as significant they have markedly different trends over time.

6.4.3 *Cluster analysis*

Using the MetaboClust visualisation to determine the effectiveness of different smoothing algorithms, taking the median of the replicates for each time point was found to be sufficient to generate time profiles. In contrast to the noisier *Medicago* dataset, with more time-points, a more complex smoothing function was not required.

Attempts were made to optimize the number of clusters using the silhouette width (s_i) and Bayesian Information Criterion (BIC) clustering performance measures. However, like for *Medicago*, both measures suggest the strongest clustering performance for the lowest value of $k = 2$, with worsening performance for greater values of k. These results, shown in Figure 6.14 are again probably due to a lack of discrete clusters in the data. However, visualization showed that differences between full time profiles (across all experimental groups) were often due to differences in just one of the experimental groups. We therefore performed cluster analysis using separate input vectors for each group. In this case, the BIC no longer shows a gradual

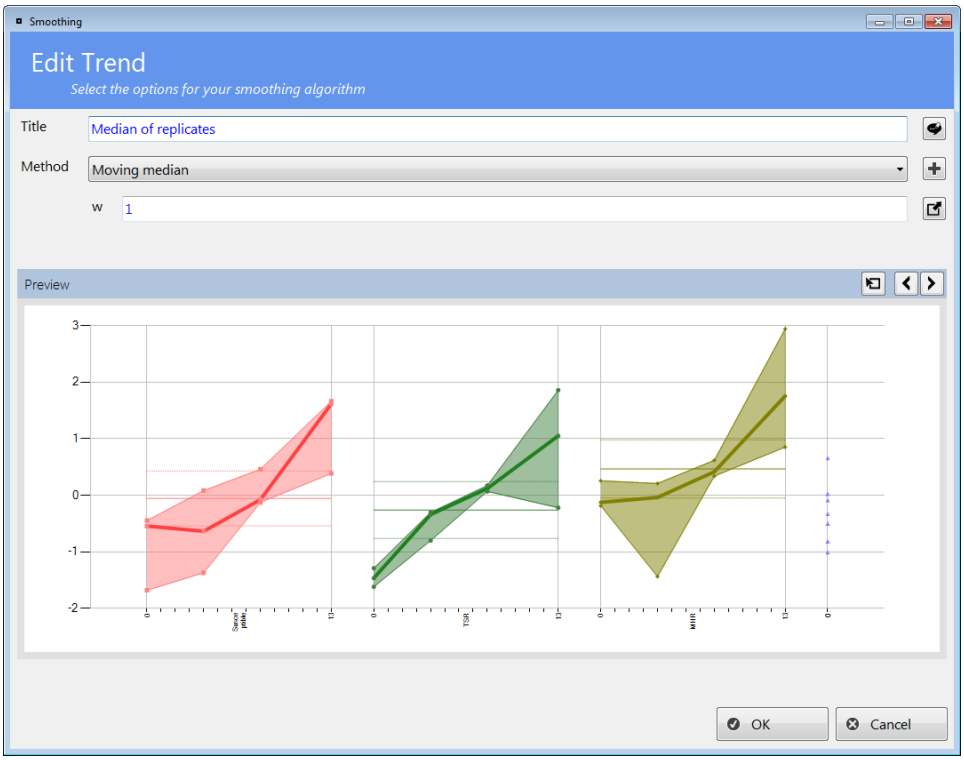
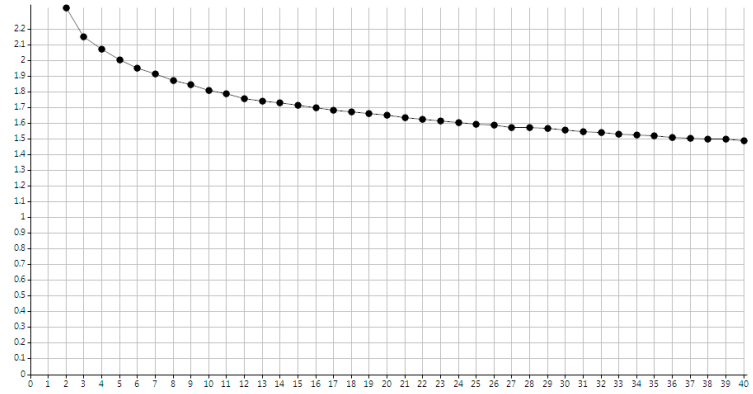


Fig. 6.13: Screen-shot showing the trend line generation window. Using the median of the replicates for each time point was found to produce a relatively smooth trend.

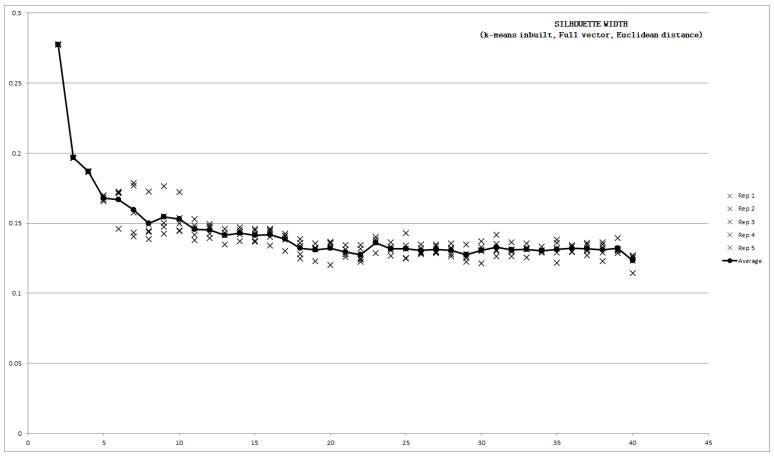
decrease in clustering performance with k, but instead shows a second peak in performance (after $k = 2$) at $k = 10$, as shown in Figure 6.15. This value of k remains consistent with when the deterministic clustering d-k-means++ algorithm is applied instead of k-means.

6.4.4 Pathway analysis

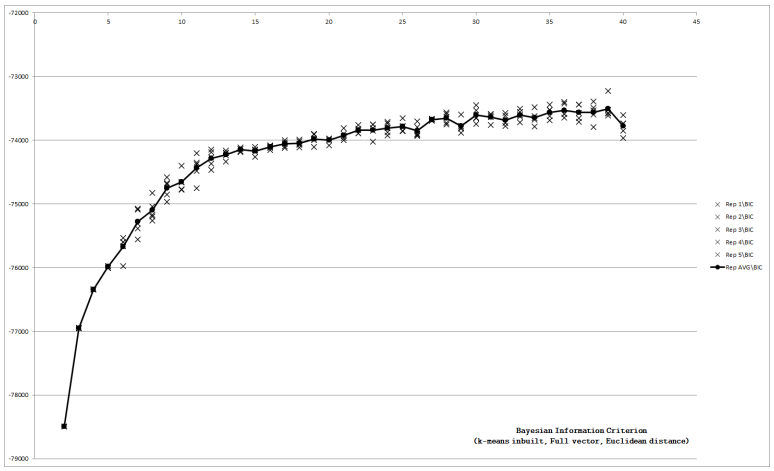
A number of interesting patterns can be observed using the group-wise clustering. For example, cluster 2 (Figure 6.16) includes profiles showing a decrease with time. The pathway breakdown in Table 6.3 shows the association of various pathways with cluster 2, with *Brassinosteroid biosynthesis* showing the strongest degree of overlap. Brassinosteroids are known to be an important class of hormonal regulators and have already been implicated in plant stress response [327].



(a) Average euclidean distance from cluster centre

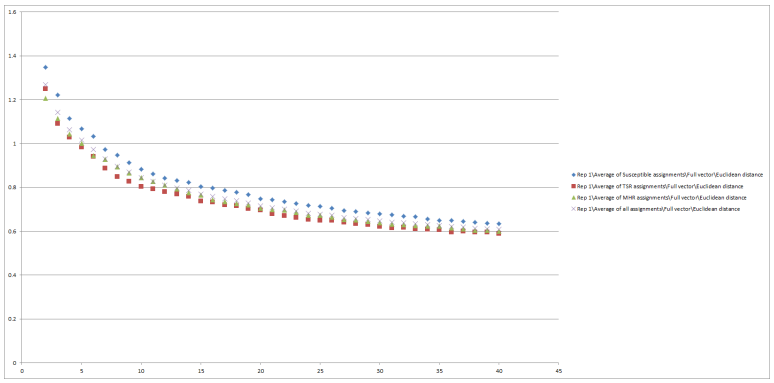


(b) Average silhouette width

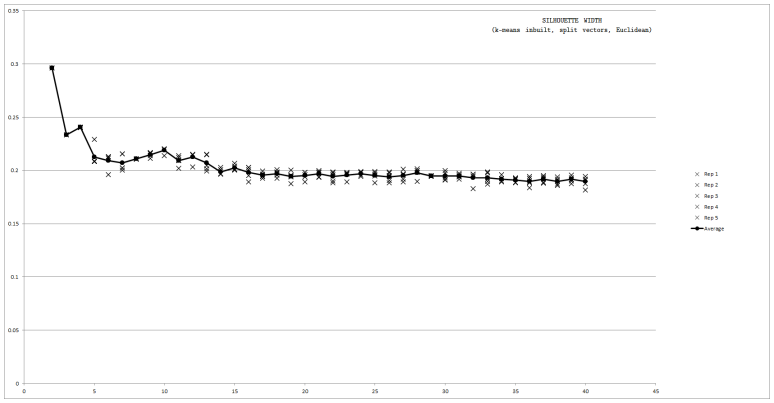


(c) BIC

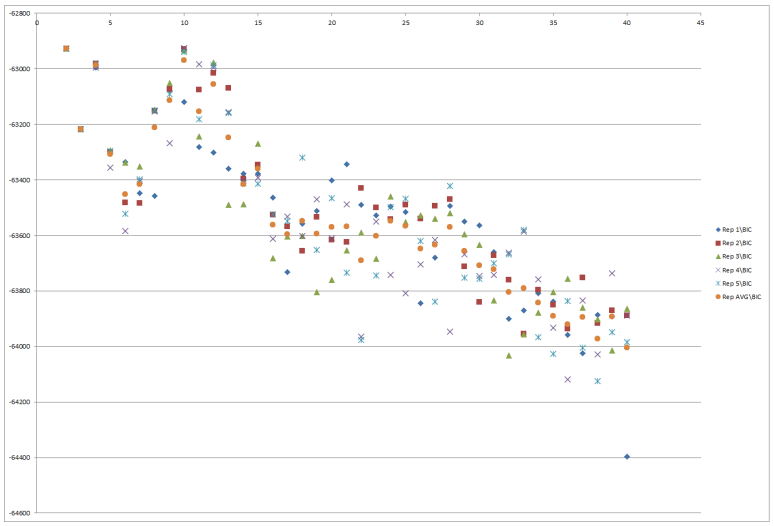
Fig. 6.14: Plots showing three different clustering performance metrics as a function of the number of clusters, k. The clustering algorithm used is k-means with the one vector per peak.



(a) Average euclidean distance from cluster centre



(b) Average silhouette width



(c) BIC

Fig. 6.15: Plots showing three different clustering performance metrics as a function of the number of clusters, k. The clustering algorithm used is k-means clustering with the one vector per experimental group per peak.

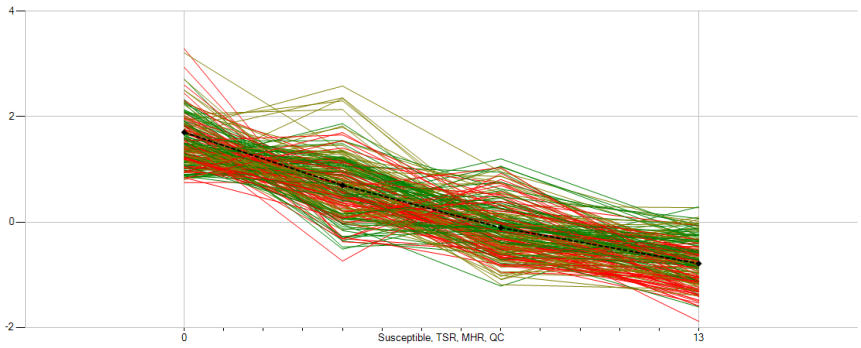


Fig. 6.16: Image showing cluster 2.

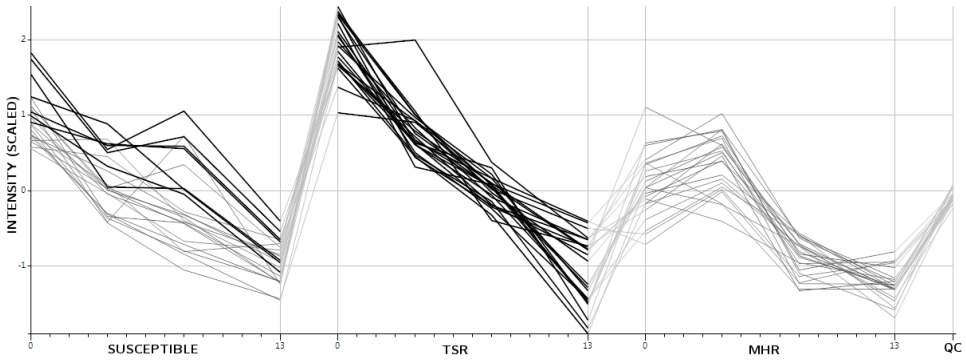
Pathway	**Compounds**	**Peaks**
compounds not assigned to any pathway	(57)	(46)
brassinosteroid biosynthesis I	(7)	(23)
simple coumarins biosynthesis	(6)	(22)
plant sterol biosynthesis II	(8)	(20)
phenylpropanoid biosynthesis	(8)	(20)
suberin biosynthesis	(7)	(20)
...	...	...

Tab. 6.3: List of pathways, in order of the number of potential peaks in cluster 2.

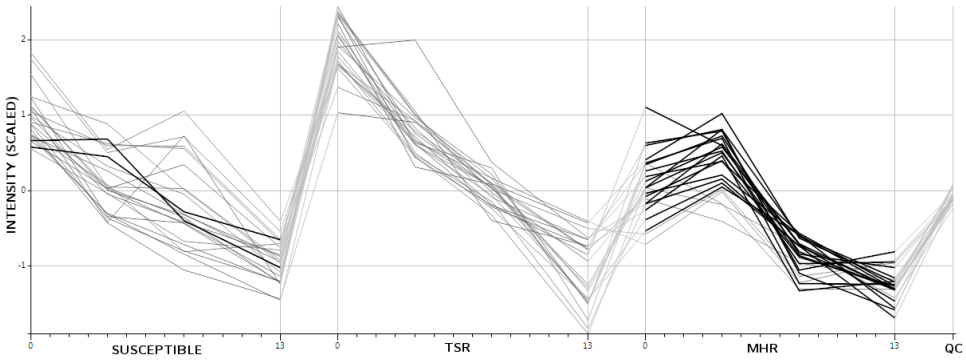
Figure 6.17 (top) shows the time profiles that potentially represent compounds in the brassinosteroid biosynthesis pathway with those that group together in cluster 2 highlighted, i.e. all profiles in the targeted herbicide resistance ($\mathcal{T}$) group together with a few from the susceptible group ($\mathcal{S}$) showing a similar trend. Figure 6.17 (bottom) shows the same time-profiles but with those that group together in a separate cluster (cluster 3) highlighted. This includes most time-profiles associated with the multiple herbicide resistance ($\mathcal{M}$) plants together with a couple from the susceptible group.

6.5 *Concluding remarks*

The easy access to statistical information and interactive visualizations in MetaboClust allow efficient and appropriate pre-processing, such as batch correction and scaling, to be performed. The software allows the effects of different pre-processing methods on data analyses to be explored. Various



(a) Cluster 2



(b) Cluster 3

Fig. 6.17: Plot of the peak intensities for peaks potentially representing compounds of the brassinosteroid biosynthesis pathway. Bold lines indicate the peaks present in cluster 2 for (a) and cluster 3 for plot (b).

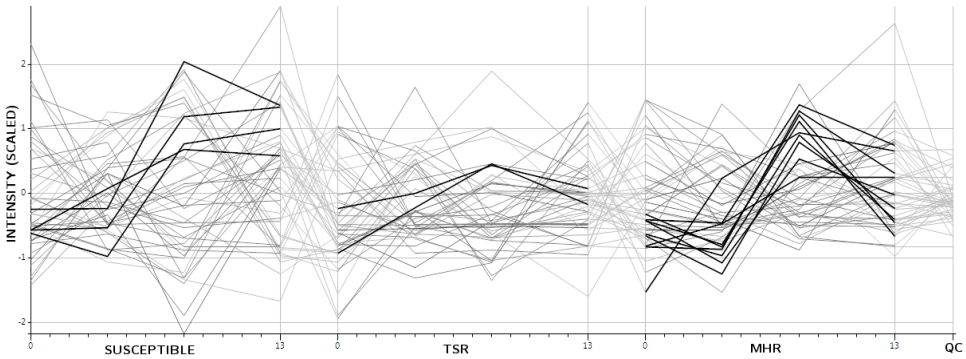


Fig. 6.18: Plot of the peak intensities for peaks potentially representing compounds of the fatty acid activation pathway. Cluster 4, highlighted in red, indicates a large number of peaks of the MHR set conform to this profile.

statistical analyses can be conducted at different stages within the software and the results easily visualised. In particular, time-profiles can be produced for cluster analysis and may be generated separately for different experimental groups or chosen to span multiple groups. The interactive user interface provides an effective means to determine profile generating and clustering methodologies.

Using the *d-k*-means++ variation in the MetaboClust software, we were able to cluster the data for two different datasets into a number of meaningful patterns (25 clusters for *Medicago* and 10 for *Alopecurus*), determined both visually and through parameter optimisation within the software. The deterministic nature of this method is particularly suited to the rapid exploration of analyses facilitated by the software. Not only were a set of potentially interesting profiles yielded, but clusters were organised in a manner indicative of the centre selection, with each cluster centre bearing the opposite profile to its predecessors. From the clusters we were able to identify differences between groups that, due to their nonlinear trends, were not identified using univariate statistics. In particular, a number of clusters displaying unique dual-stress time profiles and disease-related responses were identified in the *Medicago* data.

The pathway information sourced from external databases allows similar response profiles for which metabolite assignments are available to be linked to pathways of interest. Although the example pathways in both case studies here were implicated by tentatively assigned metabolite annotations based on accurate mass only, multiple tentative assignments were used to identify the pathways and are supported by the literature. The same method can be used with the time-profiles definitively assigned to metabolites and may also suggest where further information should be sought, for example by using standards to confirm identifications, thus saving time and money.

The software and source code are freely available for download on Bitbucket [328]. Results of the clustering, including those for the *Alopecurus* dataset, can be found on the same web-page at `https://bitbucket.org/mjr129/metabolitelevels/downloads/ThesisResults.zip`.

7. GENETIC PROGRAMMING

7.1 Introduction

The ability to make classifications based on chemometric data has a wide range of applications and studies have frequently used computerised pattern recognition methods [216, 329–331]. Several techniques are already available to classify the data, with discriminant function analysis, partial least squares, and artificial neural networks being extremely popular [331]. These methods do however lack a simple method of extracting the logic behind the classification [216, 332]. This is especially the case when large, multivariate datasets are employed and a combination of many variables influences the classification. Genetic programming (GP) has been used as a tool which can not only be used as a method to classify data, but also to identify the variables underpinning the basis on which the classifications have been made. This is especially useful with data-sets such NMR or LC-MS, where the data itself can ultimately be traced back to individual chemical species, providing a real-world target for analysis or manipulation.

7.1.1 Genetic programming

GAs are an form of evolutionary computation that seek to optimise a search heuristic by means of emulating the process of natural selection [211]. A population is created which consists of a set of individual solutions to a problem, which evolve towards a more optimal solution. The solutions or "individuals" evolve from one generation to the next following similar principals to those found in nature – through sexual reproduction, known as *crossover*, and asexual reproduction, known as *mutation*. The best-performing individuals in the population are rewarded with a higher chance of reproducing or surviving to the next generation than the worst-performing ones [215][p. 100], providing the evolutionary drive to produce "better" results.

GP is an offshoot of GA, popularised by Koza in 1992 [215, 216, 333],

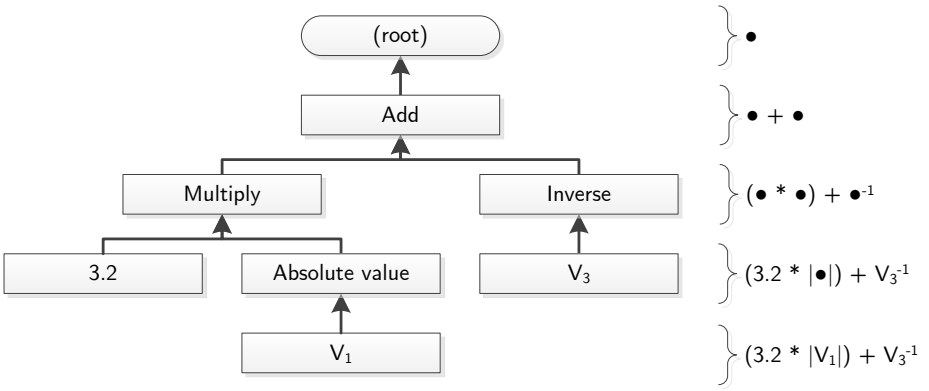
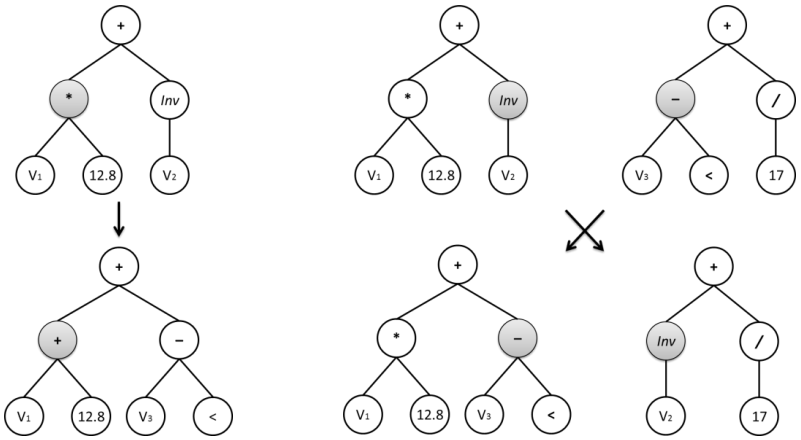


Fig. 7.1: Example of a genetic programming tree, representing the equation $(3.2 * |V_1|) + V_3^{-1}$.

although the technique had seen recognition much earlier [334], for instance in an adaptation of genetic algorithms in the manipulation of assembly code in 1975 [335] and later in the development of logical expressions in 1981 [336]. GP extends GA in that the individuals of the population are themselves computer programs and are evolved to in order to find a program that provides a good solution to the target problem. In contrast to GAs, GP represents the individuals of the population as trees, rather than linear structures. The branch-nodes of these trees represent functional operators, such as add or divide, whilst the leaf-nodes represent values, such as constants or input variables. As an example, the tree represented in Figure 7.1 can be seen to represent the equation $(3.2 * |V_1|) + V_3^{-1}$.

The process for evolving these trees proceeds similarly to that of GA. Initially a population of randomly generated individuals is created. An evaluation step then assigns a fitness value to each individual, calculated by a specified fitness function. A selection function is then used to select which individuals proceed into the next generation. There are a number of different selection methods possible but the general case is that programs with higher fitness have a higher chance of being selected. This allows the best-performing programs to have an increased chance of passing on their virtual DNA to the next generation. The selected individuals are finally used to generate the next generation through the use of one or more breeding operators, a wide variety of which are possible. [215][p. 99]

The *mutation operator*(Figure 7.2a) takes an existing individual and



(a) Mutation *(b)* Crossover

Fig. 7.2: Depiction of breeding operators in relation to GP. 7.2a: Showing mutation operator operating on a single node. 7.2b: Showing crossover operator operating on two parents and producing two children.

modifies it to produce a new individual. In the case of GP, one or more nodes are selected at random from the tree and replaced with a new, random node.

The *crossover operator* (Figure 7.2b) takes two individuals and produces two results that are each a combination of the two parents. One-point crossover generates the child individuals by swapping the parent individuals over at a single point: A node from each parent is selected at random and these nodes, including the child-nodes, are swapped over to create two children with characteristics of both parents.

Finally the *clone operator* produces an identical copy of an individual and places it directly it into the next generation. This permits the concept of elitism, whereby the best individuals of the population are maintained by always cloning a fixed number of best performers into the next generation. Similarly, a certain percentage of the worst-performing programs may always be discarded without consideration for reproduction.

A disadvantage of standard GP is that the programs generated do not necessarily reflect the abilities of a modern computer programming language. An alternative method, termed grammatical evolution (GE) has been employed to artificial genomes consisting of sequences of byte-codons, which are converted into a real programming syntax, such as Java, prior to evaluation.

This is achieved through the use of a *mapping operator*, which selects the code to insert for a particular codon based on a pre-defined set of Backus-Naur-form (BNF) syntax rules [337]. An example BNF statement describing an `if-then-else` statement in BASIC is shown here:

```
<statement> :== IF ( <condition> ) THEN ( <statement> )
                ELSE ( <statement> )
```

The text in angle brackets (`<` and `>`) denote a "class" that is defined elsewhere. Our `if` function is itself of `<statement>` class and hence one `if` may nest another. The `<condition>` class could be defined as the following:

```
<condition> :== true
<condition> :== ( <condition> OR <condition> )
<condition> :== ( <number> >= <number> )
. . .
```

The defined set of classes and their meanings defined is dependent on both the language and the problem to be solved.

In GE, the class definitions available to choose from for a particular codon of the genome is dependent upon the type of input expected by the mapping of the previous codons. An example of this is shown in Figure 7.3. This technique is advantageous in that not only can more complex bodies of code be generated, but the genomes can be manipulated much like traditional genetic algorithms, reducing computational time and increasing simplicity. Invalid syntax cannot be generated providing the BNF rules are correctly specified, for instance including the use of brackets in the definitions to avoid ambiguities.

GE has however been criticised due to poor locality [338]. Locality relates the sample space of the genotype to that of the phenotype. In the case of GE a small change in the genotype may have a dramatic impact on the phenotype – in this case both the code generated and the results of running it [339]. This is due to the fact that if a single codon is changed then for the set of cases where the BNF definition of the new function differs from that of the old one, the mapping of all subsequent elements after this codon will be changed. This problem increases the difficulty navigating the fitness landscape, since small changes produce radically different programs that are unlikely to succeed by chance alone.

A technique potentially on the border between GP and GE is Strongly-Typed Genetic Programming (ST-GP) [340]. ST-GP follows a similar pat-

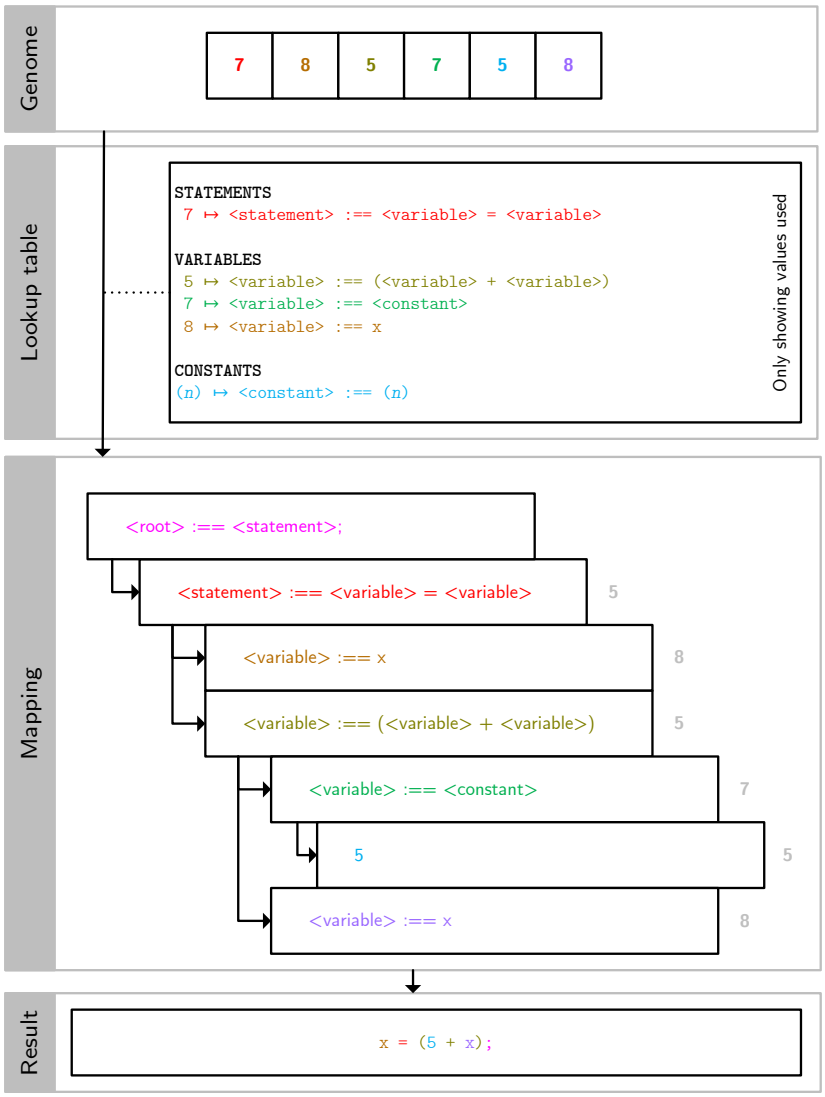


Fig. 7.3: Depiction showing the translation of an arbitrary byte-based genome through a set of BNF syntax rules into a compilable segment of code.

tern to classic GP but places restrictions upon the arrangement of the logic trees generated. Function inputs and their return values are "typed" with a class, much like GE, and the output of a function returning a value of a particular class must be paired to the input of function accepting that same class. For instance a function node taking an `integer` input must be connected to one providing an `integer` output.

In our classic-GP example shown in Figure 7.1 it can be seen that all of the nodes essentially produce the same type; they all produce a real-valued number, whether directly (3.2), from an input (V_1) or as a result of an operator (+)). Classic GP can therefore be considered a sub-case of ST-GP where there is only one class of data and nodes can be simply selected from a pre-defined list of possible definitions. ST-GP however, modifies the tree generation routine to enforce the strong-typing, "class-matching" constraint, which implies that all data are associated with a format (type), such as `integer` or `boolean`, as well as the restriction that an input of one type must be coupled to an output of the same type (strong-typed) [341]. To apply this to GP, during tree generation or modification child nodes are selected such that their output type conforms to the input type expected by the parent node. The example in Figure 7.4 shows the tree representing the ternary operator "`If`", which takes one `Boolean` input (a) and two `Real` inputs (b and c). The output type of the `If` function itself is also `Real`, being defined as b when a is true, and c when a is false. Like GE, this allows the list of potential nodes to be represented as a set of BNF syntax rules and the "tree" structure is clearly visible in Figure 7.3. Whilst this allows a much broader feature set, the benefits of standard GP are none-the-less retained: since the tree and underlying code are directly related, breeding (mutation and crossover) operators can be finely tuned.

An additional benefit of strongly typed genetic programming (ST-GP) over standard GP is the ability to perform multi-typed operations. The second goal of this chapter is therefore to explore if this is beneficial in the classification of our beef samples. Specifically we shall investigate whether class – of factorial type – and age – of numeric type – can be determined concurrently via an ST-GP algorithm and if this presents any benefit to the classification.

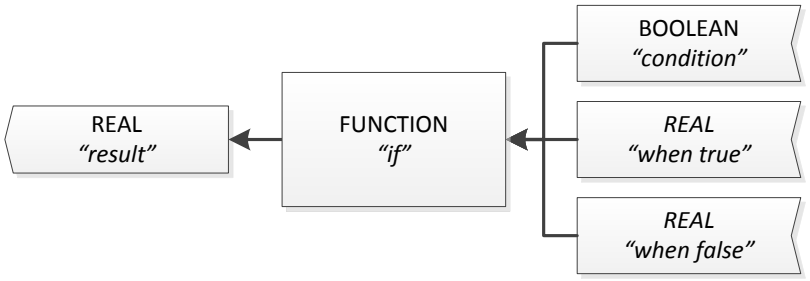


Fig. 7.4: Strong-typed `If` statement tree showing input types expected (bottom) and output type produced (top).

7.2 Method

7.2.1 Data

This paper uses the *beef* dataset previously described in Section 3.3. The dataset is truncated to only include the 119 samples from *warm* $\mathcal{W}$ and the 120 samples from the *cold* $\mathcal{C}$ groups.

7.2.2 Programming

The ST-GP framework was implemented in-house using C#. The implementation (named *Evove*), uses the .NET reflection library, which allows programs to inspect their structure and modify their data at runtime. This has the advantage over GP frameworks reliant on a text interpretation in that reflection allows programs to be generated from the GP trees with relatively little overhead in terms of computational expense. The use of reflection also allows the functions to be automatically located and parsed by the runtime, performing the majority of the construction for the GP algorithm without the need for extensive user setup. The suite of functions provided to the tree-nodes, as well as the fitness function, can be written in any language supporting compilation to CIL, including C++ and Java.

7.2.3 Breeding operators

7.2.3.1 Mutation

In [342] the authors note that ST-GP was less successful than applying standard GP to the inputs of pre-defined Java functions for the purpose of generating competitive robot controllers. However, what may be considered

a particularly harsh mutation operator is used in this case, whereby a node to be mutated is selected at random and both the node and its child nodes are regenerated. This is likely to have a significant impact on locality since, much like GE, small changes to one part of the program can have dramatic effects upon the rest – a change to the root node for instance, would replace the entire tree. For the purposes of our study we shall use instead a "classic" GA mutation operator, whereby each node is mutated with a fixed probability p_{node} and the child-nodes are not modified unless enforced by the strong-typing constraints. Whilst functional changes, such as the substitution of a multiplication operator with a subtraction operator, can have a non-trivial impact this mutation operator should be less intensive than whole-tree replacement. This method does have the disadvantage however, that the dynamic size and layout of GP trees contrasts with a linear GA setup and an additional set of rules must be employed to deal with the constraints enforced by ST-GP.

Here we allow a mutated node to be altered via one of four methods: reordering, insertion and deletion, as well as standard replacement. The exact mutation method used is selected at random, with all methods sharing an equal chance of being selected. ST-GP constraints are maintained by randomly regenerating child-nodes only when no alternative is possible: for instance a change of input type from `integer` to `text` would necessitate a regeneration of child nodes, whereas a change from `integer` to `integer` would not.

The four mutation operators are thus:

- Iterate each node N_A in the parent tree A
- Mutate N_A with a random chance p_n (if not mutated then go to the next node)
- Select a random method of mutation from the following:

  Replacement: Replace N_A with a new, randomly generated node N_B

  Retain child nodes N_A as child nodes of N_B *where* the child-node types match the expected inputs of N_B.

  Discard all other child nodes of N_B

  Randomly generate any additional required child nodes of N_B

Reordering: Swap N_A with a sibling node N_B *where* $\text{type of}(N_A) = \text{type of}(N_B)$

Insertion Insert a random node N_B above N_A *where* $\text{type of}(N_A) = \text{type of}(N_B) \wedge \text{type of}(N_A) \in \text{type of input}(N_B)$

Deletion Replace N_A with a random child of N_A, N_B *where* $\text{type of}(N_A) = \text{type of}(N_B)$

7.2.3.2 Crossover

We use the crossover operator noted in [340], with the added *common point* constraint to ensure tree size does not exceed our set maximum.

- Select a random node N_A in parent tree T_A
- Select a random node N_B in parent tree T_B

  where $\text{type of}(N_A) = \text{type of}(N_B)$ (strong typing constraint)

  $\wedge \text{depth of}(N_A) = \text{depth of}(N_B)$ (common point constraint)
- Swap nodes N_A and N_B.

7.2.4 Parameter optimisation

There are a large number of parameters available for modification in a typical GP setup and testing every combination would be an impossible task. The figure shown in 7.5 presents the complete set of parameters available for manipulation in a typical ST-GP simulation. Due to the large number the entire set cannot be optimised on a reasonable time-scale and therefore most were fixed to pre-selected values, with only the most significant parameters, mutation and crossover rate, being optimised here. Excluding mutation and crossover, the remainder of the parameters are detailed in Table 7.1.

Testing 10 values for 2 parameters would result in a $10^2 = 100$ value test matrix. Sampling methods can be used to determine an adequate coverage of the parameter space given a fixed number of possible samples. In our study we made make use of the "orthogonal sampling" method. Orthogonal sampling is based on Latin Hypercube sampling, which splits the sample space into a $n \times n$ square grid by partitioning each of the variables to be measured into n segments. A set of samples of the parameter space is then taken such that each segment, for each variable, is sampled once [343].

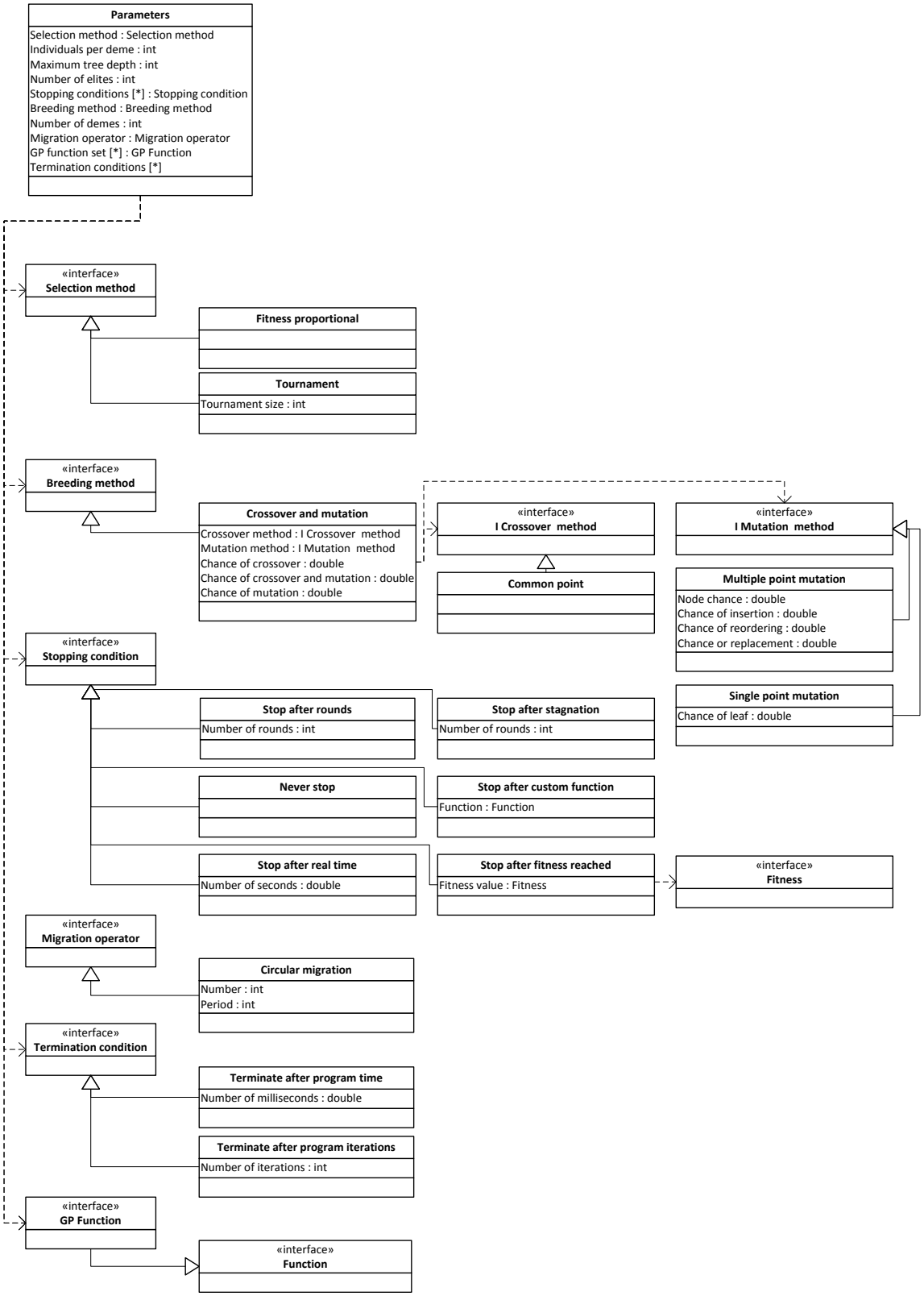


Fig. 7.5: UML diagram showing the parameters available for modification in a typical ST-GP simulation, along with the dependencies of parameters upon each other.

Parameter	Value
Number of individuals	200. We used the value given in [217] for a similar task.
Selection method	Fitness proportional.
Stopping conditions	Varied
Fitness function	Varied
Maximum evaluation time	1 second
Randomisation method	Knuth variant [345, 346]
Breeding operators	Crossover, mutation, crossover and mutation, elitism
Breeding constraints	Strong typing, maximum tree depth
Maximum tree depth	5 (about 25 nodes)
Number of elites	1
Crossover operator	Common point, strong-typed
Crossover chance	(Value optimised)
Mutation operator	Per-node, strong-typed
Mutation chance (individual)	(Value optimised)
Mutation and crossover chance	Crossover chance * Mutation chance
Mutation of node chance	0.05
Mutation of node methods	insert, reorder, replace (equal probabilities)
ST-GP Data types	Double (rational), Variable index (integral), Factorial (boolean), AgeAndClass (tuple of factorial and double)
ST-GP Function set	See appendix D

Tab. 7.1: Table of ST-GP parameters and their assigned values in our study.

Orthogonal sampling extends this by dividing the sample space again into a set of m subspaces (where $m < n$), ensuring that each subspace is sampled at an equal frequency. This offers an additional safeguard against areas of the sample space being sparsely tested.

We used the orthogonal sampling method to devise a set of samples to test the mutation × crossover sampling space, given a limit of 100 samples. The set of samples was generated using the *orthogonal sample* script for Matlab, published at [344]. Performance was measured as the fitness of the class predicate (outlined below) with a stopping condition of `rounds = 200` , averaged over 1 hour of repeated processing.

A Voronoi diagram of the parameter optimisation results is shown in

Figure 7.6. The classification *as a whole* can generally be considered to be an easy problem, with even the worst performing configuration obtaining just under 85% accuracy in the classification of the validation set. The best configuration correctly predicted 93.9% of the training set and 91.7% of the validation set, generally suggesting that overfitting is not a substantial problem. Using the best performer in the *training* set in order to avoid bias, these results suggest the optimal parameters are 0.93 for the mutation rate and 0.3 for the crossover rate. Whilst the crossover rate suggested is somewhat low, it is apparent in the figure that higher crossover rates lead to a general reduction of performance. These results complement an existing extensive study into mutation and crossover, which suggests that crossover acts merely as an extended form of the mutation operator, with mutation giving the highest increase in performance [347] .

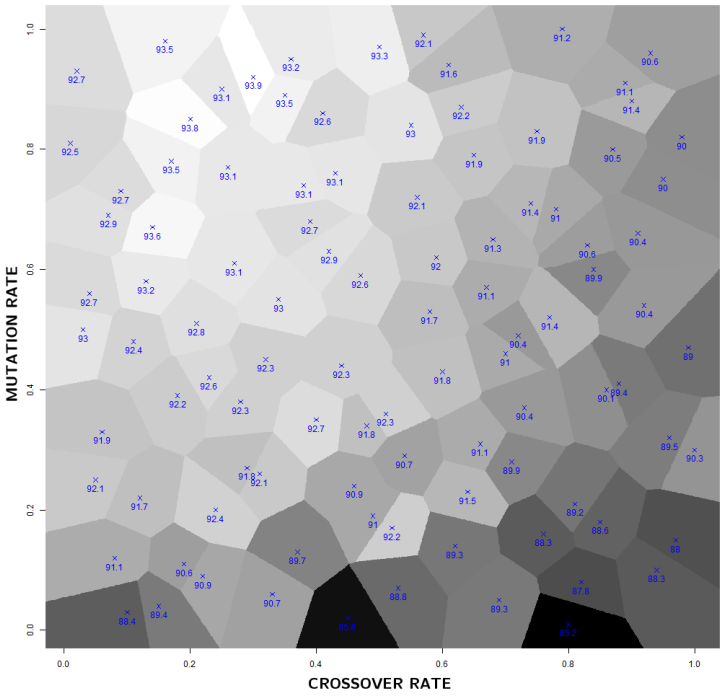
7.2.5 Fitness functions

Two varieties of the fitness function were tested: the class and age predictor and the class predicate. The function listings are presented as BNF definitions for C++, due to the language's wide support. The complete function listing is given in Appendix D and details the complete list of functions available as nodes to the GP algorithm.

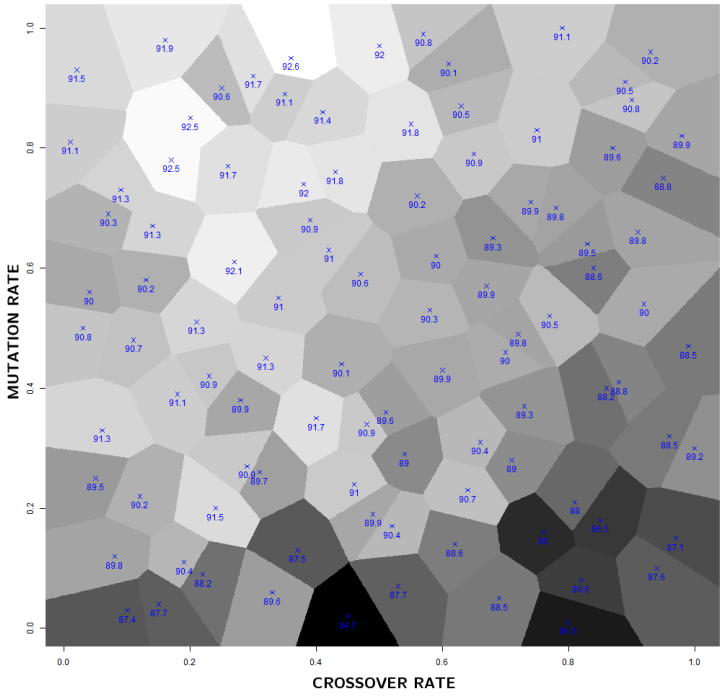
7.2.5.1 Class and age predictor

The class and age predictor (CA) seeks to predict both the class and age of the observations. For the purposes of ST-GP age is defined as `double`. The class is considered a boolean predicate where $I = 1$ and $A = 0$, which, for the purposes of ST-GP is considered as `boolean` type. The combined result of the prediction – the age *and* class object – is defined as being of `age_and_class` type and comprises a tuple of age (`double`) and class (`boolean`), as outlined in Figure 7.7.

The fitness function maps to the number of observations assigned into the correct class, less a penalty based on the number of days out the age prediction lies. Incorrect class predictions are a score of zero, per observa-



(a) Training



(b) Validation

Fig. 7.6: Voronoi diagrams depicting the performance of the parameter values tested. The crosses indicate the values tested, with the numbers below indicating the dataset indicated below the image. The cells are shaded according to the fitness values, with lighter colours indicate better performance (higher fitness). It can be seen that the performance of the training and validation sets are similar, and both present relatively smooth fitness landscape with optimal results towards the high-mutation, low-medium crossover values.

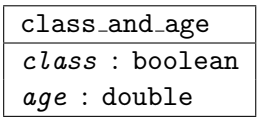


Fig. 7.7: UML diagram depicting the layout of the `class_and_age` structure.

tion.

$$f_{CA} : \{A, P\} \mapsto \left\langle argmin\left(A_i^C = P_i^C, \frac{|A_i^A - P_i^A|}{max_{j=1}^{n} A_j^A}\right)\right\rangle_{i=1}^{n} \tag{7.1}$$

Here A_i^A and P_i^A represent the actual and predicted ages of the ith element in each vector while A_i^C and P_i^C represent their classes. The class equality operator ($=$) is defined as 1 for when the classes match, and 0 otherwise (i.e. $a = b \mapsto 1 - |a - b|$). The angle brackets ($\langle\rangle_{i=1}^{n}$) denote the mean average of the set of predictions $1..n$ (i.e. $\langle X_i\rangle_{i=1}^{n} \mapsto \sum(X_i)_{i=1}^{n}/n$).

The root function, `root_ca`, maps each observation to $\mapsto P_i^C$ and is defined:

```
<root_ca> = <class_and_age>
```

Since the `class_and_age` structure is unique to the prediction the following functions are provided to the GP to yield and use this structure:

```
<class_and_age> ::= ( predictedClass = <double>,
                    class_and_age( predictedClass,
                                   <double> ) )
<class_and_age> ::= ( predictedAge = <double>,
                   class_and_age( <bool>,
                                  predictedAge) )
<bool> ::= predictedClass
<double> ::= predictedAge
```

The functions yielding `class_and_age` determine whether age or class is predicted first whilst the functions yielding `boolean` and `double` are available to retrieve the predicted class or age for an observation once it has been predicted, during the prediction of the other. Calling the retrieval functions before the value is known returns `0`.

7.2.5.2 Class predicate

The class predicate (CR) seeks to predict the class for each of the test cases. The fitness function is simply defined as the accuracy of the prediction or the fraction of matching classes, f_C:

$$f_C R : \{A, P\} \mapsto \left\langle A_i^C = P_i^C \right\rangle_{i=1}^{n} \tag{7.2}$$

To minimise the differences between the two predictors, excluding the fitness function itself, all other features are identical.

7.3 Results and discussion

7.3.1 Prediction method

The mean classification rate, against generation, is shown in Figure 7.9. It can be seen that the average training classification rate is noticeably higher for the CR method than for the CA method. This may be unexpected given that the CR fitness function is the classification rate, whilst the CA fitness function also incorporates age into its fitness function. However, the average accuracy of the validation set is almost identical for the two techniques, averaging 90% for CR and 89% for CA. As ST-GP methods are inherently stochastic, the Vargha Delaney A statistic provides a measurement of the frequency with which one method will outperform another [348]. For our ST-GP runs using the CR and CA methods, the A statistic indicates that in 56% of the cases CR will outperform CA in terms of classification accuracy.

Figure 7.10 provides insight into which observations are being incorrectly classified by the two techniques, showing the average classification accuracy of the validation set for each observation, over the generations, for each ST-GP method. Regions of incorrect classification appear as dark bands in the plot. The overlapping set of observations noted in Section 3.3 is clearly visible as two dark bands across the generational axis. Whilst getting "thinner" as time goes due to the algorithm correctly predicting more the observations at the edges of the bands, the ability to predict 6 observations located in the band centres falls to 0% accuracy. These observations correspond to 4 replicates of warm-storage day 1 and 2 replicates of warm-storage day 5. A confusion matrix considering the observations predicted incorrectly more than 50% of the time is shown in Table 7.3, alongside the results from

the PLSR-LDA analysis presented earlier in Section 3.3. From this is can be seen that both the CA and CR ST-GP methods achieve similar results, with the late cold-storage (age = 27) observations in both cases being incorrectly classified as warm-stored. PLSR-LDA meanwhile has a greater tenancy to predict the early warm (age = 5) as cold. Interestingly the CA classifier, having also predicted age, shows that rather than having incorrectly predicted the late cold samples as early warm ones, the samples are instead predicted to be mid-range to late warm. The unexpected accuracy of the age prediction may indicate age biomarkers independent of class. However, these predictions may be an artefact from an earlier stage of evolution, since, according to the CA fitness function, there is no benefit to the classifier predicting age if it is unable to predict class. This second possibility however, would require retention of unhelpful parts of the ST-GP-genome. Another possibility is that the classifier is using the age to predict the class. The function of the of the best predictor, however, shown in Figure 7.8, indicates that the most effective algorithm favoured a class-first prediction.

7.4 Conclusion

Whilst the concurrent prediction of age with class was not found to confer any additional benefit to the determination of storage conditions for the beef data, ST-GP did allow a reasonably accurate determination of age with only minor reduction (1%) in class prediction of the validation set. Whether a reduction of the fitness-function penalty exacted for age miscalculation could remedy this discrepancy warrants attention in future work. Perhaps the most beneficial factor of performing concurrent age prediction is in the insight given into the *incorrect* predictions. It has been seen that, rather than the incorrect predictions having been based upon the same set of "overlapping" samples as PLSR-LDA (early-warm), the ST-GP predictor confused late-cold with mid-to-late-warm.

Additionally, the use of GP allowed a greater number of the samples to be classified correctly than PLSR-LDA with the additional benefit of selecting a small number of variables with which the classification could be performed. In addition to having a higher percentage accuracy, ST-GP was found to predict class correctly for a different set of observations than PLSR-LDA, suggesting that complementary methods could potentially be of use.

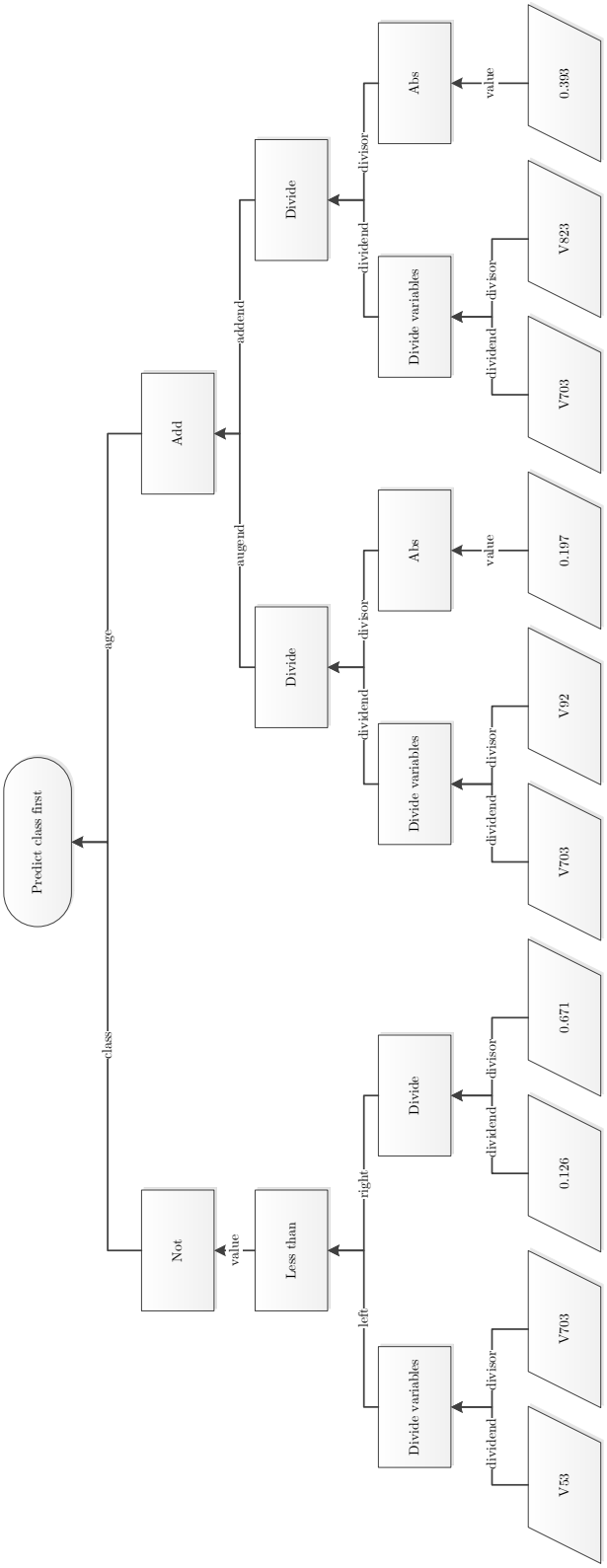


Fig. 7.8: GP tree with the highest fitness value in validation for the CA fitness function.

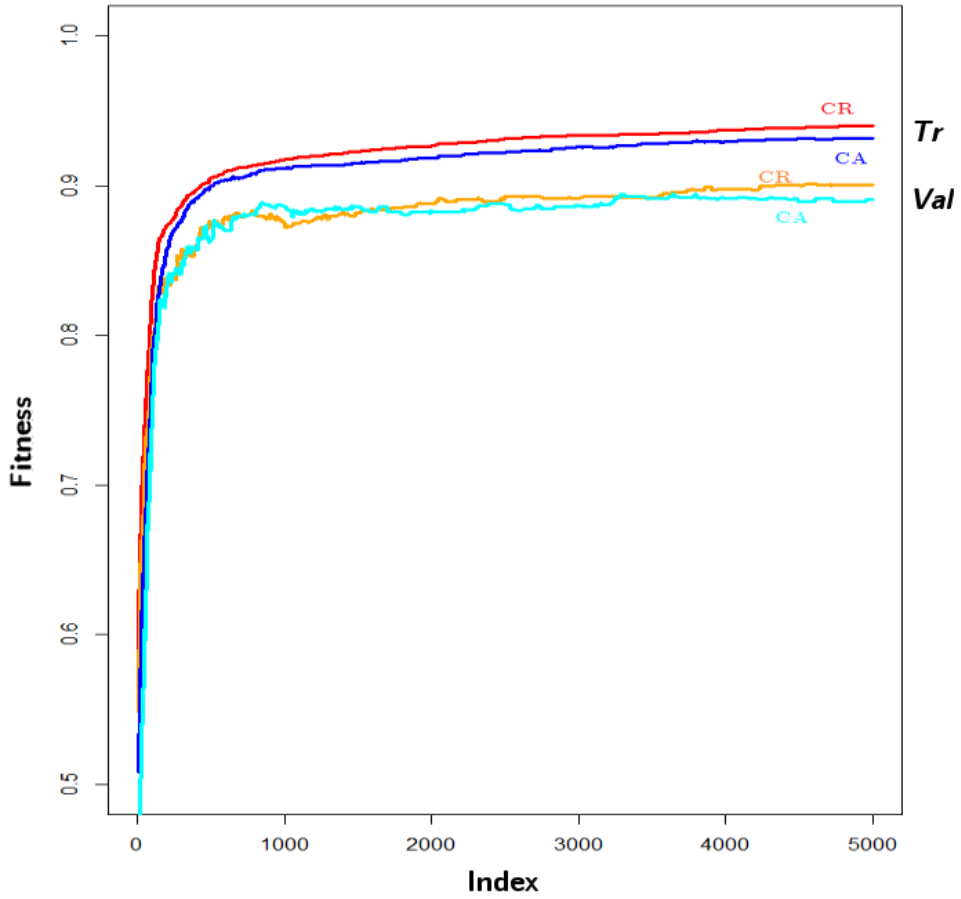


Fig. 7.9: Comparison of class predictive accuracy between the CA and CR methods for training (*Tr*) and validation (*Val*) data. The CR method provides higher predictive accuracy than the CA method, with a difference of around 1%.

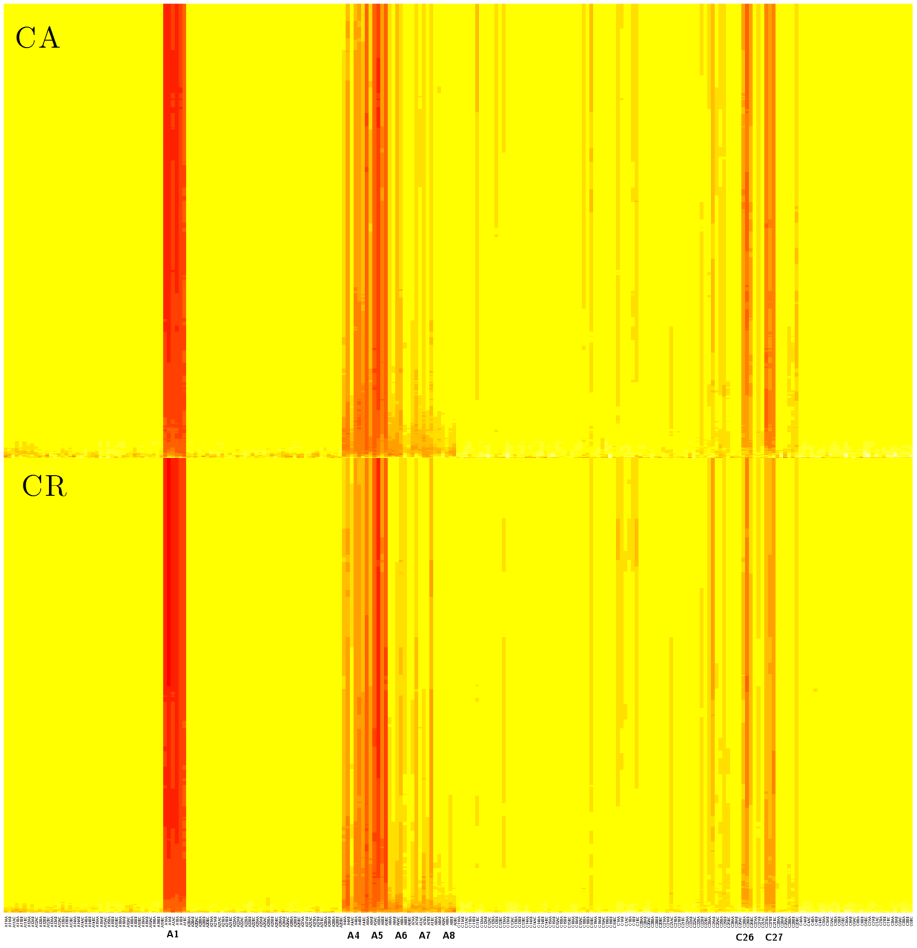


Fig. 7.10: Comparison of predictive accuracy of individual observations between the CA and CR methods. Brighter regions (yellow) indicate higher average predictive accuracy than darker (red) regions. Several "bands" of low predictive accuracy can be seen. The corresponding experimental groups for these regions are noted along the X-axis text.

Actual		Predicted				
Age and class	ID	Age and class				
$\mathcal{C}1$	C1aa	$\mathcal{W}13$	$\mathcal{C}$-1	$\mathcal{W}2$	$\mathcal{W}18$	$\mathcal{W}4$
	C1ac	$\mathcal{W}13$	$\mathcal{C}0$	$\mathcal{W}4$	$\mathcal{C}18$	$\mathcal{W}4$
	C1ba	$\mathcal{W}13$	$\mathcal{C}1$	$\mathcal{W}4$	$\mathcal{W}18$	$\mathcal{W}5$
$\mathcal{C}25$	C25ab	$\mathcal{W}24$	$\mathcal{W}10$	$\mathcal{W}19$	$\mathcal{C}17$	$\mathcal{W}18$
$\mathcal{C}26$	C26ba	$\mathcal{W}15$	$\mathcal{W}11$	$\mathcal{C}17$	$\mathcal{W}17$	$\mathcal{W}15$
	C26bb	$\mathcal{W}17$	$\mathcal{W}18$	$\mathcal{W}13$	$\mathcal{W}12$	$\mathcal{W}17$
$\mathcal{C}27$	C27ba	$\mathcal{W}8$	$\mathcal{W}12$	$\mathcal{W}18$	$\mathcal{W}16$	$\mathcal{C}14$
	C27bb	$\mathcal{W}21$	$\mathcal{C}15$	$\mathcal{W}19$	$\mathcal{C}17$	$\mathcal{W}21$
$\mathcal{C}28$	C28ba	$\mathcal{W}24$	$\mathcal{C}31$	$\mathcal{C}20$	$\mathcal{W}27$	$\mathcal{W}23$
$\mathcal{W}1$	W1ab	$\mathcal{C}6$	$\mathcal{C}4$	$\mathcal{C}5$	$\mathcal{C}8$	$\mathcal{C}15$
	W1ac	$\mathcal{C}6$	$\mathcal{C}5$	$\mathcal{C}7$	$\mathcal{C}4$	$\mathcal{C}5$
	W1ba	$\mathcal{W}2$	$\mathcal{C}2$	$\mathcal{W}0$	$\mathcal{C}8$	$\mathcal{C}3$
	W1bb	$\mathcal{C}28$	$\mathcal{W}$-4	$\mathcal{C}21$	$\mathcal{C}24$	$\mathcal{C}9$
	W1bc	$\mathcal{C}15$	$\mathcal{C}4$	$\mathcal{C}2$	$\mathcal{C}3$	$\mathcal{C}3$
$\mathcal{W}4$	W4ab	$\mathcal{C}13$	$\mathcal{C}8$	$\mathcal{W}7$	$\mathcal{W}18$	$\mathcal{C}9$
	W4ba	$\mathcal{C}8$	$\mathcal{C}18$	$\mathcal{C}9$	$\mathcal{W}8$	$\mathcal{W}7$
	W4bc	$\mathcal{C}27$	$\mathcal{W}0$	$\mathcal{W}21$	$\mathcal{C}23$	$\mathcal{C}12$
$\mathcal{W}5$	W5aa	$\mathcal{C}11$	$\mathcal{C}12$	$\mathcal{C}13$	$\mathcal{C}11$	$\mathcal{C}9$
	W5ab	$\mathcal{C}27$	$\mathcal{W}15$	$\mathcal{C}21$	$\mathcal{C}24$	$\mathcal{C}12$
	W5ac	$\mathcal{C}11$	$\mathcal{C}17$	$\mathcal{C}10$	$\mathcal{C}16$	$\mathcal{W}9$
	W5ba	$\mathcal{C}27$	$\mathcal{W}13$	$\mathcal{C}21$	$\mathcal{W}23$	$\mathcal{C}10$
	W5bc	$\mathcal{C}13$	$\mathcal{W}12$	$\mathcal{C}10$	$\mathcal{C}18$	$\mathcal{W}14$
$\mathcal{W}6$	W6ba	$\mathcal{C}13$	$\mathcal{W}10$	$\mathcal{C}13$	$\mathcal{C}12$	$\mathcal{C}11$
$\mathcal{W}7$	W7ab	$\mathcal{C}28$	$\mathcal{W}10$	$\mathcal{C}21$	$\mathcal{C}24$	$\mathcal{W}6$

Tab. 7.2: Table of actual and predicted values for the CA predictions (shown only for cases incorrect in more than 50% of the runs). The second column shows the unique sample ID.

ID	Class	Age	CA	CR	PLSR-LDA	Total incorrect
W1ab	$\mathcal{W}$	1	×	×	×	3
W1ac	$\mathcal{W}$	1	×	×	×	3
W1ba	$\mathcal{W}$	1	×	×	×	3
W1bb	$\mathcal{W}$	1	×	×	×	3
W1bc	$\mathcal{W}$	1	×	×	×	3
W1aa	$\mathcal{W}$	1	·	×	×	2
W4ab	$\mathcal{W}$	4	×	·	×	2
W4ba	$\mathcal{W}$	4	×	×	×	3
W4bc	$\mathcal{W}$	4	×	·	×	2
W4aa	$\mathcal{W}$	4	·	×	·	1
W4bb	$\mathcal{W}$	4	·	×	×	2
W5aa	$\mathcal{W}$	5	×	×	×	3
W5ab	$\mathcal{W}$	5	×	·	×	2
W5ac	$\mathcal{W}$	5	×	×	×	3
W5ba	$\mathcal{W}$	5	×	×	×	3
W5bc	$\mathcal{W}$	5	×	×	×	3
W5bb	$\mathcal{W}$	5	·	×	×	2
W6ba	$\mathcal{W}$	6	×	·	×	2
W6aa	$\mathcal{W}$	6	·	·	×	1
W6ab	$\mathcal{W}$	6	·	·	×	1
W6ac	$\mathcal{W}$	6	·	·	×	1
W7ab	$\mathcal{W}$	7	×	·	×	2
W7bb	$\mathcal{W}$	7	·	·	×	1
W7ba	$\mathcal{W}$	7	·	·	×	1
W7bc	$\mathcal{W}$	7	·	·	×	1
W25ba	$\mathcal{W}$	25	·	·	×	1
C1aa	$\mathcal{C}$	1	×	·	×	2
C1ac	$\mathcal{C}$	1	×	·	×	2
C1ba	$\mathcal{C}$	1	×	·	·	1
C1bb	$\mathcal{C}$	1	·	×	·	1
C1ab	$\mathcal{C}$	1	·	·	×	1
C22bc	$\mathcal{C}$	22	·	·	×	1
C22bb	$\mathcal{C}$	22	·	·	×	1
C25ab	$\mathcal{C}$	25	×	·	·	1
C26ba	$\mathcal{C}$	26	×	×	·	2
C26bb	$\mathcal{C}$	26	×	·	×	2
C27ba	$\mathcal{C}$	27	×	×	·	2
C27bb	$\mathcal{C}$	27	×	×	·	2
C27ab	$\mathcal{C}$	27	·	×	·	1
C27bc	$\mathcal{C}$	27	·	×	×	2
C28ba	$\mathcal{C}$	28	×	·	·	1
Total			24	20	32	

Tab. 7.3: Table of predictions for the CA, CR and PLSR-LDA predictors. For CA and CR predictions are taken as incorrect when over 50% of the trained programs failed to obtain the correct class classification in the validation set. Only observations predicted incorrectly for at least one of the predictors are listed. Incorrect predictions are denoted with a cross.

The necessity of reporting the random number generator used in GP studies has been previously reported [349]. The random number generator used in this study is a variant of Donald Knuth's original algorithm[345] presented in [350, p. 283]. This offers a reasonable compromise between computational speed and randomness. However, whilst still effective, this variant is known to contain an error, not known at the time of writing, potentially resulting in a lower random period than intended [351]. Given the nature of GP this is unlikely to have a significant effect, as the primary concern is that the search space is adequately covered, and the same generator was used in all our tests, however further work would be required to confirm this.

Perhaps the biggest drawback to ST-GP is the computational time required to execute the algorithm. As a stochastic method multiple runs are required, taking over 1 day on a consumer laptop (Intel i3-2350M, 8Gb RAM). In our particular case parallel execution performed more slowly than serial execution, likely due to the need to marshal data at the processing intersections, however by increasing population size faster convergence may be accomplished, justifying the use of multi-core evaluation.

8. CONCLUSIONS

This thesis has explored the use of chemometric analysis in the understanding of metabolomic data, identified problems and proposed solutions. This chapter summarises the overall contributions, discusses their limitations, and presents avenues for future research.

8.1 Individual contributions

In the field of metabolomics, new technologies and increased computational power have provided both the need for, and means to develop, statistical methods for the interpretation of large and complex datasets. In Chapter 2 an up to date review of the current computational technologies available to chemometric analysis of metabolomic data was presented, with particular attention to two of the most common and complementary techniques in analytical chemistry, NMR and LC-MS.

As noted in the introduction[1], the agricultural industry has many challenges. Several of these are exemplified with the datasets explored in this thesis, including yield loss due to biotic and abiotic stresses (*Medicago* dataset), herbicide resistance (*Alopecurus* dataset) and product misrepresentation (Beef dataset). Chapter 3 presented an analysis of these data using several of the more common chemometric tools described in Chapter 2.

Although both the *Medicago* and *Alopercurus* analyses yielded a number of potential biomarkers, summarised in appendix A, some limitations were encountered and discussed. Analysis of any experimental data can be hindered by noise. In LC-MS-based metabolomic studies noise due to batch differences is often present, usually as a major source of variance. This was demonstrated to be the case for the *Medicago* dataset. Whilst traditional batch-effect reduction methods rely upon the use of quality control samples, Chapter 4 shows that these methods can, in certain circumstances, pro-

[1] Chapter 1

duce sub-optimal results or even exacerbate existing batch differences. A novel, robust method of batch correction termed "background correction" was therefore presented, using a profile generated from the samples in order of LC-MS acquisition. In comparison to standard QC correction, this method provided quantitatively good results in terms of the RSDs of replicate samples. Furthermore, the PCA-MANOVA scores showed that differences between experimental groups were more pronounced post-correction than for QC-based techniques, which is beneficial for further analyses. Correction of instrumental drift is ongoing research issue in the field of LC-MS analysis. Recently, Wehrens et al. have applied correction methods, also using the full set of experimental samples, to three datasets. These methods have again shown to perform adequately in comparison to QC samples on three datasets [352], with the authors advocating a possibility of reducing the number of QC samples used in the analysis. It should be noted however, that both in this thesis and the paper by Wehrens et al., the datasets contained replicate samples. Since replicate samples (including the QCs themselves) may afford greater accuracy to full-data correction methods, the accuracy of the method on datasets without replicates has yet to be determined and, due to the use of replicates in determining the accuracy itself, would likely prove a difficult task.

Post-batch-correction, the analysis of the stress datasets (*Medicago* and *Alopecurus*), revealed a number of potential stress-responsive biomarkers. However, both univariate and multivariate analyses produced results that were difficult to interpret. In both cases, this was due to the large number of resultant peaks providing no succinct set of targets for future analysis. This was further complicated by the absence of a firm cut-off point above which peaks could be classified as responsive to the experimental conditions. In light of this, focus to the *Medicago* data was given in Chapter 5, which introduced clustering as a means of dealing with a large number of variables in an unbiased manner.

Clustering of metabolomic data was hindered by the presence of age-related changes. The added combination of age-related metabolite time-profiles made it difficult to identify the time-profiles mediated by the experimental conditions. Generation of a smoothed profile of the metabolite intensities for the control group over time was shown to permit a control-relative intensity matrix to be formed. The smoothing method avoided the

transfer of excessive noise from the control samples into the other experimental groups. Unlike a simple subtraction of trends this method permits use of the full intensity matrix in future calculations. Generation of the input vectors for clustering from a similar trend profile, calculated across each experimental group, was demonstrated to be effective in order to both increase the accuracy of the data by combining replicates and removed the need to compare unpaired samples.

Metabolomic data is typically scaled to allow low-concentration but biologically significant compounds to be detected. Even if data is not scaled, distance metrics such as the Pearson Distance remain insensitive to scale and produce the same results for scaled and unscaled data. The presence of "flat profiles" therefore, can interfere with the clustering algorithm, since in scale-free data, these become indistinguishable from noise. The use of the full intensity-matrix obtained from the control-correction stage provided a reference "flat" profile in the form of the control-corrected control group samples. This profile, combined with a simple t-test was shown to be an effective method of identifying these flat profiles. Discounting the identified profiles from the clustering algorithm provided a more useful set of resulting clusters.

Finally, it was suggested that the identification of multiple "tentatively identified" metabolites within a cluster could be used to build up evidence of pathway elicitation. Using this method several pathways were identified from the data, which were supported by the existing literature.

One of the time consuming stages in analysis of metabolomic data such as these is that the generation of the clusters from time-course profiles is not a one step process. Notably, the set of parameters across the full pipeline is not known up front to the researcher. A great deal of time is therefore lost if it becomes apparent that the set of methods and parameters previously used are not amenable to the current stage of analysis.

Chapter 6 suggests a dynamic software-driven workflow for the analysis of metabolomic data. This is demonstrated with a working computational analysis suite in which the effects of changes to parameters at one stage of analysis can be visualised and their effects immediately determined further down the line.

During visual analysis, the use of k-means clustering was found to either produce qualitatively poor results, or took too long to optimise to facilitate

rapid data exploration. As part of the software-mediated workflow, a modification of the probabilistic k-means++ algorithm was therefore developed to select the most probable configuration. This gave results comparable to optimised k-means, but without the computational overhead, making it well suited to rapid visual exploration.

The initial analysis of the beef data indicated that beef stored in cool (ideal) and warm (ambient) conditions presented a different set of metabolite concentrations at different points in time. However, the presence of a "crossover" point was noted. At this point short-aged beef, kept in ambient conditions, could emulate "28 day matured" beef stored under ideal conditions, with both sets of samples exhibiting highly similar chemical profiles. Since this presents a window for product mislabelling, attempts were made to determine whether the storage conditions of any given sample could be identified. PLS-DA analysis gave reasonable classification results, though however failed to determine the storage conditions of samples within a small "early-ambient late-ideal" window. Furthermore, PLS-DA did not yield a useful and concise set of metabolites responsible for the separation of experimental groups, for instance, amenable to a rapid field test.

Strongly-typed-genetic-programming has been recently used in financial trading [353] with good results, and, in the field of bioninformatics, to the discovery of DNA motifs [354]. At the time of writing however, there are no known applications of ST-GP in the field of metabolomics. Chapter 7 presented the use of strongly-typed-genetic-programming as a method of overcoming the shortcomings of standard algorithms such as PLS-DA, noted above. Here, the storage conditions of beef samples were determined concurrently with the time for which they had been stored. The results of this analysis suggested that there still remained a time-frame in which storage conditions were indistinguishable, albeit with better results than PLS-DA. The use of ST-GP was however able to yield additional information, predicting sample age concurrently with storage condition, whilst providing a concise set of metabolites in the results.

The background correction method performed well on the *Medicago* data. However additional work needs to be performed in order to assess its generalisability. Whilst the algorithm does not consider QC samples to have any special value, they were present in the data at regular intervals and may have assisted in the generation of the trend. Hence, this study cannot con-

clude that QC samples are not required. Any study comparing correction with and without QC samples, nonetheless needs to address a variety of datasets in order to separate QC-based differences from differences between individually recorded data.

8.2 Future work

The generation of suitable clustering vectors requires good data. In the case of the *Medicago* dataset, 3 replicates for 1 day intervals over a 12 day period yielded a reasonable trend over time. However, for longer sampling intervals, the purpose of the trend-generating function is not so clear. If the interval is too long, the relationship between adjacent samples will not be present. In these cases reliance on other methods, such as replicate averaging and more complex outlier detection, would therefore be required. Additionally, while the presence of a control-group was able to yield a control-relative intensity matrix for *Medicago*, in the case of the *Alopecurus* data control correction was not performed. Per-group input vectors were therefore used and found to be effective in keeping the number of clusters manageable.

The accessibility of tools and algorithms to the experimental end-user has become an increasing focus in bioinformatics [355, 356] and a number of software packages which include a user-friendly interface have been developed to assist with data analysis in recent years (e.g. [124, 357, 358]). In terms of the means of analysis, however, GUIs have been criticised in the field of metabolomics for ultimately taking longer than script-based tools, once users become familiar with the software [359]. However, they can assist in the analysis of unknown data where a more exploratory approach is required. It thus stands to reason that such visually driven workflows are primarily of use in exploratory analysis where the user is not yet familiar enough with the data they have acquired to justify using a predefined workflow. However, such an exploratory analysis at, or soon after the point of data acquisition, is often overlooked, especially by online tools requiring extensive data upload and processing times. A GUI, by design, requires user-presence and the visual workflow demands low computational cost in order to be worthwhile and maintain user engagement. This in turn puts constraints upon the algorithms selected. Existing tools are largely comprehensive, rather than use-case specific, requiring the user to design the

workflow a-priori. In this thesis the analysis of metabolomic data specific to time-course analysis has been extracted into a set of predefined stages, from data acquisition to pathway analysis. User-engagement is additionally considered and the use of methods amenable to give deterministic, rapid results as a preview of what more rigorous algorithms may accomplish. Whilst the use of *d*-*k*-means++ was presented to facilitate this in our analysis, additional research would need to be conducted in order to guarantee similarly near-optimal results with other datasets.

The ST-GP experiment presented here used only two classes of data (boolean- and real-valued). Arguably, this is not reaching the full potential of ST-GP and future research should address this by considering a wider variety of data types. This would further permit a broader range of functionality to be explored since it is reasonable to hypothesise that better results could be achieved by a richer function set. However, since the search space grows exponentially with the number of functions, any additional research on this matter would require more time, or, more preferably, additional computational power, both of which incur their own associated costs.

The study of non-targeted metabolomic datasets is both a data-driven and intuition-driven analysis. Whilst several statistical methods are available of determining an "optimal" set of results, there is no one-size-fits-all solution. It was noted that both the computational cluster analysis pipeline in Chapter 6 and the genetic-programming analysis described in Chapter 7 would serve better from increased computational power. Both methods require multiple iterations of the same procedures in order to optimise results, either in terms of statistics such as silhouette width, or information criteria for cluster analysis, or in terms of maximal fitness for genetic programming. Repetitive iterations such as these are ideally suited to parallel systems such as compute clusters. Future research may wish to explore this, not only in the optimisation of individual clustering parameters, but potentially in the entirety of the pipeline, reducing the need for the human mediator. Doing this without falling back into a "fishing-expedition" would inevitably present a challenge.

Whilst clustering methods can provide results in and of themselves, they are also highly suited to form part of a larger workflow themselves. Focusing the analysis onto individual clusters can reduce the search space and thus permit more computationally expensive analyses such as Bayesian network

inference.

As with any data-driven analysis, the presence of additional data would serve to strengthen any hypotheses made. The target of this thesis has been metabolomic data sourced from LC-MS and NMR. In the analysis of *Medicago* and *Alopecurus* a relatively simplistic m/z annotation method was used. By improving the identification accuracy the accuracy of any pathway predictions post-cluster analysis would also be improved.

The presence of more data could furthermore give weight to any results obtained and the application of concatenated datasets (for example LC-MS+NMR) has seen use in several -omics studies [173]. Data-fusion techniques, incorporating information from different fields, such as genomic and proteomic data, have the potential to be highly useful. The Pathway Tools databases used here incorporate information on mediating enzymes and genes. Methods of combining results from transcriptomic and protein profiling studies would present an avenue for future research that could yield improved results. For the studies considered here, the presence of additional data could serve to strengthen or refute the evidence in support of individual metabolite or pathway elicitation.

This said, a noted problem in the validation of -omics studies is that results are often compared to known values from the literature. Our knowledge of biological systems can be considered to be the set of the most overt elements, which, as our very knowledge of their existence suggests, are unlikely to possess the same features as those which have not been discovered. Additional work is required to assess the ability of the methods discussed here to elucidate these unknown unknowns.

Finally it should be noted that the analyses conducted have been primarily hypothesis finding rather than hypothesis concluding. Many of the metabolites and pathways named have only been tentatively identified. Additional work is required to confirm or reject the presence of any such references and develop a deeper understanding of the underlying biological processes.

Appendices

Appendix A

TABLES OF ELICITED PEAKS

	$\mathcal{FB}$		$\mathcal{DB}$		$\mathcal{D}$	
	LN951	-0.381989349	LN294	-0.742223643	LN108	-0.485320135
	LN886	-0.364821805	LN295	-0.736710805	LP848	-0.47708224
Min	LN1032	-0.35521775	LN293	-0.733107218	LN224	-0.476269119
	LN1026	-0.351240786	LN224	-0.716506101	LP832	-0.475621321
	LN1018	-0.34909925	LP1499	-0.680140079	LP903	-0.471149449
	LN727	0.3841029	LN244	0.743798169	LN575	0.553468985
	LP984	0.367566793	LN230	0.735670884	LN142	0.538512113
Max	LP983	0.356894825	LN211	0.733185211	LN576	0.537729086
	LP986	0.353871524	LN194	0.723249001	LN117	0.531880287
	LP1010	0.351466818	LN292	0.64551783	LN118	0.530639408

	$\mathcal{F}$		$\mathcal{B}$	
	LN956	-0.337424111	LN294	-0.460146432
	LN437	-0.327765428	LN302	-0.45806947
Min	LP1527	-0.322388165	LN295	-0.453685736
	LP825	-0.317251024	LN304	-0.453421752
	LN442	-0.311595774	LN291	-0.439784867
	LN727	0.52607581	LN244	0.551864543
	LN574	0.508921974	LN230	0.541908398
Max	LP904	0.498419817	LN211	0.525108256
	LN963	0.460840356	LP964	0.516177345
	LN149	0.443083785	LN540	0.491245881

Table of the peaks showing the strongest Pearson correlations with time for five combinations of experimental groups as outlined in Section 3.1.

Appendix B

CLUSTERING OF *MEDICAGO* DATA

The tables on the following pages detail the results of clustering the *Medicago* dataset using the *d*-*k*-means++ method. This results are for the combined dataset, with "LN" denoting those peaks originating from the $\mathcal{L}$– dataset and "LP" those from the $\mathcal{L}$+ dataset.

Cluster	1	2	3	4	5	6	7	8	9
Count	78	22	30	25	4	70	25	47	44
Peaks	LN108	LN134	LN42	LN102	LP2036	LN29	LN110	LN67	LN492
	LN149	LN130	LN188	LN121	LP2035	LN32	LN814	LN116	LN1076
	LN209	LN194	LN195	LN131	LP3157	LN74	LN896	LN162	LN1588
	LN208	LN192	LN403	LN163	LP3587	LN76	LN893	LN312	LN1763
	LN229	LN211	LN413	LN432		LN66	LN998	LN600	LP998
	LN226	LN230	LN410	LN479		LN65	LN1070	LN599	LP1128
	LN224	LN244	LN412	LN655		LN64	LN1261	LN678	LP1152
	LN232	LN292	LN411	LN697		LN92	LN1367	LN686	LP1161
	LN242	LN408	LN427	LN887		LN87	LN1369	LN754	LP1307
	LN235	LN571	LN426	LN976		LN84	LN1366	LN762	LP1338
	LN234	LN1030	LN544	LN992		LN147	LN1480	LN764	LP1346
	LN243	LP964	LN611	LN1054		LN191	LN1720	LN785	LP1372
	LN249	LP1042	LN1038	LN1294		LN328	LN1914	LN810	LP1381
	LN248	LP1055	LN1148	LP984		LN415	LP883	LN866	LP1633
	LN247	LP1396	LN1149	LP1027		LN430	LP1079	LN946	LP1815
	LN258	LP1403	LN1184	LP1154		LN442	LP1109	LN933	LP1849
	LN264	LP1473	LN1252	LP1162		LN438	LP1183	LN952	LP1976
	LN270	LP1490	LN1253	LP1202		LN437	LP1250	LN950	LP1990
	LN266	LP1503	LN1255	LP1203		LN444	LP1242	LN1031	LP2054
	LN288	LP1960	LN1273	LP1275		LN464	LP2098	LN1057	LP2134
	LN291	LP1968	LN1764	LP1420		LN651	LP2169	LN1066	LP2249
	LN293	LP2033	LP130	LP2061		LP123	LP2168	LN1080	LP2266
	LN295		LP577	LP2243		LP143	LP2429	LN1246	LP2310
	LN294		LP706	LP2515		LP214	LP2987	LN1351	LP2363
	LN304		LP1515	LP3311		LP218	LP3112	LN1470	LP2362
	LN302		LP2569			LP211		LN1472	LP2546
	LN301		LP2635			LP202		LN1629	LP2545
	LN299		LP2897			LP204		LN1658	LP2535
	LN307		LP2914			LP199		LN1894	LP2547
	LN559		LP2912			LP205		LP664	LP2543
	LN641					LP206		LP663	LP2538
	LN706					LP196		LP693	LP2542
	LN892					LP195		LP696	LP2532
	LN1085					LP142		LP712	LP2539
	LP745					LP201		LP1198	LP2534
	LP859					LP193		LP1291	LP2552
	LP855					LP190		LP1528	LP2548
	LP849					LP188		LP1534	LP2585
	LP848					LP191		LP1547	LP3241
	LP857					LP192		LP2058	LP3248
	LP948					LP194		LP2078	LP3260
	LP1023					LP189		LP2242	LP3318
	LP1378					LP484		LP2300	LP3671
	LP1428					LP482		LP2342	LP3735
	LP1433					LP476		LP2991	
	LP1442					LP658		LP3331	
	LP1443					LP656		LP3546	
	LP1451					LP710			
	LP1456					LP748			
	LP1464					LP759			
	LP1455					LP757			
	LP1465					LP784			
	LP1498					LP774			
	LP1502					LP825			
	LP1499					LP979			
	LP1545					LP1039			
	LP1552					LP1051			
	LP1574					LP1122			
	LP1580					LP1121			
	LP1565					LP1160			
	LP1577					LP1527			
	LP1573					LP1939			
	LP1578					LP1950			
	LP1561					LP1951			
	LP1562					LP1949			
	LP1553					LP1948			
	LP1591					LP1952			
	LP1593					LP1957			
	LP1588					LP2047			
	LP1596					LP2379			
	LP1597								
	LP1595								
	LP1557								
	LP1598								
	LP1600								
	LP2040								
	LP2113								
	LP2877								

10	11	12	13	14	15	16	17	18
76	83	75	39	8	77	63	68	36
LN9	LN81	LN90	LN57	LN540	LN433	LN99	LN14	LN322
LN198	LN80	LN97	LN465	LN658	LN519	LN148	LN53	LN390
LN197	LN88	LN137	LN656	LN867	LN574	LN133	LN30	LN852
LN193	LN86	LN212	LN675	LN884	LN592	LN152	LN77	LN1020
LN269	LN132	LN217	LN695	LP986	LN623	LN166	LN69	LN1190
LN282	LN246	LN222	LN941	LP983	LN625	LN213	LN120	LN1203
LN591	LN453	LN225	LN944	LP982	LN624	LN281	LN112	LN1229
LN824	LN496	LN236	LN943	LP1010	LN620	LN334	LN111	LN1267
LN900	LN495	LN257	LN935		LN619	LN512	LN127	LN1622
LN1023	LN560	LN262	LN961		LN631	LN663	LN125	LN1882
LN1333	LN578	LN268	LN957		LN644	LN719	LN156	LP701
LN1412	LN733	LN414	LN956		LN640	LN747	LN159	LP714
LN1444	LN740	LN469	LN965		LN666	LN753	LN167	LP751
LN1456	LN765	LN468	LN967		LN662	LN869	LN187	LP750
LN1475	LN790	LN483	LN1007		LN659	LN868	LN203	LP781
LN1569	LN786	LN672	LN1182		LN704	LN1006	LN200	LP765
LN1604	LN792	LN717	LN1189		LN701	LN1064	LN214	LP1034
LN1648	LN783	LN795	LN1395		LN707	LN1063	LN227	LP1059
LN1647	LN781	LN804	LN1486		LN727	LN1127	LN283	LP1191
LN1933	LN780	LN801	LN1612		LN729	LN1144	LN434	LP1308
LN1954	LN779	LN855	LN1807		LN766	LN1141	LN441	LP1299
LP18	LN858	LN853	LP107		LN767	LN1197	LN445	LP1367
LP17	LN850	LN870	LP153		LN791	LN1265	LN463	LP1517
LP90	LN864	LN924	LP220		LN891	LN1256	LN482	LP1646
LP93	LN885	LN942	LP660		LN890	LN1259	LN499	LP1647
LP94	LN899	LN1008	LP705		LN889	LN1258	LN501	LP1682
LP133	LN897	LN1011	LP860		LN904	LN1335	LN508	LP1683
LP315	LN926	LN1003	LP1923		LN898	LN1435	LN532	LP1905
LP367	LN969	LN1024	LP2039		LN927	LN1497	LN531	LP1906
LP505	LN989	LN1145	LP2221		LN963	LP1009	LN530	LP1908
LP554	LN986	LN1155	LP2225		LN980	LP1052	LN613	LP1911
LP480	LN982	LN1192	LP2369		LN1005	LP1057	LN618	LP2335
LP475	LN981	LN1250	LP2446		LN1004	LP1251	LN693	LP2469
LP653	LN1016	LN1373	LP2461		LN1017	LP1306	LN746	LP2746
LP685	LN1028	LN1477	LP2649		LN1071	LP1303	LN848	LP2768
LP846	LN1055	LN1476	LP2658		LN1073	LP1936	LN877	LP3442
LP1048	LN1065	LN1505	LP2867		LN1216	LP1984	LN920	
LP1185	LN1101	LN1499	LP3368		LN1362	LP2093	LN918	
LP1194	LN1180	LN1514	LP3496		LN1490	LP2099	LN913	
LP1400	LN1204	LN1553			LP829	LP2198	LN934	
LP1404	LN1205	LN1571			LP904	LP2317	LN929	
LP1432	LN1551	LN1737			LP917	LP2396	LN1019	
LP1482	LP732	LN1889			LP1040	LP2416	LN1132	
LP1500	LP733	LN1899			LP1143	LP2650	LN1198	
LP1504	LP735	LP687			LP1197	LP3054	LN1623	
LP1508	LP731	LP831			LP1425	LP3053	LN1637	
LP1636	LP730	LP1006			LP1448	LP3052	LP486	
LP1792	LP729	LP1007			LP1575	LP3050	LP472	
LP1818	LP746	LP1005			LP1937	LP3106	LP631	
LP1841	LP755	LP1179			LP2111	LP3105	LP626	
LP1843	LP761	LP1298			LP2118	LP3332	LP844	
LP1855	LP756	LP1361			LP2116	LP3329	LP1076	
LP1859	LP812	LP1383			LP2115	LP3330	LP1077	
LP1866	LP818	LP1413			LP2137	LP3328	LP1068	
LP2218	LP858	LP1416			LP2163	LP3325	LP1101	
LP2264	LP873	LP1417			LP2162	LP3326	LP1097	
LP2336	LP863	LP1427			LP2199	LP3374	LP1116	
LP2457	LP919	LP1601			LP2219	LP3375	LP1126	
LP2678	LP905	LP1606			LP2417	LP3372	LP1132	
LP2692	LP954	LP1607			LP2424	LP3371	LP1131	
LP3281	LP1016	LP1919			LP2425	LP3387	LP1330	
LP3376	LP1030	LP2010			LP2430	LP3527	LP1512	
LP3430	LP1021	LP2056			LP2463	LP3786	LP1526	
LP3475	LP1037	LP2095			LP2485		LP1642	
LP3509	LP1041	LP2100			LP2800		LP1652	
LP3701	LP1046	LP2122			LP2799		LP2049	
LP3769	LP1049	LP2154			LP2798		LP2184	
LP3775	LP1063	LP2156			LP2796		LP2788	
LP3778	LP1603	LP2182			LP2795			
LP3795	LP2081	LP2214			LP2794			
LP3807	LP2102	LP2209			LP2876			
LP3810	LP2129	LP2215			LP2886			
LP3818	LP2195	LP2339			LP2882			
LP3820	LP2241	LP2467			LP2884			
LP3824	LP2244	LP2464			LP2881			
LP3825	LP2246				LP2885			
	LP2245				LP2878			
	LP2250							
	LP2253							
	LP2252							
	LP2419							
	LP2438							
	LP2594							

19	20	21	22	23	24	25
87	84	18	67	47	72	66
LN98	LN41	LN517	LN58	LN256	LN82	LN96
LN135	LN39	LN627	LN44	LN491	LN118	LN100
LN154	LN51	LN713	LN36	LN570	LN117	LN109
LN153	LN71	LN718	LN38	LN646	LN122	LN165
LN151	LN205	LN715	LN61	LN716	LN142	LN184
LN150	LN240	LN714	LN31	LN741	LN139	LN185
LN186	LN404	LN911	LN28	LN744	LN155	LN179
LN189	LN577	LN902	LN63	LN748	LN182	LN545
LN220	LN874	LN901	LN175	LN745	LN174	LN561
LN231	LN1048	LP835	LN173	LN763	LN171	LN589
LN245	LN1171	LP893	LN204	LN835	LN170	LN734
LN261	LN1339	LP1305	LN223	LN923	LN169	LN757
LN259	LN1345	LP1971	LN431	LN925	LN196	LN756
LN305	LN1353	LP2193	LN448	LN1079	LN201	LN822
LN300	LN1411	LP2192	LN486	LN1153	LN398	LN856
LN321	LN1432	LP2200	LN542	LN1230	LN423	LN876
LN504	LN1503	LP2197	LN557	LN1322	LN443	LN888
LN650	LN1606	LP2420	LN572	LN1400	LN515	LN997
LN667	LN1635		LN567	LN1548	LN510	LN1086
LN712	LN1649		LN732	LN1641	LN507	LN1115
LP881	LN1776		LN730	LN1705	LN566	LN1130
LP1376	LN1806		LN768	LN1944	LN576	LN1143
LP1377	LN1831		LN808	LP847	LN575	LN1159
LP1375	LN1871		LN812	LP889	LN603	LN1169
LP1382	LN1880		LN922	LP1213	LN607	LN1173
LP1379	LN1898		LN937	LP1232	LN602	LP740
LP1384	LN1907		LN1185	LP1226	LN601	LP753
LP1389	LN1947		LN1337	LP1205	LN617	LP845
LP1386	LP393		LN1492	LP1789	LN628	LP832
LP1385	LP374		LN1512	LP1802	LN679	LP836
LP1387	LP347		LN1734	LP1910	LN708	LP833
LP1392	LP314		LN1866	LP1935	LN731	LP842
LP1394	LP552		LP165	LP1944	LN831	LP853
LP1399	LP675		LP330	LP1954	LP474	LP907
LP1410	LP684		LP230	LP2140	LP676	LP906
LP1415	LP975		LP212	LP2175	LP690	LP903
LP1419	LP997		LP200	LP2176	LP742	LP952
LP1424	LP1167		LP249	LP2185	LP741	LP985
LP1431	LP1467		LP479	LP2208	LP762	LP1032
LP1435	LP1494		LP633	LP2207	LP764	LP1024
LP1438	LP1523		LP624	LP2217	LP795	LP1020
LP1429	LP1837		LP623	LP2216	LP777	LP1025
LP1445	LP1895		LP627	LP2895	LP823	LP1036
LP1444	LP1901		LP670	LP3094	LP824	LP1043
LP1450	LP1902		LP704	LP3298	LP843	LP1060
LP1441	LP2031		LP703	LP3297	LP1058	LP1106
LP1454	LP2150		LP816	LP3757	LP1089	LP1129
LP1452	LP2149		LP1104		LP1091	LP1176
LP1430	LP2223		LP1124		LP1080	LP1182
LP1457	LP2316		LP1123		LP1050	LP1180
LP1462	LP2319		LP1134		LP1135	LP1181
LP1460	LP2422		LP1133		LP1144	LP1211
LP1477	LP2427		LP1149		LP1141	LP1237
LP1478	LP2445		LP1146		LP1195	LP1300
LP1491	LP2516		LP1265		LP1208	LP1341
LP1544	LP2578		LP1397		LP1931	LP1535
LP1548	LP2588		LP1398		LP1938	LP1537
LP1551	LP2600		LP1481		LP1996	LP1539
LP1576	LP2679		LP2005		LP2002	LP1617
LP1563	LP2680		LP2004		LP2001	LP1985
LP1583	LP2757		LP2021		LP2019	LP2038
LP1581	LP2836		LP2020		LP2069	LP2059
LP1568	LP2899		LP2105		LP2068	LP2060
LP1579	LP2908		LP2605		LP2067	LP2112
LP1567	LP2928		LP2739		LP2070	LP2127
LP1558	LP3000		LP3802		LP2084	LP2555
LP1566	LP3012		LP3828		LP2083	
LP1556	LP3037				LP2089	
LP1560	LP3077				LP2090	
LP1549	LP3080				LP2088	
LP1559	LP3131				LP2087	
LP1555	LP3145				LP2303	
LP1554	LP3143					
LP1564	LP3187					
LP1590	LP3273					
LP1587	LP3344					
LP1594	LP3453					
LP1609	LP3458					
LP1612	LP3468					
LP1659	LP3593					
LP1661	LP3699					
LP1662	LP3759					
LP1660	LP3770					
LP1664	LP3811					
LP1666						
LP1668						
LP1669						

Appendix C

METABOCLUST USER GUIDE

C.1 System requirements

For large datasets, a 64-bit system with 8Gb RAM is recommended.

MetaboClust is dependent on the .NET framework. If you are running a recent version of Windows it is more than likely that this is already installed on your computer. If not you will need to download the installer version (see below), or install the framework from one of the following URLs:

- **Windows** – download the Microsoft .NET from `https://www.microsoft.com/net/download`
- **Windows/Linux/Mac** – download The Mono Project from `http://www.mono-project.com/download/`

C.2 Compiling from source

MetaboClust is written in C# using Visual Studio 2015. The source consists of three projects, all of which must be downloaded:

Project	**Relative path**	**Contents**	**Download URL**
MetaboliteLevels	`./MetaboliteLevels/MetaboliteLevels/MetaboliteLevels.csproj`	The main application	`https://bitbucket.org/mjr129/metabolitelevels`
MChart	`./MChart/MChart/MChart.csproj`	Charting library	`https://bitbucket.org/mjr129/mchart`
MGui	`./MGui/MGui/MGui.csproj`	Helper library	`https://bitbucket.org/mjr129/mgui`

From the *downloads* page of each of the projects, select *download repository*. Unzip each of the downloads to a new folder on your disk. If any of the above libraries show as missing make sure they are present in the correct folder, or modify your solution to target the correct path.

MetaboClust also requires the following libraries. Initially these will show as *missing*, but should be downloaded automatically by *NuGet* during the first build. If you have disabled *NuGet* in VS2015 you will need to add the libraries to the solution manually.

C.2.1 Running the source

Build and run the *MetaboliteLevels* project to start the application. Note that due to optimisations being skipped, the application will run considerably slower if the build mode is set to *≪debug≫ and/or* a debugger is attached.

- MathNet.Numerics
- RDotNet
- JetBrains.Annotations

C.3 Downloading binaries

If you are not compiling from source, download the application from `https://bitbucket.org/mjr129/metabolitelevels/downloads`. The downloads

come in two flavours, *Installer* and *Exe*. MetaboClust is a stand-alone application and should not require any special install, hence the *Exe* version is fine. However, if a full installation is preferred, which includes the .NET framework (if required), desktop and start-menu shortcuts and an un-installer, the *Installer* version can be downloaded instead.

C.3.1 Running the stand alone version

After downloading and unzipping, launch *MetaboliteLevels.exe* to start the application.

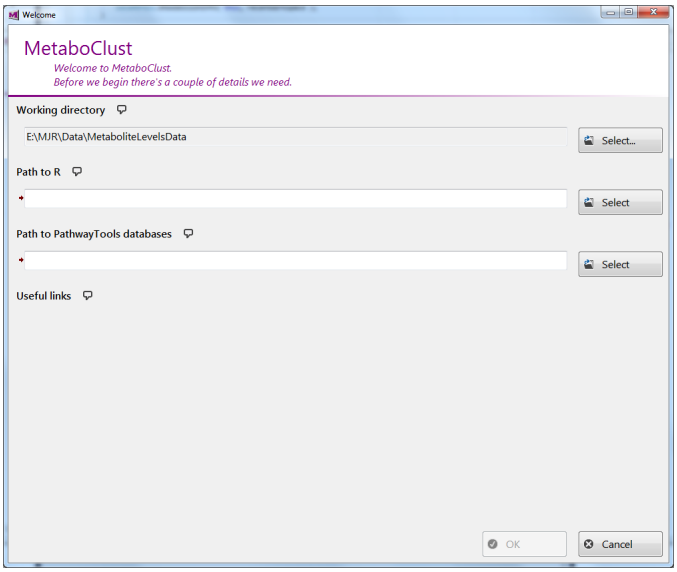
C.3.2 Running the installer

After downloading and unzipping, run *Setup.exe* and follow the on-screen instructions. The application will be installed using Microsoft ClickOnce – see `https://msdn.microsoft.com/en-us/library/t71a733d.aspx` for troubleshooting and details. After the install you should be able to run the application from your start menu, or by launching *MetaboliteLevels.exe* from the folder you installed the application to.

Note

If an error message appears when you try to start the application, check that the latest version of the .NET framework is installed and working.

C.4 Initial setup



When MetaboClust starts for the first time the initial setup screen shown above is presented. This requires the following information.

Initial setup options

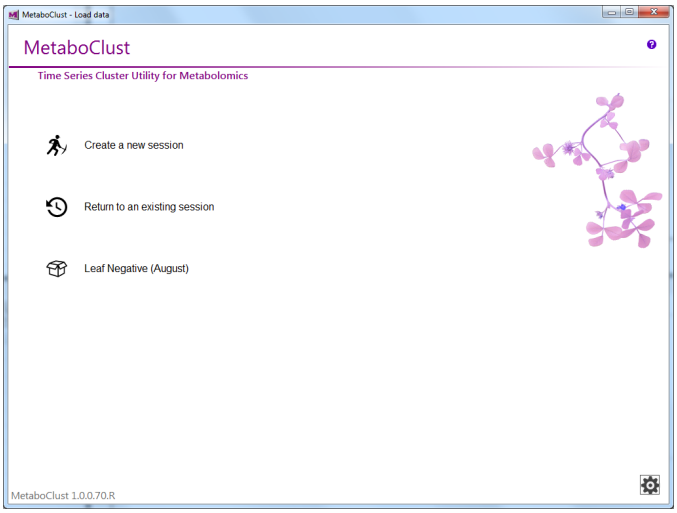
- **Working directory** – This is where the application stores its data. By default this is the application's home directory. The default value should suffice in most case but can be changed (e.g. if administrator permissions deny read-write access to that folder).
- **Path to R** – MetaboClust uses R to operate and needs to know where R is located. Clicking the *≪select≫* button to the right of the text box should automatically detect the location of R and present a drop-down list of the versions of R available. If MetaboClust cannot find an R installation, the path to R will need to be specified manually. Pressing the *≪select≫* button (and then, if required, the *≪browse≫* option) will prompt you to locate the R installation. On Windows, R is usually located at `C:`
  `Program Files`
  `R`
  `R-x.x.x`

`bin`
`x64`, where `x.x.x` is the version. This folder can be identified by the presence of the R library, `R.dll`.

- **Pathway tools databases** – MetaboClust uses Pathway Tools databases to make identifications. If any databases are already present on the system MetaboClust can be directed to them here. If no databases are available the *≪select≫* button will offer a default location which can be used to put the databases in when you get some.

When you are done, click the *≪OK≫* button to commit the selections. MetaboClust detects the presence of errors on most screens. The software will check a connection to R can be established, and check to make sure it has read/write access to the data folders. A greyed out *≪OK≫* button indicates an error and a small red arrow should point in the direction of anything amiss. Hover the mouse over the arrow for more details.

C.5 Loading data



Once the initial setup is completed the application will start on the data-load screen shown above. The ✿ icon in the bottom right of the window presents a drop down menu and the *≪edit paths and libraries≫* option here will return you to the *initial setup* screen.

C.6 Creating a new session

A MetaboClust "session" is a database of your data, annotations and analyses. You need to create a session before any analysis is performed. Select ≪*create a new session*≫ on the data-load screen to create a new session. The application will walk you through its creation.

Important note

Clicking the ≪*show help*≫ button (or in newer versions the  icon) will show a ***context sensitive*** help bar at the side of the screen containing up-to-date details of the input fields. For inputs requesting files the ≪*show file format details*≫ button within the help bar describes the expected layout of input files.

Clicking the ≪*Next*≫ button progresses to the next stage of input. If this greyed out a small red arrow will point to anything amiss. Hovering the mouse over the arrow should describe the problem.

Loading data

- **Template** – Allows you to start from a previous setup. Normally you will start with the ≪*blank template*≫.
- **Session name** – For your reference only
- **Data set**

  Source – If you have LC-MS data MetaboClust needs to know how the adducts are formed. If the data is not sourced from LC-MS, or automated annotations are not required, then select ≪*Source = Other*≫, otherwise select the column mode. The ≪*Source = Mixed mode*≫ option allows you to mix modes, but your ≪*peaks*≫ file must then contain an extra column specifying the mode of each peak (≪*1*≫ or ≪*-1*≫).

  Intensity matrix – The intensity matrix is a grid containing

the recorded intensities, with 1 row per observation and 1 column per variable (peak). Row and column headers must be provided and must specify unique names for all observations and peaks. See the help bar as described above for exact details.

Observation information – The observations matrix describes details about each observation, with one observation on each row and one field of information in each column. Row headers should contain the observation IDs as specified for the *≪Intensities≫* matrix and column headers should contain the field names. Most fields are optional, but if you don't specify them then some features won't work (for instance batch correction requires the *≪batch≫* and/or *≪acquisition order≫* fields). Since the exact file format may change with each release, please see the help bar in the software itself for the list of fields (column headers) available.

Peak information – Like the observations matrix, this provides details about each dependent variable. The software refers to all dependent variable as peaks to avoid ambiguity with other variables, such as algorithm parameters. Please see the help bar for the list of fields available.

Alternate intensities – Sometimes another version of your data may be available, such as one prior to noise removal or scaling. The alternate intensities option allows this to be loaded in for quick reference later. Aside from allowing you to view it, it will have no effect on the actual analysis. This feature is not present from version 1.2 as an unlimited number of intensity matrices can be loaded from the file menu.

Condition names – If your experimental groups have unintuitive names, such as "1", "2" and "3" then this allows you to map these to more a readable title.

- **Conditions**

  Specify conditions – Details of the experimental groups can be provided here. The conditions should be given as in your *observation information* file or, if present, your *condition names* file. This information is not mandatory, but if specified the software will be able to generate default statistics and filters (described

later) for you. If you don't specify the conditions these can still be added manually later.

- **Statistics**

  Auto-create statistics – You can choose to generate t-tests for your experimental groups against control, as well as pearson correlations of your intensities for each group against time. These options are not available if you didn't specify the conditions earlier. If you don't do this, you can add the statistics manually later.

  Perform corrections – You can add the UV-scale and centre data correction to your pipeline here. This and other corrections can always be added or modified later.

- **Compound libraries** – These are the compound and pathway libraries used for annotations and pathway analysis. One or more of these must be selected to enable automated annotations. If you don't have any libraries on your system then the list will be empty; see Section C.4 on how to specify a library folder.

  Adduct libraries – These are the adduct libraries used for automated annotations. There are two built into MetaboClust, *All* and *Refined*. The *All* library contains all adducts listed at [321], whilst the *Refined* library contains a common subset of these.

- **Automated identification** – This will annotate peaks with potential metabolite identifications. The option will be unavailable if required information is missing. If this is selected the ≪*tolerance*≫ must be specified, as well as the ≪*annotation status*≫ to assign the automated annotations. These statuses are ≪*tentative*≫ (unconfirmed identity), ≪*affirmed*≫ (computationally confirmed) or ≪*confirmed*≫ (experimentally confirmed).

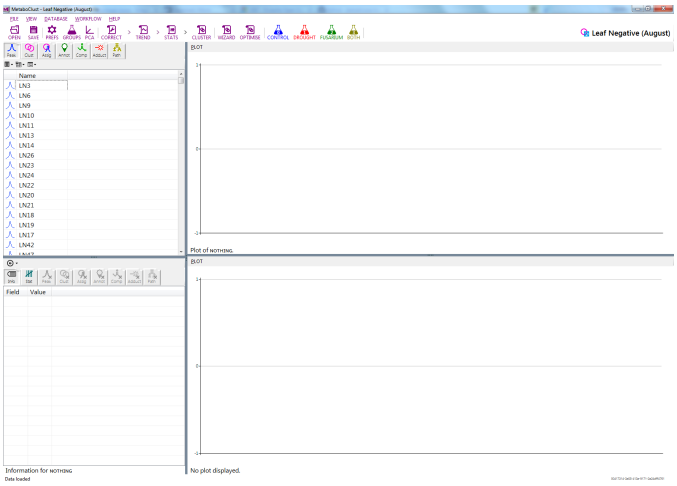
  Peak-peak-matching – This annotates peaks with other peaks based on m/z similarity and is primarily used to search for related compounds.

  Manual identifications – Manual identifications can be loaded from disk. Again, see the help bar for the exact file format.

> The ≪*annotation status*≫ specified here will only be used if that information is missing from the file itself.

When all the fields you wish to select are complete click the ≪*OK*≫ button to load the data. This may take a few minutes, especially if automated peak-compound annotations are being performed. Saving the session will avoid this delay in future.

C.7 Data exploration



Once you have created or loaded a session you will be presented with the main screen, shown above. As there is no fixed set of steps in analysing a dataset but a brief overview will be presented here. The images here are taken from the analysis of the *Medicago* leaf data. This dataset comprises 184 observations and 2920 peaks, with four experimental groups ($\mathcal{C}$, $\mathcal{D}$, $\mathcal{F}$, $\mathcal{B}$), as well as QC samples. Peaks were automatically annotated using the MedicCyc database [320].

C.8 Univariate statistics

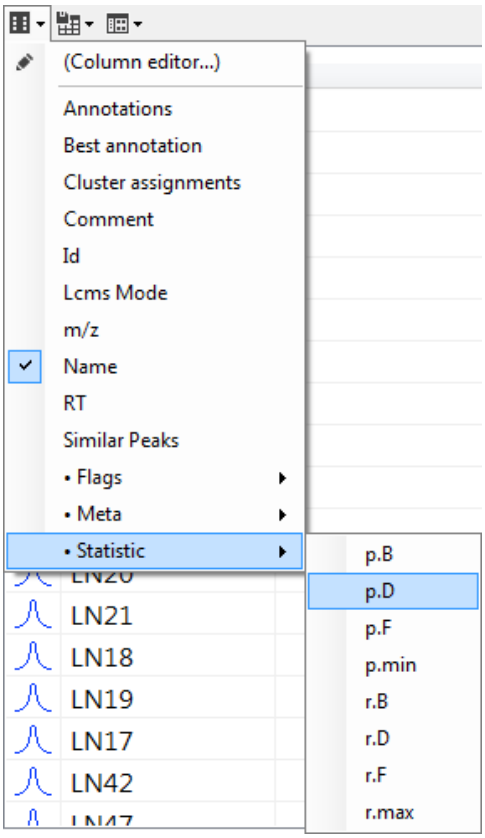
Double clicking a peak in the list to the top-left of the window will present a plot of the chosen peak should be displayed to the right. This is a useful first step to ensure the data has loaded correctly.

Graph controls

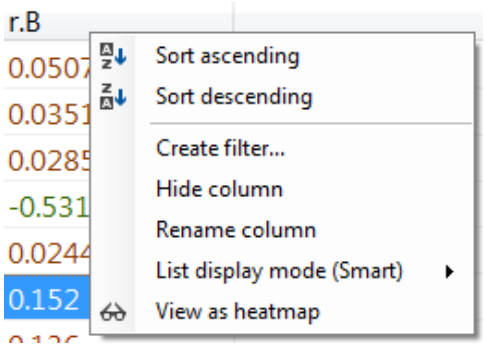
- **Left or right click** – Select point or series. Details on that point will be displayed above the graph. Repeatedly clicking will cycle through any overdrawn points.
- **Left click and drag** – Box-zoom
- **Mouse wheel** – Zoom in or out
- **Middle click** – Restore zoom and cancel selection

The *≪plot≫* button above the graph provides plotting options, including exporting the plot to a file and toggling display of the legend.

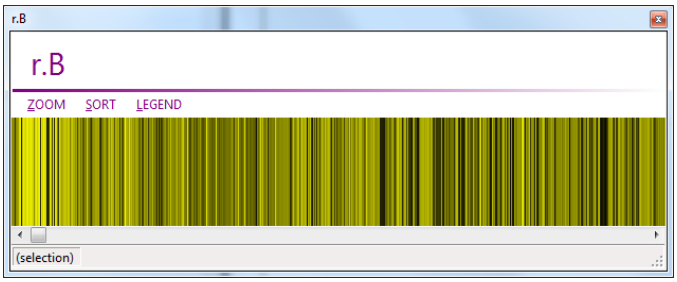
To show more information about each peak, click the  icon above the peak list. Try showing one of the statistics:



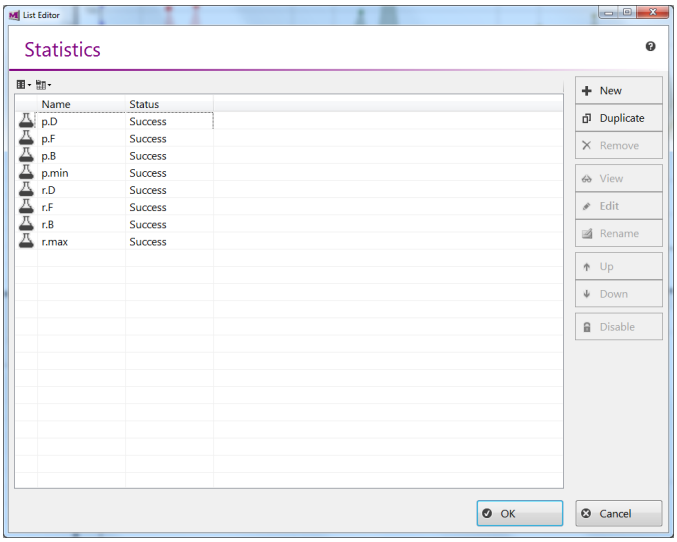
You should then be able to sort the column by that statistic, allowing you to locate the most "significant" peaks:



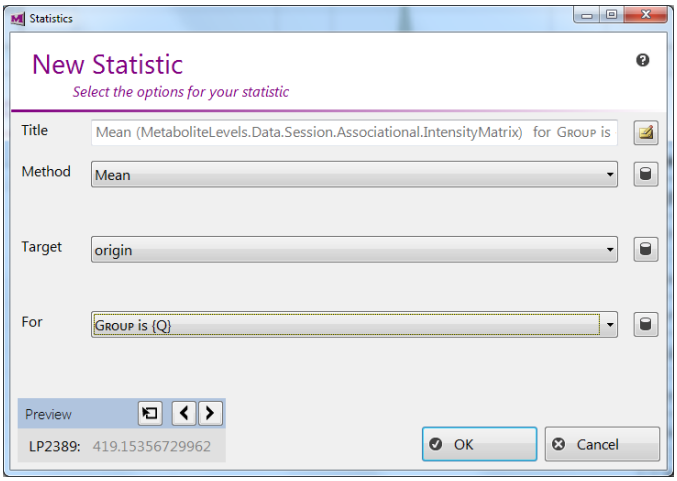
Clicking *≪view as heatmap≫* will present a heatmap of the column. Note that the peaks are ordered in the heat-map in the same order as the column, so if you sorted the column first, the heat-map will be sorted as well and will appear as a gradient.



To add univariate statistics, click the STATS *≪stats≫* option from the toolbar, or select *≪Database/Workflow/Statistics≫* from the menu.



Select ≪*New*≫ to create a new statistic.



Statistic fields

- **Title** – The title of your statistic. A name will be provided for you if you don't specify one. Clicking the  icon provides space to add detailed comments.
- **Method** – The statistic to calculate, click the  button to the right of the method to define your own methods.
- **Parameters** – If the method takes any parameters, enter them

here. Multiple parameters are separated by commas. Clicking the button next to the text-box displays the parameters as individual inputs rather than a single-line text-box.

- **Target** – The intensity matrix to work on, from various stages of your analysis. "origin" indicates the original intensity matrix you loaded in and will be the only option until you perform data-correction. Items marked with an asterisk designate dynamic sources. Pre-version 1.2 only the latest two intensity matrices are available, which are the latest set of observations (≪**final correction*≫) and trend (≪**final trend*≫).

- **For** or **Compare** – Selects the filter to input vector to the statistic, defining the set of observations to use. Click the  button to define new filters.

- **Against** – Only available for bivariate statistics, specifies the second input vector:

  The corresponding time – The times corresponding to the first input vector (e.g. to correlate intensity against time)

  A different peak – The set of intensities corresponding to the first input vector for a different peak (e.g. to find similar peaks)

  The same peak – The set of intensities sourced from different observations on the same peak (e.g. to contrast experimental and control observations).

For instance to calculate the mean of the QC samples select ≪*Method = Mean*≫ and ≪*For = Group is Q*≫. The ≪*Preview*≫ box allows you to preview the result of your calculation on individual peaks. Select ≪*OK*≫ when you are done.

Click ≪*OK*≫ again to leave the ≪*List editor*≫. Any new or modified statistics will be recalculated. Edit the columns above the peaks list to show your new statistic.

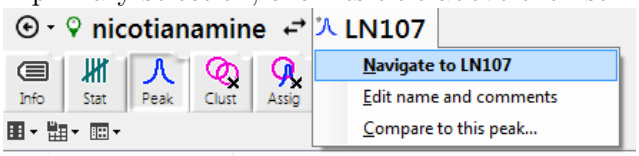
C.9 Exploring annotations

The set of coloured icons above the list allows paging between the database contents. If automated annotation was chosen when loading the data try selecting the Annot tab and viewing the annotations.

Double click an annotation to view it. Since annotations don't have graphs nothing will be displayed in the top-right, but the secondary list in the bottom-left should update to reflect the selected annotation. Above the secondary list select the *≪peak≫* tab to display the peaks associated with the selected annotation. Double-click the peak which appears in the list to plot the peak associated with the annotation.

Data exploration

Almost all of the data in MetaboClust can be explored in this way. The primary list (top) selects items within the dataset, whilst the secondary list (bottom) allows you to explore items within the context of the primary selection. To select an item in the secondary list as the primary selection, click its title above the list:



C.10 Multivariate statistics

An overview of your data can be obtained using PCA. Click the PCA button in the menu strip to launch the PCA window.

C.11 PCA

The PCA window presents a PCA plot of the dataset. The options to the left control the method of PCA and the display of the scores.

PCA controls

- **Method** – Switch between PCA and PLSR plots
- **Source** – Decide whether you are performing PCA on the observations or the variables (the peaks)
- **View** – Toggle between scores and loadings plots.
- **Legend** – Select what the colours on the graph represent
- **Corrections** – View your data with various corrections.[a]
- **Input** – Choose between performing PCA of all observations, or just your trend line (useful for noisy datasets)[1] *a*
- **Observations** – Select a filter on the set of observations to explore
- **Peaks** – Select a filter on the set of peaks to explore
- **View on main** – Displays the selected peak or observation on the main screen.[b]
- **Mark as outlier** – Applies an observation or peak filter, excluding the selected observation. [2]b
- **Next component** – Views the next principal components
- **Previous component** – Views the previous principal components
- **Plot options** – Displays the set of plot options, including toggling display of the legend.

[a] Corrections and trends are defined from the main screen
[b] Requires an object in the plot to have been selected first

If your data was collected in batches for instance, click the Legend – the Batch to colour the plot by batch.

Certain subsets of the data can also be selected, click the Observations menu should show a list of observation filters, allowing you to filter on ex-

perimental group. As when creating your statistic, if no filters are available you can click *≪Observations/New filter...≫* to create a new filter. The same can be done with peak filters by selecting *≪Peaks/New filter...≫*.

PCA can also be used for outlier removal. Click an observation in the plot and select the MARK AS OUTLIER button. A new filter will be created, excluding that observation (or peak) from the dataset.

C.12 Data correction

Select *≪Correct≫* from the menu-bar of the main screen to open the corrections list. It's empty right now so click *≪new≫* to create a new one.

The data correction window presents a list of data-correction methods, as well as trend generation methods. Data-correction methods, such as scaling and centring act alone, whilst the trend-generators can be used to perform batch correction and control correction.

Data correction options

- **Title** – As described in Section C.8.
- **Source** – As described in Section C.8.
- **Method** – As described in Section C.8.
- **Parameters** – As described in Section C.8.
- **Source** – As described in Section C.8.
- **Operator** – *Only available for trend-based corrections.* The correction takes the form $x\prime = f(x,t)$, where f is defined as / or $-$. Generally a batch correction will use *≪divide≫*, and control correction *≪subtract≫*.
- **Filter** – *Only available for trend-based corrections.* Selects the set of points used to generate the trend

The preview window allows you to preview the correction on an individual peak. For trend-based corrections the trend used will be highlighted

to the left.

C.12.1 Examples

C.12.1.1 QC correction

Dividing by the mean of the QC samples in the batch is a fairly standard method of correcting for batch-differences in LC-MS, to use this select: *≪Method = straight line across mean≫*, *≪Corrector = Batch≫*, *≪Operator = Division≫* and *≪Filter = Group is Q≫*.

C.12.1.2 Background correction

To perform background correction, as described in Chapter 4 select *≪Method = moving median≫*, *≪Corrector = Batch≫* and *≪Filter = All≫*. You will need to enter the window width *≪w≫* parameter in this case. Experiment with values to find one that looks good in the plot.

C.12.1.3 Scale and centre

Select *≪Method = UV scale and centre≫*. As a direct correction, rather than a trend, there are no other options to choose. This correction should generally be performed *after* batch correction and therefore the *≪Source≫* parameter should point to the intensity matrix generated by your batch correction – QC or Background correction as described above.

C.12.2 Viewing corrections

Back on the main screen you will need to select your corrected dataset before your changes can be viewed. Click DATASET *≪dataset≫* or the drop-down list next to it to select your modified data. You can use the *≪*Final correction≫* meta-option to always keep your display up-to-date with the latest correction.

C.13 Trend line generation

You might have noticed the bold lines through 0 on your peak plots (or no lines at all post-version 1.2). These are present because there is currently

no trend line defined. Select TREND *≪trend≫* from the tool-bar or *≪Database/-workflow/trends≫* to define a trend.

You will be presented with a list much like the *≪correction≫* window. (Pre-version 1.2 this will containing a *≪no-trend≫* entry, click *≪remove≫* to get rid of it). Click *≪new≫* to create a new trend.

Trend options

The *≪New trend≫* options are largely the same as those described in Section C.12.

C.13.1 Examples

C.13.1.1 Replicate removal

The simplest trend is the mean for each time-point. Select METHOD = MOVING MEAN with value W = 1. (W is the window width for the moving mean – 1 simply indicates a window width of 1, effectively a mean of replicates).

C.13.2 Viewing trends

Back on the main screen you will need to select your trend before your changes can be viewed. Click *≪trend≫* or the drop-down list next to it to select your modified data. You can use the *≪*Final trend≫* meta-option to always keep your display up-to-date with the latest correction.

After selecting your trend any peaks you plot will use the specified trend.

C.14 Clustering

Clusters are created in the same way as corrections, statistics or trends. Select the *≪Cluster≫* option from the tool-bar of the main window.

- Title – As described in Section C.8.

- Method – As described in Section C.8.
- Parameters – As described in Section C.8.
- Peaks – Which peaks to cluster. Since some peaks can interfere with clustering it can be good to filter them out. If you haven't got a suitable filter defined, select the  icon next to the list.
- Distance – The distance metric to use. Whilst not used for externally provided algorithms this is still used to calculate certain statistics (next option).
- Parameters – Parameters to the distance metric, as described in Section C.8.
- Statistics – Statistics to calculate for clusters
- Source – As described in Section C.8.
- Observations – The set of observations to use in the clustering vectors
- One vector per experimental group – Normally one vector is created per-peak, select this option to "split" the peaks into one vector for each experimental group.
- Parameter optimiser – If the clustering algorithm takes parameters this option can be used to optimise them using statistics such as *silhouette width* or *BIC*.

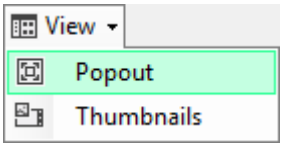
Alternatively, selecting the *≪wizard≫* option from the main menu will guide you through clustering using the *d-k-means++* algorithm, which is a deterministic variant of k-means developed for this software and useful for rapid data exploration.

C.14.1 Viewing clusters

On the main screen click the *≪cluster≫* icon above the primary list to view the clusters. Double click a cluster to plot it in the cluster plot area. Note that clusters are always plotted using the vectors with which they were created, so the *≪trend≫* and *≪dataset≫* visual options will have no effect

on the cluster plot.

Clicking a vector within the cluster plot will select the peak associated with that vector as the secondary selection. Alternatively, select the ≪*peaks*≫ tab from the secondary list to show a list of peaks assigned to the selected cluster. Double clicking a peak in this list will plot the peak and highlight it in the cluster plot.



If you want to see a quick overview of all clusters, then click the ≪*View*≫ and ≪*Popout*≫ options above the list of clusters. By default each plot is scaled to fit the plot area, so flat clusters may appear as noisy. To change this and scale all clusters to the same Y-axis, change the plot options by going to the ≪*Prefs*≫ window and setting the ≪*Cluster*≫ – ≪*Y-axis range*≫ to ≪*Scale to matrix*≫.

C.14.2 Metabolite and pathway exploration

With a cluster selected, clicking the ≪*compounds*≫ or ≪*pathways*≫ options will show compounds and pathways potentially highlighted by that cluster. Double-clicking these compounds or pathways will highlight the overlap between them and the cluster in the cluster plot. You can show or hide the degree of overlap, or sort clusters by overlap, by selecting the  icon in the secondary list.

A reverse exploration can also be performed, selecting a ≪*pathway*≫ or ≪*compound*≫ in the primary list will plot the trends of the peaks associated with the pathway or compound in the cluster plot. (You can change the ≪*dataset*≫ or ≪*trend*≫ in this case.) As for the clusters, selecting individual trends will plot the actual peak. Selecting ≪*clusters*≫ in the secondary list will show the clusters affected by peaks annotated with the pathway or compound.

C.15 General options

- Show or hide observations from experimental groups – Click the group icon in the main tool-bar to toggle group visibility, or select the 

≪*groups*≫ icon.

- Rename groups, peaks, etc. – Select the ≪*Database*≫ menu to show the database, then edit the group or peak. The groups database can be accessed quickly from the GROUPS ≪*groups*≫ icon in the tool-bar. Clicking the name of the session in the top-right of the main screen allows you to rename the session.
- Change display options – Select the PREFS icon from the tool-bar.
- Find out which files were used to create a session – Select ≪*Help/Session information*≫ from the main menu
- Find an individual item – Click the ≪*name*≫ column of the peaks list and select ≪*filter*≫ from the menu to search for individual items.
- Get an overview of the session, including peak and observation counds – Select ≪*View/Miscellaneous functions*≫ and then ≪*View statistics*≫ from the window that appears.

C.16 Known bugs

MetaboClust is beta software. A list of known bugs is maintained on the download page. Please submit any bugs you find to this list.

Appendix D

BNF FUNCTION LISTING FOR GP

The listing is given as a series of BNF definitions for C++. The unique expression `Constant(min, max)` represents a randomly generated constant of the declaring type, between `min` and `max`, whose value is fixed once obtained.

```
*** ROOT ***
<root> ::= <class_and_age> // root function

*** CLASS_AND_AGE ***
<class_and_age> ::= ( predictedClass = <double>,
                    class_and_age( predictedClass,
                                   <double> ) ) // predict class first
<class_and_age> ::= ( predictedAge = <double>,
                   class_and_age( <boolean>,
                                 predictedAge) ) // predict age first

*** BOOLEAN ***
<boolean> ::= Constant(false, true) // constant
<boolean> ::= !<bool> // not
<boolean> ::= (<bool> && <bool>) // and
<boolean> ::= (<bool> || <bool>) // or
<boolean> ::= (<bool> ^ <bool>) // xor
<boolean> ::= (<double> == <double>) // equal to
<boolean> ::= (<double> < <double>) // less than
<boolean> ::= (<double> > <double>) // more than
<boolean> ::= (<double> >= <double>) // more than or equal to
<boolean> ::= (<double> <= <double>) // less than or equal to
<boolean> ::= (<double> != <double>) // not equal to
<boolean> ::= predictedClass

*** DOUBLE ***
<double> ::= Constant(0.0, 1.0) // constant
```

```
<double> ::= (<double> + <double>) // add
<double> ::= (<double> - <double>) // subtract
<double> ::= (<double> * <double>) // multiply
<double> ::= (<double> / <double>) // divide
<double> ::= Sin(<double>) // sin
<double> ::= Cos(<double>) // cos
<double> ::= Tan(<double>) // tan
<double> ::= Max(<double>, <double>) // arg. max
<double> ::= Min(<double>, <double>) // arg. min
<double> ::= IEEERemainder(<double>, <double>) // remainder
<double> ::= data[<int>] // get variable
<double> ::= (data[<int>] / data[<int>]) // ratio of variables
<double> ::= ((data[<int>] + data[<int>]) / 2) // average of variables
<double> ::= predictedAge

*** INT ***
<int> ::= Constant(0, variable_count) // constant
<int> ::= ((int)<double>) // index from number
```

Appendix E

LIST OF ABBREVIATIONS

d-k-means++ Clustering method based on k-means. 145, 150, 160, 165, 193, 197

k-means Clustering method. 127, 136, 144, 150, 190, 191

k-means++ Clustering method based on k-means. 136, 145, 150, 191

t-test Univariate statistic. 144

t_r retention time. 22

$\mathcal{B}$ Experimental group of the *Medicago* study – both drought-conditioned and *Fusarium*-inoculated. 62, 71, 74, 116, 132, 133, 146, 147, 209

$\mathcal{C}$ Experimental group of the *Medicago* study – unaffected plants. Experimental group of the Beef study – cold stored. 62, 71, 74, 77, 90, 116, 146, 156, 172, 185, 186, 209

$\mathcal{D}$ Experimental group of the *Medicago* study – drought-conditioned plants. 62, 71, 74, 76, 116, 146, 147, 156, 209

$\mathcal{F}$ Experimental group of the *Medicago* study – *Fusarium*-inoculated plants. Experimental group of the Beef study – frozen stored. 62, 71, 74, 77, 116, 146, 147, 209

$\mathcal{L}+$ leaf-positive. 4, 9, 10, 63, 64, 66, 69, 102, 105, 107, 108, 123, 133, 197

$\mathcal{L}+/-$ leaf-combined. 66, 71, 74, 135

$\mathcal{L}-$ leaf-negative. 9, 63–68, 70, 105, 197

$\mathcal{M}$ Experimental group of the *Alopecurus* study – Multiple herbicide resistant – a variety of herbicide resistant plant. 157, 159

$\mathcal{S}$ Experimental group of the *Alopecurus* study – susceptible – a variety of non-herbicide resistant plant. 157

$\mathcal{T}$ Experimental group of the *Alopecurus* study – Target site resistant – a variety of herbicide resistant plant. 157, 159

$\mathcal{W}$ Experimental group of the Beef study – warm stored. 77, 90, 172, 185, 186

$\mathcal{Z}$ Day zero samples group of the Beef study. 77

Medicago truncatula A model legume. 115

^1H *NMR* Variety of NMR specific to hydrogen-1 nuclei. 23, 32, 41, 42, 45, 46, 48, 50, 54, 55, 127, 129

ANN artificial neural network. 54

ANOVA analysis of variance. 40, 41, 46, 66

AP affinity propagation. 128, 129, 136

ASCA analysis of variance simultaneous component analysis. 42

autoscaled Scaled to unit variance and mean centred. 64

BASIC Beginner's All-purpose Symbolic Instruction Code. 169

BNF Backus-Naur-form. 169, 171

CCA canonical correlation analysis. 48, 51, 52

CLASSY cluster analysis statistical spectroscopy. 56

CODA the component detection algorithm. 31

COSY correlation spectroscopy. 55, 56

COW covariance-optimized warping. 31

CSV comma separated value. 143, 146

CV cross validation. 56

CVA canonical variate analysis. 51

CWT continuous wavelet transform. 30

DBSCAN density-based spatial clustering of applications with noise. 29

DTW dynamic time warping. 31

ECVA extended canonical variate analysis. 52

EIC extracted-ion chromatogram. 29, 35, 37

elicitor Substances that stimulate the formation of certain compounds *in-vivo*. 115

ESI electrospray ionisation. 21

FID free induction decay. 27, 28, 39

FTICR Fourier transform ion cyclotron resonance. 21

FTICR-MS Fourier transform ion cyclotron resonance mass spectrometry. 28

FWHM full width at half maximum. 63

GA genetic algorithm. 54, 166, 167

GBA guilt-by-association. 115

GC gas chromatography. 21, 22

GC-EI-TOFMS gas chromatography—electron impact—time of flight mass spectrometry. 46

GE grammatical evolution. 168, 171

GP genetic programming. 166–168, 171, 174

GPCA generalized principal components analysis. 44, 45

GPLS-DA generalized partial least squares-discriminant analysis. 51

GRN gene regulatory network. 115

HCA hierarchical cluster analysis. 127, 129, 144

HPCA hierarchical principal components analysis. 46

HPLC high performance liquid chromatography. 22, 94

iPLS interval partial least squares. 51, 52

kNN k nearest neighbours. 54, 127

KOPLS kernel orthogonal PLS. 52

KPLS kernel PLS. 52

LC liquid chromatography. 21, 22, 94

LC-HRMS liquid chromatography–high resolution mass spectrometry. 94

LC-MS liquid chromatography–mass spectrometry. 21, 22, 24, 27, 29, 31, 35, 42, 46, 48, 49, 53, 54, 63–65, 68, 69, 98, 129, 146, 166

LDA linear discriminant analysis. 46, 48, 51

LN150 Leaf-negative peak #150, m/z = 279.0527967, rt = 2.059583333, identify unknown. 71

LOD limit of detection. 28

LOO leave-one-out. 58, 72

MAD median absolute deviation. 43

MANOVA multivariate analysis of variance. 41

MEND matched filtration with experimental noise determination. 29, 37

MetaboClust Software package. 138, 141, 143, 153, 159, 163, 165, 201–203, 206

MPA mobile phase A. 63, 94

MPB mobile phase B. 63, 94

MS mass spectrometry. 21–23, 33, 59, 63, 64, 94

MS-NRBF Microsoft .NET Remoting Binary Format. 143

MSCA multilevel simultaneous component analysis. 42

NB naive Bayes. 54

NIPALS nonlinear iterative partial least squares. 48

NIR near infra-red. 52

NMR nuclear magnetic resonance. 21–24, 27, 28, 31, 35, 39, 41, 42, 44, 45, 49, 51, 54, 56, 59, 129, 166

NPCA non-negative principal components analysis. 44

OPLS orthogonal partial least squares. 49, 50, 52

OSC orthogonal signal correction. 49, 50

PAGA peak alignment using a genetic algorithm. 32

PC principal component. 42–44, 48, 64

PCA principal components analysis. 24, 31, 35, 38, 39, 42–45, 48, 49, 59, 64, 66, 82, 85, 101, 129, 143, 146

PCC Pearson correlation coefficient. 71, 72, 116, 123, 126

PCR principal components regression. 48

PLF partial linear fit. 32

PLS partial least squares. 50–52

PLS-DA partial least squares discriminant analysis. 31, 48, 51, 101

PLSR partial least squares regression. 48, 52, 143

QC quality control. 38, 63, 64, 66, 94, 95

RCCA regularized canonical correlation analysis. 51, 52

RMSECV root mean squared error of cross-validation. 51

RMSEP root mean squared error of prediction. 72

ROI rectangular regions of interest. 35

RPLS regularised PLS. 51

RSD relative standard deviation. 99, 132, 189

RSPA recursive segment-wise peak alignment. 32

SGPCA sparse non-negative generalized principal components analysis. 44, 45

SMRS standard metabolic reporting structures. 58

SNR signal-to-noise ratio. 27

SPCA sparse principal components analysis. 82, 85

SPLS sparse partial least squares. 50

ST-GP strongly typed genetic programming. 171

STOCSY statistical total correlation spectroscopy. 55, 56

UML unified modelling language. 139

VIP variable importance in projection. 49, 72

WATERGATE water suppression by gradient tailored excitation. 27

XC-MS chromatography coupled mass spectrometry. 31

BIBLIOGRAPHY

[1] G. S. Fraenkel, "The raison d'etre of secondary plant substances these odd chemicals arose as a means of protecting plants from insects and now guide insects to food," *Science*, vol. 129, pp. 1466–1470, 1959.

[2] R. Roark, "Some promising insecticidal plants," *Economic Botany*, vol. 1, pp. 437–445, 1947.

[3] J. A. Nathanson, E. J. Hunnicutt, L. Kantham, and C. Scavone, "Cocaine as a naturally occurring insecticide," *Proceedings of the National Academy of Sciences*, vol. 90, pp. 9645–9648, 1993.

[4] K.-M. Oksman-Caldentey and K. Saito, "Integrating genomics and metabolomics for engineering plant metabolic pathways," *Current Opinion in Biotechnology*, vol. 16, pp. 174–179, 2005.

[5] O. Fiehn, "Combining genomics, metabolome analysis, and biochemical modelling to understand metabolic networks," *Comparative and Functional Genomics*, vol. 2, pp. 155–168, 2001.

[6] W. Filipowicz, S. N. Bhattacharyya, and N. Sonenberg, "Mechanisms of post-transcriptional regulation by microRNAs: are the answers in sight?" *Nature Reviews Genetics*, vol. 9, pp. 102–114, 2008.

[7] O. Fiehn, J. Kopka, P. Dörmann, T. Altmann, R. N. Trethewey, and L. Willmitzer, "Metabolite profiling for plant functional genomics," *Nature biotechnology*, vol. 18, pp. 1157–1161, 2000.

[8] A. R. Joyce and B. O. Palsson, "The model organism as a system: integrating 'omics' data sets," *Nature Reviews Molecular Cell Biology*, vol. 7, pp. 198–210, 2006.

[9] E. P. van Someren, L. Wessels, E. Backer, and M. Reinders, "Genetic network modeling," *Pharmacogenomics*, vol. 3, pp. 507–525, 2002.

[10] K. Saito and F. Matsuda, "Metabolomics for functional genomics, systems biology, and biotechnology," *Annual Review of Plant Biology*, vol. 61, pp. 463–489, 2010.

[11] G. Le Gall, I. J. Colquhoun, A. L. Davis, G. J. Collins, and M. E. Verhoeyen, "Metabolite profiling of tomato (*Lycopersicon esculentum*) using spectroscopy as a tool to detect potential unintended effects following a genetic modification," *Journal of Agricultural and Food Chemistry*, vol. 51, pp. 2447–2456, 2003.

[12] B. Médina, M. Salagoïty, F. Guyon, J. Gaye, P. Hubert, F. Guillaume, and P. Brereton, "Using new analytical approaches to verify the origin of wine," *New Analytical Approaches for Verifying the Origin of Food*, p. 149, 2013.

[13] A. J. Charlton, W. H. Farrington, and P. Brereton, "Application of ^{1}H NMR and multivariate statistics for screening complex mixtures: quality control and authenticity of instant coffee," *Journal of agricultural and food chemistry*, vol. 50, pp. 3098–3103, 2002.

[14] C. Ho, C. Lam, M. Chan, R. Cheung, L. Law, L. Lit, K. Ng, M. Suen, and H. Tai, "Electrospray ionisation mass spectrometry: principles and clinical applications," *Clinical Biochemist Reviews*, vol. 24, pp. 3–12, 2003.

[15] W. Paul, "Electromagnetic traps for charged and neutral particles," *Reviews of Modern Physics*, vol. 62, p. 531, 1990.

[16] A. Makarov, E. Denisov, O. Lange, and S. Horning, "Dynamic range of mass accuracy in LTQ orbitrap hybrid mass spectrometer," *Journal of the American Society for Mass Spectrometry*, vol. 17, pp. 977–982, 2006.

[17] A. G. Marshall, C. L. Hendrickson, and G. S. Jackson, "Fourier transform ion cyclotron resonance mass spectrometry: a primer," *Mass Spectrometry Reviews*, vol. 17, pp. 1–35, 1998.

[18] L. R. Snyder, J. J. Kirkland, and J. W. Dolan, *Introduction to modern liquid chromatography*. John Wiley & Sons, 2011.

[19] A. J. P. Martin and R. L. M. Synge, "Separation of the higher monoamino-acids by counter-current liquid-liquid extraction: the amino-acid composition of wool," *Biochemical Journal*, vol. 35, pp. 91–121, 1941, 16747393[pmid] Biochem J.

[20] J. C. Giddings, "Dynamics of chromatography," 1965.

[21] J. F. K. Huber, "High efficiency, high speed liquid chromatography in columns," *Journal of Chromatographic Science*, vol. 7, pp. 85–90, 1969.

[22] B. L. Karger, "HPLC: early and recent perspectives," *Journal of Chemical Education*, vol. 74, p. 45, 1997.

[23] E. M. Purcell, "Spontaneous emission probabilities at radio frequencies," *Physical Review*, vol. 69, p. 681, 1946.

[24] F. Bloch, W. Hansen, and M. Packard, "Nuclear induction," *Physical Review*, vol. 69, p. 127, 1946.

[25] E. Andrew and E. Szczesniak, "A historical account of NMR in the solid state," *Progress in Nuclear Magnetic Resonance Spectroscopy*, vol. 28, pp. 11–36, 1995.

[26] M. Saunders, A. Wishnia, and J. G. Kirkwood, "The nuclear magnetic resonance spectrum of ribonuclease1," *Journal of the American Chemical Society*, vol. 79, pp. 3289–3290, 1957.

[27] D. S. Wishart, "Metabolomics: applications to food science and nutrition research," *Trends in Food Science & Technology*, vol. 19, pp. 482–493, 2008.

[28] M. E. Bollard, E. G. Stanley, J. C. Lindon, J. K. Nicholson, and E. Holmes, "NMR-based metabonomic approaches for evaluating physiological influences on biofluid composition," *NMR in Biomedicine*, vol. 18, pp. 143–162, 2005.

[29] J. Nicholson, J. Lindon, and E. Holmes, "Metabonomics: understanding the metabolic responses of living systems to pathophysiological stimuli via multivariate statistical analysis of biological NMR spectroscopic data." *Xenobiotica*, vol. 29, pp. 1181–1189, 1999.

[30] W. B. Dunn, D. I. Broadhurst, H. J. Atherton, R. Goodacre, and J. L. Griffin, "Systems level studies of mammalian metabolomes: the roles of mass spectrometry and nuclear magnetic resonance spectroscopy," *Chemical Society Reviews*, vol. 40, pp. 387–426, 2011.

[31] M. Rusilowicz, S. O'Keefe, A. Charlton, and J. Wilson, *Chemometrics Applied to NMR Analysis*. John Wiley & Sons, Ltd, 2014.

[32] K. Pearson, "On lines and planes of closest fit to systems of points in space," *The London, Edinburgh, and Dublin Philosophical Magazine and Journal of Science*, vol. 2, pp. 559–572, 1901.

[33] H. Hotelling, "Analysis of a complex of statistical variables into principal components," *Journal of Educational Psychology*, vol. 24, p. 417, 1933.

[34] P. Geladi and K. Esbensen, "The start and early history of chemometrics: Selected interviews. part 1," *Journal of Chemometrics*, vol. 4, pp. 337–354, 1990.

[35] K. Esbensen and P. Geladi, "The start and early history of chemometrics: Selected interviews. part 2," *Journal of Chemometrics*, vol. 4, pp. 389–412, 1990.

[36] C. Reilly and B. Kowalski, "Nuclear magnetic resonance spectral interpretation by pattern recognition," *The Journal of Physical Chemistry*, vol. 75, pp. 1402–1411, 1971.

[37] R. Madsen, T. Lundstedt, and J. Trygg, "Chemometrics in metabolomics – a review in human disease diagnosis," *Analytica Chimica Acta*, vol. 659, pp. 23–33, 2010.

[38] L. Lin, Z. Huang, Y. Gao, X. Yan, J. Xing, and W. Hang, "LC-MS based serum metabonomic analysis for renal cell carcinoma diagnosis, staging, and biomarker discovery," *Journal of Proteome Research*, vol. 10, pp. 1396–1405, 2011.

[39] T. Gebregiworgis and R. Powers, "Application of NMR metabolomics to search for human disease biomarkers," *Combinatorial chemistry & high throughput screening*, vol. 15, pp. 595–610, 2012.

[40] Y. Levin, E. Schwarz, L. Wang, F. M. Leweke, and S. Bahn, "Label-free LC-MS/MS quantitative proteomics for large-scale biomarker discovery in complex samples," *Journal of Separation Science*, vol. 30, pp. 2198–2203, 2007.

[41] J. L. Griffin, "Metabonomics: NMR spectroscopy and pattern recognition analysis of body fluids and tissues for characterisation of xenobiotic toxicity and disease diagnosis," *Current Opinion in Chemical Biology*, vol. 7, pp. 648–654, 2003.

[42] D. J. Crockford, E. Holmes, J. C. Lindon, R. S. Plumb, S. Zirah, S. J. Bruce, P. Rainville, C. L. Stumpf, and J. K. Nicholson, "Statistical heterospectroscopy, an approach to the integrated analysis of NMR and UPLC-MS data sets: application in metabonomic toxicology studies," *Analytical chemistry*, vol. 78, pp. 363–371, 2006.

[43] A. A. Bletsou, J. Jeon, J. Hollender, E. Archontaki, and N. S. Thomaidis, "Targeted and non-targeted liquid chromatography-mass spectrometric workflows for identification of transformation products of emerging pollutants in the aquatic environment," *TrAC Trends in Analytical Chemistry*, vol. 66, pp. 32–44, 2015.

[44] G. Le Gall, M. Puaud, and I. J. Colquhoun, "Discrimination between orange juice and pulp wash by ^{1}H nuclear magnetic resonance spectroscopy: Identification of marker compounds," *Journal of Agricultural and Food Chemistry*, vol. 49, pp. 580–588, 2001.

[45] G. Le Gall, I. J. Colquhoun, and M. Defernez, "Metabolite profiling using ^{1}H NMR spectroscopy for quality assessment of green tea, *Camellia sinensis* (l.)," *Journal of Agricultural and Food Chemistry*, vol. 52, pp. 692–700, 2004.

[46] Y. Jung, J. Lee, J. Kwon, K.-S. Lee, D. H. Ryu, and G.-S. Hwang, "Discrimination of the geographical origin of beef by ^{1}H NMR-based metabolomics," *Journal of Agricultural and Food Chemistry*, vol. 58, pp. 10 458–10 466, 2010.

[47] C. V. Di Anibal, M. P. Callao, and I. Ruisánchez, "^{1}H NMR and UV-visible data fusion for determining Sudan dyes in culinary spices," *Talanta*, vol. 84, pp. 829–833, 2011.

[48] L. Jaitz, K. Siegl, R. Eder, G. Rak, L. Abranko, G. Koellensperger, and S. Hann, "LC-MS/MS analysis of phenols for classification of red wine according to geographic origin, grape variety and vintage," *Food Chemistry*, vol. 122, pp. 366–372, 2010.

[49] C. Albrecht, M. Stander, M. Grobbelaar, J. Colling, J. Kossmann, P. Hills, and N. Makunga, "LC-MS-based metabolomics assists with quality assessment and traceability of wild and cultivated plants of *Sutherlandia frutescens* (*Fabaceae*)," *South African Journal of Botany*, vol. 82, pp. 33–45, 2012.

[50] A. Charlton, T. Allnutt, S. Holmes, J. Chisholm, S. Bean, N. Ellis, P. Mullineaux, and S. Oehlschlager, "NMR profiling of transgenic peas," *Plant Biotechnology Journal*, vol. 2, pp. 27–35, 2004.

[51] Y. Chang, C. Zhao, Z. Zhu, Z. Wu, J. Zhou, Y. Zhao, X. Lu, and G. Xu, "Metabolic profiling based on LC/MS to evaluate unintended effects of transgenic rice with cry1Ac and sck genes," *Plant Molecular Biology*, vol. 78, pp. 477–487, 2012.

[52] W. Lorizio, A. H. Wu, M. S. Beattie, H. Rugo, S. Tchu, K. Kerlikowske, and E. Ziv, "Clinical and biomarker predictors of side effects from tamoxifen," *Breast Cancer Research and Treatment*, vol. 132, pp. 1107–1118, 2012.

[53] S. Y. Um, M. W. Chung, K.-B. Kim, S. H. Kim, J. S. Oh, H. Y. Oh, H. J. Lee, and K. H. Choi, "Pattern recognition analysis for the prediction of adverse effects by nonsteroidal anti-inflammatory drugs using ^{1}H NMR-based metabolomics in rats," *Analytical Chemistry*, vol. 81, pp. 4734–4741, 2009.

[54] L. M. Samuelsson, L. Förlin, G. Karlsson, M. Adolfsson-Erici, and D. J. Larsson, "Using NMR metabolomics to identify responses of an environmental estrogen in blood plasma of fish," *Aquatic Toxicology*, vol. 78, pp. 341–349, 2006.

[55] J. Lopes-Da-Silva, D. M. Santos, A. Freitas, C. Brites, and A. M. Gil, "Rheological and nuclear magnetic resonance (NMR) study of the hydration and heating of undeveloped wheat doughs," *Journal of Agricultural and Food Chemistry*, vol. 55, pp. 5636–5644, 2007.

[56] J. A. Westerhuis, T. Kourti, and J. F. MacGregor, "Comparing alternative approaches for multivariate statistical analysis of batch process data," *Journal of Chemometrics*, vol. 13, pp. 397–413, 1999.

[57] M. Sargent, "Guide to achieving reliable quantitative LC-MS measurements," *RSC Analytical Methods Committee, Teddington*, 2013.

[58] D. S. Wishart, L. M. Querengesser, B. A. Lefebvre, N. A. Epstein, R. Greiner, and J. B. Newton, "Magnetic resonance diagnostics: a new technology for high-throughput clinical diagnostics," *Clinical Chemistry*, vol. 47, pp. 1918–1921, 2001.

[59] A. M. Weljie, J. Newton, P. Mercier, E. Carlson, and C. M. Slupsky, "Targeted profiling: quantitative analysis of metabolomics data," *Analytical Chemistry*, vol. 78, pp. 4430–4442, 2006.

[60] M. Piotto, V. Saudek, and V. Sklenář, "Gradient-tailored excitation for single-quantum NMR spectroscopy of aqueous solutions," *Journal of Biomolecular NMR*, vol. 2, pp. 661–665, 1992.

[61] R. J. Ogg, R. Kingsley, and J. S. Taylor, "WET, a T_1- and B_1- insensitive water-suppression method for *in Vivo* localized spectroscopy," *Journal of Magnetic Resonance, Series B*, vol. 104, pp. 1–10, 1994.

[62] T.-L. Hwang and A. Shaka, "Water suppression that works. excitation sculpting using arbitrary wave-forms and pulsed-field gradients," *Journal of Magnetic Resonance, Series A*, vol. 112, pp. 275–279, 1995.

[63] P. Hors, "A new method for water suppression in the proton NMR spectra of aqueous solutions," *Journal of Magnetic Resonance (1969)*, vol. 54, pp. 539–542, 1983.

[64] D. Andrzejewski, J. A. Roach, M. L. Gay, and S. M. Musser, "Analysis of coffee for the presence of acrylamide by LC-MS/MS," *Journal of Agricultural and Food Chemistry*, vol. 52, pp. 1996–2002, 2004.

[65] H. Ono, Y. Chuda, M. Ohnishi-Kameyama, H. Yada, M. Ishizaka, H. Kobayashi, and M. Yoshida, "Analysis of acrylamide by LC-MS/MS and GC-MS in processed Japanese foods," *Food Additives & Contaminants*, vol. 20, pp. 215–220, 2003.

[66] W. D. Van Horn, A. J. Beel, C. Kang, and C. R. Sanders, "The impact of window functions on NMR-based paramagnetic relaxation enhancement measurements in membrane proteins," *Biochimica et Biophysica Acta (BBA)-Biomembranes*, vol. 1798, pp. 140–149, 2010.

[67] P. Giraudeau and S. Akoka, "Sensitivity and lineshape improvement in ultrafast 2D NMR by optimized apodization in the spatially encoded dimension," *Magnetic Resonance in Chemistry*, vol. 49, pp. 307–313, 2011.

[68] M. K. Titulaer, D. A. Mustafa, I. Siccama, M. Konijnenburg, P. C. Burgers, A. C. Andeweg, P. Smitt, J. M. Kros, and T. M. Luider, "A software application for comparing large numbers of high resolution MALDI-FTICR MS spectra demonstrated by searching candidate biomarkers for glioma blood vessel formation," *BMC Bioinformatics*, vol. 9, p. 1, 2008.

[69] L. Zhang, Y. Zhang, S. Zhao, K. H. Chung, C. Xu, and Q. Shi, "Effect of apodization on FT-ICR mass spectrometry analysis of petroleum," *International Journal of Mass Spectrometry*, vol. 373, pp. 27–33, 2014.

[70] C. W. R. Greg Wells, Harry Prest, "Signal, noise, and detection limits in mass spectrometry," 2011. [Online]. Available: https://www.agilent.com/cs/library/technicaloverviews/Public/5990-7651EN.pdf

[71] A. E. Derome, "Modern NMR techniques for chemistry research," 1987.

[72] S. Halouska and R. Powers, "Negative impact of noise on the principal component analysis of NMR data," *Journal of Magnetic Resonance*, vol. 178, pp. 88–95, 2006.

[73] B. Walczak and D. Massart, "Noise suppression and signal compression using the wavelet packet transform," *Chemometrics and Intelligent Laboratory Systems*, vol. 36, pp. 81–94, 1997.

[74] C. Zheng and Y. Zhang, "Low-field pulsed NMR signal denoising based on wavelet transform," in *Signal Processing and Communications Applications, 2007. SIU 2007. IEEE 15th*, Conference Proceedings, pp. 1–4.

[75] D. L. Donoho and J. M. Johnstone, "Ideal spatial adaptation by wavelet shrinkage," *Biometrika*, vol. 81, pp. 425–455, 1994.

[76] V. P. Andreev, T. Rejtar, H.-S. Chen, E. V. Moskovets, A. R. Ivanov, and B. L. Karger, "A universal denoising and peak picking algorithm for LC-MS based on matched filtration in the chromatographic time domain," *Analytical chemistry*, vol. 75, pp. 6314–6326, 2003.

[77] C. Tang, "An analysis of baseline distortion and offset in NMR spectra," *Journal of Magnetic Resonance, Series A*, vol. 109, pp. 232–240, 1994.

[78] D. G. Davis, "Elimination of baseline distortions and minimization of artifacts from phased 2D NMR spectra," *Journal of Magnetic Resonance*, vol. 81, pp. 603 – 607, 1969.

[79] A. Heuer and U. Haeberlen, "A new method for suppressing baseline distortions in FT NMR," *Journal of Magnetic Resonance (1969)*, vol. 85, pp. 79–94, 1989.

[80] P. Güntert and K. Wüthrich, "FLATT – a new procedure for high-quality baseline correction of multidimensional NMR spectra," *Journal of Magnetic Resonance (1969)*, vol. 96, pp. 403–407, 1992.

[81] W. Dietrich, C. H. Rü, del, and M. Neumann, "Fast and precise automatic baseline correction of one-and two-dimensional NMR spectra," *Journal of Magnetic Resonance (1969)*, vol. 91, pp. 1–11, 1991.

[82] J. C. Cobas, M. A. Bernstein, M. Martín-Pastor, and P. G. Tahoces, "A new general-purpose fully automatic baseline-correction procedure for 1d and 2d NMR data," *Journal of Magnetic Resonance*, vol. 183, pp. 145–151, 2006.

[83] S. Golotvin and A. Williams, "Improved baseline recognition and modeling of FT NMR spectra," *Journal of Magnetic Resonance*, vol. 146, pp. 122–125, 2000.

[84] D. Chang, C. D. Banack, and S. L. Shah, "Robust baseline correction algorithm for signal dense NMR spectra," *Journal of Magnetic Resonance*, vol. 187, pp. 288–292, 2007.

[85] Y. Xi and D. Rocke, "Baseline correction for NMR spectroscopic metabolomics data analysis," *BMC Bioinformatics*, vol. 9, p. 324, 2008.

[86] Q. Bao, J. Feng, F. Chen, W. Mao, Z. Liu, K. Liu, and C. Liu, "A new automatic baseline correction method based on iterative method," *Journal of Magnetic Resonance*, vol. 218, pp. 35–43, 2012.

[87] J. T. Pearce, T. J. Athersuch, T. M. Ebbels, J. C. Lindon, J. K. Nicholson, and H. C. Keun, "Robust algorithms for automated chemical shift calibration of 1D ^{1}H NMR spectra of blood serum," *Analytical Chemistry*, vol. 80, pp. 7158–7162, 2008.

[88] A. Prakash, B. Piening, J. Whiteaker, H. Zhang, S. A. Shaffer, D. Martin, L. Hohmann, K. Cooke, J. M. Olson, and S. Hansen, "Assessing bias in experiment design for large scale mass spectrometry-based quantitative proteomics," *Molecular & Cellular Proteomics*, vol. 6, pp. 1741–1748, 2007.

[89] M. Defernez and I. J. Colquhoun, "Factors affecting the robustness of metabolite fingerprinting using spectra," *Phytochemistry*, vol. 62, pp. 1009–1017, 2003.

[90] K. A. Veselkov, J. C. Lindon, T. M. Ebbels, D. Crockford, V. V. Volynkin, E. Holmes, D. B. Davies, and J. K. Nicholson, "Recursive segment-wise peak alignment of biological ^{1}H NMR spectra for improved metabolic biomarker recovery," *Analytical Chemistry*, vol. 81, pp. 56–66, 2008.

[91] A. Kassidas, J. F. MacGregor, and P. A. Taylor, "Synchronization of batch trajectories using dynamic time warping," *AIChE Journal*, vol. 44, pp. 864–875, 1998.

[92] N.-P. V. Nielsen, J. M. Carstensen, and J. Smedsgaard, "Aligning of single and multiple wavelength chromatographic profiles for chemometric data analysis using correlation optimised warping," *Journal of Chromatography A*, vol. 805, pp. 17–35, 1998.

[93] V. Pravdova, B. Walczak, and D. Massart, "A comparison of two algorithms for warping of analytical signals," *Analytica Chimica Acta*, vol. 456, pp. 77–92, 2002.

[94] R. J. Torgrip, M. Å, berg, B. Karlberg, and S. P. Jacobsson, "Peak alignment using reduced set mapping," *Journal of Chemometrics*, vol. 17, pp. 573–582, 2003.

[95] C. Christin, A. K. Smilde, H. C. Hoefsloot, F. Suits, R. Bischoff, and P. L. Horvatovich, "Optimized time alignment algorithm for LC-MS data: Correlation optimized warping using component detection algorithm-selected mass chromatograms," *Analytical chemistry*, vol. 80, pp. 7012–7021, 2008.

[96] S. B. Kim, Z. Wang, and C. M. Duran, "A Bayesian approach for the alignment of high-resolution NMR spectra," in *Proceedings of the INFORMS Artificial Intelligence and Data Mining Workshop, Pittsburgh, PA, USA*, Conference Proceedings, pp. 1–6.

[97] J. Vogels, A. Tas, J. Venekamp, and J. Van Der Greef, "Partial linear fit: A new NMR spectroscopy preprocessing tool for pattern recognition applications," *Journal of Chemometrics*, vol. 10, pp. 425–438, 1996.

[98] N. MacKinnon, W. Ge, A. P. Khan, B. S. Somashekar, P. Tripathi, J. Siddiqui, J. T. Wei, A. M. Chinnaiyan, T. M. Rajendiran, and A. Ramamoorthy, "Variable reference alignment: An improved peak alignment protocol for NMR spectral data with large intersample variation," *Analytical Chemistry*, vol. 84, pp. 5372–5379, 2012.

[99] J. Forshed, I. Schuppe-Koistinen, and S. P. Jacobsson, "Peak alignment of NMR signals by means of a genetic algorithm," *Analytica Chimica Acta*, vol. 487, pp. 189–199, 2003.

[100] J. W. Wong, C. Durante, and H. M. Cartwright, "Application of fast Fourier transform cross-correlation for the alignment of large chromatographic and spectral datasets," *Analytical Chemistry*, vol. 77, pp. 5655–5661, 2005.

[101] F. Savorani, G. Tomasi, and S. B. Engelsen, "icoshift: A versatile tool for the rapid alignment of 1D NMR spectra," *Journal of Magnetic Resonance*, vol. 202, pp. 190–202, 2010.

[102] W. Zhang, Z. Lei, D. Huhman, L. W. Sumner, and P. X. Zhao, "MET-XAlign: a metabolite cross-alignment tool for LC/MS-based comparative metabolomics," *Analytical Chemistry*, vol. 87, pp. 9114–9119, 2015.

[103] T. N. Vu and K. Laukens, "Getting your peaks in line: a review of alignment methods for NMR spectral data," *Metabolites*, vol. 3, pp. 259–276, 2013.

[104] O. Cloarec, M. E. Dumas, J. Trygg, A. Craig, R. H. Barton, J. C. Lindon, J. K. Nicholson, and E. Holmes, "Evaluation of the orthogonal projection on latent structure model limitations caused by chemical shift variability and improved visualization of biomarker changes in spectroscopic metabonomic studies," *Analytical Chemistry*, vol. 77, pp. 517–526, 2005.

[105] G. F. Giskeødegård, T. G. Bloemberg, G. Postma, B. Sitter, M.-B. Tessem, I. S. Gribbestad, T. F. Bathen, and L. M. Buydens, "Alignment of high resolution magic angle spinning magnetic resonance spectra using warping methods," *Analytica Chimica Acta*, vol. 683, pp. 1–11, 2010.

[106] T.-H. Tsai, M. G. Tadesse, C. Di Poto, L. K. Pannell, Y. Mechref, Y. Wang, and H. W. Ressom, "Multi-profile Bayesian alignment model for LC-MS data analysis with integration of internal standards," *Bioinformatics*, vol. 29, no. 21, pp. 2774–2780, 2013.

[107] B. Fischer, J. Grossmann, V. Roth, W. Gruissem, S. Baginsky, and J. M. Buhmann, "Semi-supervised LC/MS alignment for differential proteomics," *Bioinformatics*, vol. 22, no. 14, pp. e132–e140, 2006.

[108] R. Smith, D. Ventura, and J. T. Prince, "LC-MS alignment in theory and practice: a comprehensive algorithmic review," *Briefings in Bioinformatics*, vol. 16, pp. 104–117, 2015.

[109] O. Beckonert, M. E Bollard, T. Ebbels, H. C. Keun, H. Antti, E. Holmes, J. C. Lindon, and J. K. Nicholson, "NMR-based metabonomic toxicity classification: hierarchical cluster analysis and k-nearest-neighbour approaches," *Analytica Chimica Acta*, vol. 490, pp. 3–15, 2003.

[110] J. Forshed, R. J. Torgrip, K. M. Åberg, B. Karlberg, J. Lindberg, and S. P. Jacobsson, "A comparison of methods for alignment of NMR peaks in the context of cluster analysis," *Journal of Pharmaceutical and Biomedical Analysis*, vol. 38, pp. 824–832, 2005.

[111] R. A. Davis, A. J. Charlton, J. Godward, S. A. Jones, M. Harrison, and J. C. Wilson, "Adaptive binning: An improved binning method for metabolomics data using the undecimated wavelet transform," *Chemometrics and Intelligent Laboratory Systems*, vol. 85, pp. 144–154, 2007.

[112] T. De Meyer, D. Sinnaeve, B. Van Gasse, E. Tsiporkova, E. R. Rietzschel, M. L. De Buyzere, T. C. Gillebert, S. Bekaert, J. C. Martins, and W. Van Criekinge, "NMR-based characterization of metabolic alterations in hypertension using an adaptive, intelligent binning algorithm," *Analytical Chemistry*, vol. 80, pp. 3783–3790, 2008.

[113] P. E. Anderson, N. V. Reo, N. J. DelRaso, T. E. Doom, and M. L. Raymer, "Gaussian binning: a new kernel-based method for processing NMR spectroscopic data for metabolomics," *Metabolomics*, vol. 4, pp. 261–272, 2008.

[114] N. Trbovic, F. Dancea, T. Langer, and U. Gü, nther, "Using wavelet de-noised spectra in NMR screening," *Journal of Magnetic Resonance*, vol. 173, pp. 280–287, 2005.

[115] C. D. DeHaven, A. M. Evans, H. Dai, and K. A. Lawton, "Organization of GC/MS and LC/MS metabolomics data into chemical libraries," *Journal of cheminformatics*, vol. 2, p. 9, 2010.

[116] R. Tautenhahn, C. Böttcher, and S. Neumann, "Highly sensitive feature detection for high resolution LC/MS," *BMC Bioinformatics*, vol. 9, p. 1, 2008.

[117] I. A. Lewis, S. C. Schommer, and J. L. Markley, "rNMR: open source software for identifying and quantifying metabolites in NMR spectra," *Magnetic Resonance in Chemistry*, vol. 47, p. S123, 2009.

[118] H.-W. Koh, S. Maddula, J. Lambert, R. Hergenröder, and L. Hildebrand, "An approach to automated frequency-domain feature

extraction in nuclear magnetic resonance spectroscopy," *Journal of Magnetic Resonance*, vol. 201, pp. 146–156, 2009.

[119] J. S. McKenzie, A. J. Charlton, J. A. Donarski, A. D. MacNicoll, and J. C. Wilson, "Peak fitting in 2D ^{1}h-^{13}c HSQC NMR spectra for metabolomic studies," *Metabolomics*, vol. 6, pp. 574–582, 2010.

[120] Y. Xi, J. S. de Ropp, M. R. Viant, D. L. Woodruff, and P. Yu, "Automated screening for metabolites in complex mixtures using 2D COSY NMR spectroscopy," *Metabolomics*, vol. 2, pp. 221–233, 2006.

[121] Y. Xi, J. S. de Ropp, M. R. Viant, D. L. Woodruff, and P. Yu, "Improved identification of metabolites in complex mixtures using HSQC NMR spectroscopy," *Analytica Chimica Acta*, vol. 614, pp. 127–133, 2008.

[122] A. Smolinska, L. Blanchet, L. Buydens, and S. S. Wijmenga, "NMR and pattern recognition methods in metabolomics: from data acquisition to biomarker discovery: a review," *Analytica Chimica Acta*, vol. 750, pp. 82–97, 2012.

[123] D. M. Horn, R. A. Zubarev, and F. W. McLafferty, "Automated reduction and interpretation of high resolution electrospray mass spectra of large molecules," *Journal of the American Society for Mass Spectrometry*, vol. 11, pp. 320–332, 2000.

[124] C. A. Smith, E. J. Want, G. O'Maille, R. Abagyan, and G. Siuzdak, "XCMS: processing mass spectrometry data for metabolite profiling using nonlinear peak alignment, matching, and identification," *Analytical Chemistry*, vol. 78, pp. 779–787, 2006.

[125] M. Katajamaa, J. Miettinen, and M. Orešič, "Mzmine: toolbox for processing and visualization of mass spectrometry based molecular profile data," *Bioinformatics*, vol. 22, pp. 634–636, 2006.

[126] A. Lommen, "MetAlign: interface-driven, versatile metabolomics tool for hyphenated full-scan mass spectrometry data preprocessing," *Analytical Chemistry*, vol. 81, pp. 3079–3086, 2009.

[127] N. Dynamics, "Progenesis QI," 2017. [Online]. Available: http://www.nonlinear.com/progenesis/qi/

[128] B. A. Ejigu, D. Valkenborg, G. Baggerman, M. Vanaerschot, E. Witters, J.-C. Dujardin, T. Burzykowski, and M. Berg, "Evaluation of normalization methods to pave the way towards large-scale LC-MS-based metabolomics profiling experiments," *Omics: a journal of integrative biology*, vol. 17, pp. 473–485, 2013.

[129] A. Craig, O. Cloarec, E. Holmes, J. K. Nicholson, and J. C. Lindon, "Scaling and normalization effects in NMR spectroscopic metabonomic data sets," *Analytical Chemistry*, vol. 78, pp. 2262–2267, 2008.

[130] C. Ludwig and M. R. Viant, "Two-dimensional j-resolved NMR spectroscopy: review of a key methodology in the metabolomics toolbox," *Phytochemical Analysis*, vol. 21, pp. 22–32, 2010.

[131] F. Dieterle, A. Ross, t. Schlotterbeck, Gö, and H. Senn, "Probabilistic quotient normalization as robust method to account for dilution of complex biological mixtures. application in metabonomics," *Analytical Chemistry*, vol. 78, pp. 4281–4290, 2006.

[132] R. Torgrip, K. Åberg, E. Alm, I. Schuppe-Koistinen, and J. Lindberg, "A note on normalization of biofluid 1D ^{1}h-nmr data," *Metabolomics*, vol. 4, pp. 114–121, 2008.

[133] J. Dong, K.-K. Cheng, J. Xu, Z. Chen, and J. L. Griffin, "Group aggregating normalization method for the preprocessing of NMR-based metabolomic data," *Chemometrics and Intelligent Laboratory Systems*, vol. 108, pp. 123–132, 2011.

[134] E. Zelena, W. B. Dunn, D. Broadhurst, S. Francis-McIntyre, K. M. Carroll, P. Begley, S. O'Hagan, J. D. Knowles, A. Halsall, and I. D. Wilson, "Development of a robust and repeatable UPLC-MS method for the long-term metabolomic study of human serum," *Analytical Chemistry*, vol. 81, pp. 1357–1364, 2009.

[135] R. A. van den Berg, H. C. Hoefsloot, J. A. Westerhuis, A. K. Smilde, and t. J. van der Werf, Marië, "Centering, scaling, and transformations: improving the biological information content of metabolomics data," *BMC Genomics*, vol. 7, p. 142, 2006.

[136] H. C. Keun, T. M. Ebbels, H. Antti, M. E. Bollard, O. Beckonert, E. Holmes, J. C. Lindon, and J. K. Nicholson, "Improved analysis of multivariate data by variable stability scaling: application to NMR-based metabolic profiling," *Analytica Chimica Acta*, vol. 490, pp. 265–276, 2003.

[137] P. V. Purohit, D. M. Rocke, M. R. Viant, and D. L. Woodruff, "Discrimination models using variance-stabilizing transformation of metabolomic NMR data," *Omics: a Journal of Integrative Biology*, vol. 8, pp. 118–130, 2004.

[138] H. M. Parsons, C. Ludwig, U. L. Günther, and M. R. Viant, "Improved classification accuracy in 1-and 2-dimensional NMR metabolomics data using the variance stabilising generalised logarithm transformation," *BMC Bioinformatics*, vol. 8, p. 234, 2007.

[139] R. Stoyanova, A. W. Nicholls, J. K. Nicholson, J. C. Lindon, and T. R. Brown, "Automatic alignment of individual peaks in large high-resolution spectral data sets," *Journal of Magnetic Resonance*, vol. 170, pp. 329–335, 2004.

[140] M. Bland, *An introduction to medical statistics*. Oxford University Press (UK), 2015.

[141] D. G. Smith, J. Clemens, W. Crede, M. Harvey, and E. J. Gracely, "Impact of multiple comparisons in randomized clinical trials," *The American Journal of Medicine*, vol. 83, pp. 545–550, 1987.

[142] J. M. Bland and D. G. Altman, "Multiple significance tests: the Bonferroni method," *BMJ*, vol. 310, p. 170, 1995.

[143] Y. Benjamini and Y. Hochberg, "Controlling the false discovery rate: a practical and powerful approach to multiple testing," *Journal of the Royal Statistical Society. Series B (Methodological)*, pp. 289–300, 1995.

[144] J. Klepacki, J. Klawitter, J. Klawitter, A. Karimpour-fard, J. Thurman, G. Ingle, D. Patel, and U. Christians, "Amino acids in a targeted versus a non-targeted metabolomics LC-MS/MS assay. are the results consistent?" *Clinical Biochemistry*, 2016.

[145] K. Verwaest, *^{1}H NMR Based Metabolomics Studies of Biofluids and Brain Extracts from Animal Models for Neurodegenerative Diseases: Proefschrift.* Universiteit Antwerpen, Faculteit Wetenschappen, Departement Chemie, 2011.

[146] E. Nevedomskaya, T. Pacchiarotta, A. Artemov, A. Meissner, C. van Nieuwkoop, J. T. van Dissel, O. A. Mayboroda, and M. Deelder, André, "^{1}H-NMR-based metabolic profiling of urinary tract infection: combining multiple statistical models and clinical data," *Metabolomics*, vol. 8, pp. 1227–1235, 2012.

[147] L. Bala, U. C. Ghoshal, U. Ghoshal, P. Tripathi, A. Misra, G. Gowda, and C. Khetrapal, "Malabsorption syndrome with and without small intestinal bacterial overgrowth: A study on upper-gut aspirate using ^{1}H NMR spectroscopy," *Magnetic resonance in medicine*, vol. 56, pp. 738–744, 2006.

[148] M. R. Pears, D. Rubtsov, H. M. Mitchison, J. D. Cooper, D. A. Pearce, R. J. Mortishire-Smith, and J. L. Griffin, "Strategies for data analyses in a high resolution ^{1}H NMR based metabolomics study of a mouse model of Batten disease," *Metabolomics*, vol. 3, pp. 121–136, 2007.

[149] V.-P. Mäkinen, P. Soininen, C. Forsblom, M. Parkkonen, P. Ingman, K. Kaski, P.-H. Groop, M. Ala-Korpela, and F. S. Group, "Diagnosing diabetic nephropathy by ^{1}H NMR metabonomics of serum," *Magnetic Resonance Materials in Physics, Biology and Medicine*, vol. 19, pp. 281–296, 2006.

[150] T. De Meyer, D. Sinnaeve, B. Van Gasse, E.-R. Rietzschel, M. L. De Buyzere, M. R. Langlois, S. Bekaert, J. C. Martins, and W. Van Criekinge, "Evaluation of standard and advanced preprocessing methods for the univariate analysis of blood serum ^{1}H-NMR spectra," *Analytical and Bioanalytical Chemistry*, vol. 398, pp. 1781–1790, 2010.

[151] J. T. Leek and J. D. Storey, "A general framework for multiple testing dependence," *Proceedings of the National Academy of Sciences*, vol. 105, pp. 18 718–18 723, 2008.

[152] L. Stahle and S. Wold, "Multivariate analysis of variance (manova)," *Chemometrics and Intelligent Laboratory Systems*, vol. 9, pp. 127–141, 1990.

[153] A. K. Smilde, J. J. Jansen, H. C. Hoefsloot, R.-J. A. Lamers, J. Van Der Greef, and M. E. Timmerman, "Anova-simultaneous component analysis (asca): a new tool for analyzing designed metabolomics data," *Bioinformatics*, vol. 21, pp. 3043–3048, 2005.

[154] N. Cañ, ellas, R. Solà, Alberich, J. Brezmes, R. Mallol, R.-M. Valls, M. A. Rodrí, guez, M. Vinaixa, A. Anguera, and X. Correig, "Use of multivariate chemometric algorithms on data to assess a soluble fiber (plantago ovata husk) nutritional intervention," *Chemometrics and Intelligent Laboratory Systems*, vol. 121, pp. 1–8, 2013.

[155] D. Rago, K. Mette, G. Gürdeniz, F. Marini, M. Poulsen, and L. O. Dragsted, "A LC-MS metabolomics approach to investigate the effect of raw apple intake in the rat plasma metabolome," *Metabolomics*, vol. 9, pp. 1202–1215, 2013.

[156] M. Vinaixa, M. A. Rodriguez, S. Samino, M. Díaz, A. Beltran, R. Mallol, C. Bladé, L. Ibañez, X. Correig, and O. Yanes, "Metabolomics reveals reduction of metabolic oxidation in women with polycystic ovary syndrome after pioglitazone-flutamide-metformin polytherapy," *PloS One*, vol. 6, p. e29052, 2011.

[157] A. Lemanska, M. Grootveld, C. J. Silwood, and R. G. Brereton, "Chemometric variance analysis of ^{1}H NMR metabolomics data on the effects of oral rinse on saliva," *Metabolomics*, vol. 8, pp. 64–80, 2012.

[158] B. Worley and R. Powers, "Multivariate analysis in metabolomics," *Curr Metabolomics*, vol. 1, pp. 92–107, 2013, worley, Bradley Powers, Robert Curr Metabolomics. 2013;1(1):92-107. doi:10.2174/2213235X11301010092.

[159] I. Jolliffe, *Principal component analysis*. Wiley Online Library, 2002.

[160] M. Daszykowski, K. Kaczmarek, Y. Vander Heyden, and B. Walczak, "Robust statistics in data analysis – a review: basic concepts," *Chemometrics and intelligent laboratory systems*, vol. 85, pp. 203–219, 2007.

[161] C. Croux and A. Ruiz-Gazen, "A fast algorithm for robust principal components based on projection pursuit," in *Compstat*. Springer, Conference Proceedings, pp. 211–216.

[162] C. Croux, P. Filzmoser, and M. R. Oliveira, "Algorithms for projection–pursuit robust principal component analysis," *Chemometrics and Intelligent Laboratory Systems*, vol. 87, pp. 218–225, 2007.

[163] H. Xu, C. Caramanis, and S. Sanghavi, "Robust PCA via outlier pursuit," *IEEE Transactions on Information Theory*, vol. 58, pp. 3047–3064, 2012.

[164] I. T. Jolliffe, N. T. Trendafilov, and M. Uddin, "A modified principal component technique based on the lasso," *Journal of Computational and Graphical Statistics*, vol. 12, pp. 531–547, 2003.

[165] H. Zou, T. Hastie, and R. Tibshirani, "Sparse principal component analysis," *Journal of Computational and Graphical Statistics*, vol. 15, pp. 265–286, 2006.

[166] G. I. Allen, L. Grosenick, and J. Taylor, "A generalized least squares matrix decomposition," 2014.

[167] P. Sajda, S. Du, T. R. Brown, R. Stoyanova, D. C. Shungu, X. Mao, and L. C. Parra, "Nonnegative matrix factorization for rapid recovery of constituent spectra in magnetic resonance chemical shift imaging of the brain," *Medical Imaging, IEEE Transactions on*, vol. 23, pp. 1453–1465, 2004.

[168] L. Deng, K.-K. Cheng, J. Dong, J. L. Griffin, and Z. Chen, "Non-negative principal component analysis for NMR-based metabolomic data analysis," *Chemometrics and Intelligent Laboratory Systems*, 2012.

[169] G. I. Allen and M. Maletić, Savatić, "Sparse non-negative generalized pca with applications to metabolomics," *Bioinformatics*, vol. 27, pp. 3029–3035, 2011.

[170] R. Boqué and A. K. Smilde, "Monitoring and diagnosing batch processes with multiway covariates regression models," *AIChE Journal*, vol. 45, pp. 1504–1520, 1999.

[171] H. T. Pedersen, M. Dyrby, S. B. Engelsen, and R. Bro, "Application of multi-way analysis to 2D NMR data," *Annual reports on NMR spectroscopy*, vol. 59, pp. 207–233, 2006.

[172] I. Montoliu, F.-P. J. Martin, S. Collino, S. Rezzi, and S. Kochhar, "Multivariate modeling strategy for intercompartmental analysis of tissue and plasma ^{1}H NMR spectrotypes," *Journal of Proteome Research*, vol. 8, pp. 2397–2406, 2009.

[173] J. Forshed, H. Idborg, and S. P. Jacobsson, "Evaluation of different techniques for data fusion of LC/MS and ^{1}H-NMR," *Chemometrics and Intelligent Laboratory Systems*, vol. 85, pp. 102–109, 2007.

[174] B. Biais, J. W. Allwood, C. Deborde, Y. Xu, M. Maucourt, B. Beauvoit, W. B. Dunn, D. Jacob, R. Goodacre, D. Rolin *et al.*, "GC-EI-TOFMS, and data set correlation for fruit metabolomics: Application to spatial metabolite analysis in melon," *Analytical chemistry*, vol. 81, pp. 2884–2894, 2009.

[175] S. Wold, "Pattern recognition by means of disjoint principal components models," *Pattern Recognition*, vol. 8, pp. 127–139, 1976.

[176] C. H. Park and H. Park, "A comparison of generalized linear discriminant analysis algorithms," *Pattern Recognition*, vol. 41, pp. 1083–1097, 2008.

[177] S. Rezzi, D. E. Axelson, K. Héberger, F. Reniero, C. Mariani, and C. Guillou, "Classification of olive oils using high throughput flow ^{1}H NMR fingerprinting with principal component analysis, linear discriminant analysis and probabilistic neural networks," *Analytica Chimica Acta*, vol. 552, pp. 13–24, 2005.

[178] M. A. Alvarez-Fernandez, A. B. Cerezo, A. M. Canete-Rodriguez, A. M. Troncoso, and M. C. Garcia-Parrilla, "Composition of non-anthocyanin polyphenols in alcoholic-fermented strawberry products using LC-MS (QTRAP), high-resolution MS (UHPLC-orbitrap-MS), LC-DAD, and antioxidant activity," *Journal of agricultural and food chemistry*, vol. 63, pp. 2041–2051, 2015.

[179] C. Fauhl, F. Reniero, and C. Guillou, "^{1}H NMR as a tool for the analysis of mixtures of virgin olive oil with oils of different botanical origin," *Magnetic Resonance in Chemistry*, vol. 38, pp. 436–443, 2000.

[180] O. Beckonert, J. Monnerjahn, U. Bonk, and D. Leibfritz, "Visualizing metabolic changes in breast-cancer tissue using 1h-nmr spectroscopy and self-organizing maps," *NMR in Biomedicine*, vol. 16, pp. 1–11, 2003.

[181] M. Barker and W. Rayens, "Partial least squares for discrimination," *Journal of chemometrics*, vol. 17, pp. 166–173, 2003.

[182] H. Wold, *Estimation of Principal Components and Related Models by Iterative Least squares*. New York: Academic Press, 1966, pp. 391–420.

[183] S. Wold, M. Sjöström, and L. Eriksson, "PLS-regression: a basic tool of chemometrics," *Chemometrics and Intelligent Laboratory Systems*, vol. 58, pp. 109 – 130, 2001.

[184] I.-G. Chong and C.-H. Jun, "Performance of some variable selection methods when multicollinearity is present," *Chemometrics and Intelligent Laboratory Systems*, vol. 78, pp. 103–112, 2005.

[185] X. Shao and Y. Li, "Classification and prediction by LF NMR," *Food and Bioprocess Technology*, vol. 5, pp. 1817–1823, 2012.

[186] K. Skov, N. Hadrup, J. Smedsgaard, and H. Frandsen, "LC-MS analysis of the plasma metabolome—a novel sample preparation strategy," *Journal of Chromatography B*, vol. 978, pp. 83–88, 2015.

[187] J. Trygg and S. Wold, "Orthogonal projections to latent structures (O-PLS)," *Journal of Chemometrics*, vol. 16, pp. 119–128, 2002.

[188] H. Wu, X. Li, X. Yan, L. An, K. Luo, M. Shao, Y. Jiang, R. Xie, and F. Feng, "An untargeted metabolomics-driven approach based on LC-TOF/MS and LC-MS/MS for the screening of xenobiotics and metabolites of zhi-zi-da-huang decoction in rat plasma," *Journal of Pharmaceutical and Biomedical Analysis*, vol. 115, pp. 315–322, 2015.

[189] J. Trygg, "O2-PLS for qualitative and quantitative analysis in multivariate calibration," *Journal of Chemometrics*, vol. 16, pp. 283–293, 2002.

[190] G. M. Kirwan, T. Hancock, K. Hassell, J. O. Niere, D. Nugegoda, S. Goto, and M. J. Adams, "NMR metabonomic profiling using tO2PLS," *Analytica Chimica Acta*, 2013.

[191] H. S. Tapp and E. K. Kemsley, "Notes on the practical utility of OPLS," *TrAC Trends in Analytical Chemistry*, vol. 28, pp. 1322–1327, 2009.

[192] K.-A. Lê Cao, D. Rossouw, C. Robert-Granié, and P. Besse, "A sparse pls for variable selection when integrating omics data," *Statistical applications in genetics and molecular biology*, vol. 7, 2008.

[193] M. Jiang, C. Wang, Y. Zhang, Y. Feng, Y. Wang, and Y. Zhu, "Sparse partial-least-squares discriminant analysis for different geographical origins of *Salvia miltiorrhiza* by ^{1}H-NMR-based metabolomics," *Phytochemical Analysis*, vol. 25, pp. 50–58, 2014.

[194] R. Sabatier, M. Vivien, and P. Amenta, *Two approaches for discriminant partial least squares.* Springer, 2003, pp. 100–108.

[195] G. I. Allen, C. Peterson, M. Vannucci, and M. Maletić, Savatić, "Regularized partial least squares with an application to NMR spectroscopy," *Statistical Analysis and Data Mining*, vol. 6, pp. 302–314, 2013.

[196] L. Norgaard, A. Saudland, J. Wagner, J. P. Nielsen, L. Munck, and S. Engelsen, "Interval partial least-squares regression (iPLS): a comparative chemometric study with an example from near-infrared spectroscopy," *Applied Spectroscopy*, vol. 54, pp. 413–419, 2000.

[197] R. B. Darlington, S. L. Weinberg, and H. J. Walberg, "Canonical variate analysis and related techniques," *Review of Educational Research*, vol. 43, pp. 433–454, 1973.

[198] H. Yamamoto, H. Yamaji, E. Fukusaki, H. Ohno, and H. Fukuda, "Canonical correlation analysis for multivariate regression and its applic-

ation to metabolic fingerprinting," *Biochemical Engineering Journal*, vol. 40, pp. 199–204, 2008.

[199] L. Norgaard, R. Bro, F. Westad, and S. B. Engelsen, "A modification of canonical variates analysis to handle highly collinear multivariate data," *Journal of Chemometrics*, vol. 20, pp. 425–435, 2006.

[200] L. Norgaard, G. Söletormos, N. Harrit, M. Albrechtsen, O. Olsen, D. Nielsen, K. Kampmann, and R. Bro, "Fluorescence spectroscopy and chemometrics for classification of breast cancer samples – a feasibility study using extended canonical variates analysis," *Journal of Chemometrics*, vol. 21, pp. 451–458, 2007.

[201] L. Munck, "A new holistic exploratory approach to systems biology by near infrared spectroscopy evaluated by chemometrics and data inspection," *Journal of Chemometrics*, vol. 21, pp. 406–426, 2007.

[202] H. Winning, E. Roldan-Marin, L. O. Dragsted, N. Viereck, M. Poulsen, C. Sánchez-Moreno, M. P. Cano, and S. B. Engelsen, "An exploratory NMR nutri-metabonomic investigation reveals dimethyl sulfone as a dietary biomarker for onion intake," *Analyst*, vol. 134, pp. 2344–2351, 2009.

[203] A. Aizerman, E. M. Braverman, and L. Rozoner, "Theoretical foundations of the potential function method in pattern recognition learning," *Automation and remote control*, vol. 25, pp. 821–837, 1964.

[204] F. Lindgren, P. Geladi, and S. Wold, "The kernel algorithm for PLS," *Journal of Chemometrics*, vol. 7, pp. 45–59, 1993.

[205] S. Rä, nnar, F. Lindgren, P. Geladi, and S. Wold, "A PLS kernel algorithm for data sets with many variables and fewer objects. Part 1: Theory and algorithm," *Journal of Chemometrics*, vol. 8, pp. 111–125, 1994.

[206] F. Mehl, G. Marti, P. Merle, E. Delort, L. Baroux, H. Sommer, J.-L. Wolfender, S. Rudaz, and J. Boccard, "Integrating metabolomic data from multiple analytical platforms for a comprehensive characterisation of lemon essential oils," *Flavour and Fragrance Journal*, vol. 30, pp. 131–138, 2015.

[207] M. Rantalainen, M. Bylesjö, O. Cloarec, J. K. Nicholson, E. Holmes, and J. Trygg, "Kernel-based orthogonal projections to latent structures (K-OPLS)," *Journal of Chemometrics*, vol. 21, pp. 376–385, 2007.

[208] N. Cristianini and J. Shawe-Taylor, *An introduction to support vector machines and other kernel-based learning methods.* Cambridge university press, 2000.

[209] P. Bernini, I. Bertini, A. Calabro, G. la Marca, G. Lami, C. Luchinat, D. Renzi, and L. Tenori, "Are patients with potential celiac disease really potential? The answer of metabonomics," *Journal of Proteome Research*, vol. 10, pp. 714–721, 2010.

[210] J. M. Fonville, M. Bylesjö, M. Coen, J. K. Nicholson, E. Holmes, J. C. Lindon, and M. Rantalainen, "Non-linear modeling of metabonomic data using kernel-based orthogonal projections to latent structures optimized by simulated annealing," *Analytica Chimica Acta*, vol. 705, pp. 72–80, 2011.

[211] T. Bäck, *Evolutionary Algorithms in Theory and Practice.* Oxford University Press, 1996.

[212] H. C. Yeo, B. K.-S. Chung, W. Chong, J. X. Chin, K. S. Ang, M. Lakshmanan, Y. S. Ho, and D.-Y. Lee, "A genetic algorithm-based approach for pre-processing metabolomics and lipidomics LC-MS data," *Metabolomics*, vol. 12, pp. 1–13, 2016.

[213] H. E. Johnson, D. Broadhurst, R. Goodacre, and A. R. Smith, "Metabolic fingerprinting of salt-stressed tomatoes," *Phytochemistry*, vol. 62, pp. 919–928, 2003.

[214] B. Tran, B. Xue, and M. Zhang, "Genetic programming for feature construction and selection in classification on high-dimensional data," *Memetic Computing*, vol. 8, pp. 3–15, 2016.

[215] J. R. Koza, *Genetic programming: on the programming of computers by means of natural selection.* Cambridge, MA, USA: MIT Press, 1992.

[216] H. F. Gray, R. J. Maxwell, I. Martínez-Pérez, C. Arús, and S. Cerdán, "Genetic programming for classification and feature selection: analysis of ^{1}H nuclear magnetic resonance spectra from human brain tumour biopsies," *NMR in Biomedicine*, vol. 11, pp. 217–224, 1998.

[217] R. A. Davis, A. J. Charlton, S. Oehlschlager, and J. C. Wilson, "Novel feature selection method for genetic programming using metabolomic ^{1}H NMR data," *Chemometrics and Intelligent Laboratory Systems*, vol. 81, pp. 50–59, 2006.

[218] J. McDermott, D. White, S. Luke, L. Manzoni, M. Castelli, L. Vanneschi, W. Jaskowski, K. Krawiec, R. Harper, K. De Jong *et al.*, "Genetic programming needs better benchmarks," in *Proceedings of the fourteenth international conference on Genetic and evolutionary computation conference*, Conference Proceedings, pp. 791–798.

[219] J. A. Donarski, S. A. Jones, and A. J. Charlton, "Application of cryoprobe ^{1}H nuclear magnetic resonance spectroscopy and multivariate analysis for the verification of Corsican honey," *Journal of Agricultural and Food Chemistry*, vol. 56, pp. 5451–5456, 2008.

[220] J. A. Donarski, S. A. Jones, M. Harrison, M. Driffield, and A. J. Charlton, "Identification of botanical biomarkers found in corsican honey," *Food Chemistry*, vol. 118, pp. 987–994, 2010.

[221] J. M. Andrade-Garda, A. Carlosena-Zubieta, M. a. P. Gó, mez-Carracedo, and M. Gestal-Pose, "Multivariate regression using artificial neural networks," *Basic Chemometric Techniques in Atomic Spectroscopy*, p. 244.

[222] J. Thomsen and B. Meyer, "Pattern recognition of the ^{1}H NMR spectra of sugar alditols using a neural network," *Journal of Magnetic Resonance (1969)*, vol. 84, pp. 212–217, 1989.

[223] V. Kvasnička, "An application of neural networks in chemistry," *Chemical Papers*, vol. 44, pp. 775–792, 1990.

[224] M. Kjær and F. M. Poulsen, "Identification of 2D ^{1}H NMR antiphase cross peaks using a neural network," *Journal of Magnetic Resonance (1969)*, vol. 94, pp. 659–663, 1991.

[225] D. A. Latino and J. Aires-de Sousa, "Automatic NMR-based identification of chemical reaction types in mixtures of co-occurring reactions," *PloS One*, vol. 9, p. e88499, 2014.

[226] Maulidiani, F. Abas, A. Khatib, M. Shitan, K. Shaari, and N. H. Lajis, "Comparison of partial least squares and artificial neural network for the prediction of antioxidant activity in extract of *Pegaga* (*Centella*) varieties from ^{1}H nuclear magnetic resonance spectroscopy," *Food Research International*, vol. 54, pp. 852–860, 2013.

[227] K. Munro, T. H. Miller, C. P. Martins, A. M. Edge, D. A. Cowan, and L. P. Barron, "Artificial neural network modelling of pharmaceutical residue retention times in wastewater extracts using gradient liquid chromatography-high resolution mass spectrometry data," *Journal of Chromatography A*, vol. 1396, pp. 34–44, 2015.

[228] L. P. Barron and G. L. McEneff, "Gradient liquid chromatographic retention time prediction for suspect screening applications: A critical assessment of a generalised artificial neural network-based approach across 10 multi-residue reversed-phase analytical methods," *Talanta*, vol. 147, pp. 261–270, 2016.

[229] S. Rezzi, I. Giani, K. Héberger, D. E. Axelson, V. M. Moretti, F. Reniero, and C. Guillou, "Classification of gilthead sea bream (*Sparus aurata*) from ^{1}H NMR lipid profiling combined with principal component and linear discriminant analysis," *Journal of Agricultural and Food Chemistry*, vol. 55, pp. 9963–9968, 2007.

[230] J. Kaartinen, Y. Hiltunen, P. Kovanen, and M. Ala-Korpela, "Application of self-organizing maps for the detection and classification of human blood plasma lipoprotein lipid profiles on the basis of ^{1}H NMR spectroscopy data," *NMR in Biomedicine*, vol. 11, pp. 168–176, 1998.

[231] S. Kalelkar, E. R. Dow, J. Grimes, M. Clapham, and H. Hu, "Automated analysis of proton NMR spectra from combinatorial rapid parallel synthesis using self-organizing maps," *Journal of Combinatorial Chemistry*, vol. 4, pp. 622–629, 2002.

[232] O. Cloarec, M.-E. Dumas, A. Craig, R. H. Barton, J. Trygg, J. Hudson, C. Blancher, D. Gauguier, J. C. Lindon, E. Holmes *et al.*, "Statistical

total correlation spectroscopy: an exploratory approach for latent biomarker identification from metabolic ^{1}H NMR data sets," *Analytical Chemistry*, vol. 77, pp. 1282–1289, 2005.

[233] S. L. Robinette, K. A. Veselkov, E. Bohus, M. Coen, H. C. Keun, T. M. Ebbels, O. Beckonert, E. C. Holmes, J. C. Lindon, and J. K. Nicholson, "Cluster analysis statistical spectroscopy using nuclear magnetic resonance generated metabolic data sets from perturbed biological systems," *Analytical Chemistry*, vol. 81, pp. 6581–6589, 2009.

[234] M. Defernez and E. K. Kemsley, "The use and misuse of chemometrics for treating classification problems," *TrAC Trends in Analytical Chemistry*, vol. 16, pp. 216–221, 1997.

[235] D. M. Hawkins, "The problem of overfitting," *Journal of Chemical Information and Computer Sciences*, vol. 44, pp. 1–12, 2004.

[236] J. C. Lindon, J. K. Nicholson, E. Holmes, H. C. Keun, A. Craig, J. T. Pearce, S. J. Bruce, N. Hardy, S.-A. Sansone, and H. Antti, "Summary recommendations for standardization and reporting of metabolic analyses," *Nature Biotechnology*, vol. 23, pp. 833–838, 2005.

[237] D. I. Broadhurst and D. B. Kell, "Statistical strategies for avoiding false discoveries in metabolomics and related experiments," *Metabolomics*, vol. 2, pp. 171–196, 2006.

[238] P. Tripathi, B. S. Somashekar, M. Ponnusamy, A. Gursky, S. Dailey, P. Kunju, C. T. Lee, A. M. Chinnaiyan, T. M. Rajendiran, and A. Ramamoorthy, "HR-MAS NMR tissue metabolomic signatures cross-validated by mass spectrometry distinguish bladder cancer from benign disease," *Journal of Proteome Research*, vol. 12, pp. 3519–3528, 2013.

[239] C. D. Wijetunge, Z. Li, I. Saeed, J. Bowne, A. L. Hsu, U. Roessner, A. Bacic, and S. K. Halgamuge, "Exploratory analysis of high-throughput metabolomic data," *Metabolomics*, pp. 1–10, 2013.

[240] J. M. Fonville, S. E. Richards, R. H. Barton, C. L. Boulange, T. Ebbels, J. K. Nicholson, E. Holmes, and M.-E. Dumas, "The evolution of partial least squares models and related chemometric ap-

proaches in metabonomics and metabolic phenotyping," *Journal of Chemometrics*, vol. 24, pp. 636–649, 2010.

[241] M.-E. Dumas, R. H. Barton, A. Toye, O. Cloarec, C. Blancher, A. Rothwell, J. Fearnside, R. Tatoud, V. Blanc, and J. C. Lindon, "Metabolic profiling reveals a contribution of gut microbiota to fatty liver phenotype in insulin-resistant mice," *Proceedings of the National Academy of Sciences*, vol. 103, pp. 12 511–12 516, 2006.

[242] M. L. Green and P. D. Karp, "A Bayesian method for identifying missing enzymes in predicted metabolic pathway databases," *BMC Bioinformatics*, vol. 5, p. 76, 2004.

[243] J. A. Westerhuis, T. Kourti, and J. F. MacGregor, "Analysis of multiblock and hierarchical PCA and PLS models," *Journal of Chemometrics*, vol. 12, pp. 301–321, 1998.

[244] M. Rantalainen, O. Cloarec, O. Beckonert, I. Wilson, D. Jackson, R. Tonge, R. Rowlinson, S. Rayner, J. Nickson, and R. W. Wilkinson, "Statistically integrated metabonomic-proteomic studies on a human prostate cancer xenograft model in mice," *Journal of Proteome Research*, vol. 5, pp. 2642–2655, 2006.

[245] T. Crews and M. Peoples, "Legume versus fertilizer sources of nitrogen: ecological tradeoffs and human needs," *Agriculture, Ecosystems & Environment*, vol. 102, pp. 279–297, 2004.

[246] P. H. Graham and C. P. Vance, "Legumes: importance and constraints to greater use," *Plant physiology*, vol. 131, pp. 872–877, 2003.

[247] K. Leath and R. Byers, "Interaction of fusarium root rot with pea aphid and potato leafhopper feeding on forage legumes," *Phytopathology*, vol. 67, pp. 226–229, 1977.

[248] P. B. . and et al., "Improving the resistance of legume crops to combined abiotic and biotic stress," *Unpublished (The Food and Environment Research Agency, York, UK.)*.

[249] E. A. Curl, "Control of plant diseases by crop rotation," *The Botanical Review*, vol. 29, pp. 413–479, 1963.

[250] A. J. Cortés, F. A. Monserrate, J. Ramírez-Villegas, S. Madriñán, and M. W. Blair, "Drought tolerance in wild plant populations: the case of common beans (phaseolus vulgaris l.)," *PloS one*, vol. 8, p. e62898, 2013.

[251] J. S. McKenzie, "Assessment of the complementarity of data from multiple analytical techniques," Thesis, 2013. [Online]. Available: http://etheses.whiterose.ac.uk/4471

[252] DEFRA, "Agriculture in the United Kingdom," 2014. [Online]. Available: https://www.gov.uk/government/collections/agriculture-in-the-united-kingdom

[253] R. J. Epley, "Aging beef," *University of Minnesota – Agriculture*, vol. AG-FS-5968-A, 1992.

[254] S. Graham, T. Kennedy, O. Chevallier, A. Gordon, L. Farmer, C. Elliott, and B. Moss, "The application of NMR to study changes in polar metabolite concentrations in beef longissimus dorsi stored for different periods post mortem," *Metabolomics*, vol. 6, pp. 395–404, 2010.

[255] S. F. Graham, D. Farrell, T. Kennedy, A. Gordon, L. Farmer, C. Elliott, and B. Moss, "Comparing gc-ms, hplc and analysis of beef longissimus dorsi tissue extracts to determine the effect of suspension technique and ageing," *Food Chemistry*, vol. 134, pp. 1633–1639, 2012.

[256] D. Ferguson, H. Bruce, J. Thompson, A. Egan, D. Perry, and W. Shorthose, "Factors affecting beef palatability - farmgate to chilled carcass," *Animal Production Science*, vol. 41, pp. 879–891, 2001.

[257] C. Doyle, R. Cousens, and S. Moss, "A model of the economics of controlling alopecurus myosuroides huds. in winter wheat," *Crop Protection*, vol. 5, pp. 143–150, 1986.

[258] I. Heap, "The international survey of herbicide resistant weeds," 2016. [Online]. Available: http://www.weedscience.org/

[259] S. S. Kaundun, S.-J. Hutchings, R. P. Dale, and E. McIndoe, "Role of a novel I1781T mutation and other mechanisms in conferring resistance to Acetyl-CoA carboxylase inhibiting herbicides in a black-grass population," *PloS one*, vol. 8, p. e69568, 2013.

[260] J. A. C. Gardin, J. Gouzy, S. Carrère, and C. Délye, "ALOMYbase, a resource to investigate non-target-site-based resistance to herbicides inhibiting acetolactate-synthase (ALS) in the major grass weed *Alopecurus myosuroides* (black-grass)," *BMC Genomics*, vol. 16, p. 1, 2015.

[261] I. Cummins, D. J. Wortley, F. Sabbadin, Z. He, C. R. Coxon, H. E. Straker, J. D. Sellars, K. Knight, L. Edwards, and D. Hughes, "Key role for a glutathione transferase in multiple-herbicide resistance in grass weeds," *Proceedings of the National Academy of Sciences*, vol. 110, pp. 5812–5817, 2013.

[262] I. Heap, *Herbicide resistant weeds*. Springer, 2014, pp. 281–301.

[263] G. Bailly, R. Dale, S. Archer, D. Wright, and S. Kaundun, "Role of residual herbicides for the management of multiple herbicide resistance to ACCase and ALS inhibitors in a black-grass population," *Crop Protection*, vol. 34, pp. 96–103, 2012.

[264] W. A. Korfmacher, "Foundation review: Principles and applications of LC-MS in new drug discovery," *Drug Discovery Today*, vol. 10, pp. 1357–1367, 2005.

[265] X. Lu, X. Zhao, C. Bai, C. Zhao, G. Lu, and G. Xu, "LC-MS-based metabonomics analysis," *Journal of Chromatography B*, vol. 866, pp. 64–76, 2008.

[266] K. Zhang, J. W. Wong, P. Yang, K. Tech, A. L. DiBenedetto, N. S. Lee, D. G. Hayward, C. M. Makovi, A. J. Krynitsky, and K. Banerjee, "Multiresidue pesticide analysis of agricultural commodities using acetonitrile salt-out extraction, dispersive solid-phase sample clean-up, and high-performance liquid chromatography–tandem mass spectrometry," *Journal of Agricultural and Food Chemistry*, vol. 59, pp. 7636–7646, 2011.

[267] L. M. Shalaby, F. Q. Bramble, and P. W. Lee, "Application of thermospray LC/MS for residue analysis of sulfonylurea herbicides and their degradation products," *Journal of Agricultural and Food Chemistry*, vol. 40, pp. 513–517, 1992.

[268] M. S. Kostich, A. L. Batt, and J. M. Lazorchak, "Concentrations of prioritized pharmaceuticals in effluents from 50 large wastewater treatment plants in the us and implications for risk estimation," *Environmental Pollution*, vol. 184, pp. 354–359, 2014.

[269] R. Nakabayashi and K. Saito, "Metabolomics for unknown plant metabolites," *Analytical and Bioanalytical Chemistry*, vol. 405, pp. 5005–5011, 2013.

[270] G. Gürdeniz, D. Rago, N. T. Bendsen, F. Savorani, A. Astrup, and L. O. Dragsted, "Effect of trans fatty acid intake on LC-MS and NMR plasma profiles," *PloS one*, vol. 8, p. e69589, 2013.

[271] L. Lai, F. Michopoulos, H. Gika, G. Theodoridis, R. W. Wilkinson, R. Odedra, J. Wingate, R. Bonner, S. Tate, and I. D. Wilson, "Methodological considerations in the development of HPLC-MS methods for the analysis of rodent plasma for metabonomic studies," *Molecular Biosystems*, vol. 6, pp. 108–120, 2009.

[272] K. E. A. Ohlsson and P. H. Wallmark, "Novel calibration with correction for drift and non-linear response for continuous flow isotope ratio mass spectrometry applied to the determination of δ ^{15}N, total nitrogen, δ ^{13}C and total carbon in biological material†," *Analyst*, vol. 124, pp. 571–577, 1999.

[273] W. B. Dunn, D. Broadhurst, P. Begley, E. Zelena, S. Francis-McIntyre, N. Anderson, M. Brown, J. D. Knowles, A. Halsall, and J. N. Haselden, "Procedures for large-scale metabolic profiling of serum and plasma using gas chromatography and liquid chromatography coupled to mass spectrometry," *Nature protocols*, vol. 6, pp. 1060–1083, 2011.

[274] F. M. Van Der Kloet, I. Bobeldijk, E. R. Verheij, and R. H. Jellema, "Analytical error reduction using single point calibration for accurate and precise metabolomic phenotyping," *Journal of Proteome Research*, vol. 8, pp. 5132–5141, 2009.

[275] P. Begley, S. Francis-McIntyre, W. B. Dunn, D. I. Broadhurst, A. Halsall, A. Tseng, J. Knowles, R. Goodacre, and D. B. Kell, "Development and performance of a gas chromatography–time-of-flight mass

spectrometry analysis for large-scale nontargeted metabolomic studies of human serum," *Analytical Chemistry*, vol. 81, pp. 7038–7046, 2009.

[276] J. Kirwan, D. Broadhurst, R. Davidson, and M. Viant, "Characterising and correcting batch variation in an automated direct infusion mass spectrometry (dims) metabolomics workflow," *Analytical and bioanalytical chemistry*, vol. 405, pp. 5147–5157, 2013.

[277] U. D. of Health and H. Services, "Guidance for industry – bioanalytical method validation," 2001. [Online]. Available: http://www.fda.gov/downloads/Drugs/Guidance/ucm070107.pdf

[278] P. H. Eilers, "A perfect smoother," *Analytical chemistry*, vol. 75, pp. 3631–3636, 2003.

[279] W. S. Cleveland, "Robust locally weighted regression and smoothing scatterplots," *Journal of the American statistical association*, vol. 74, pp. 829–836, 1979.

[280] M. L. Salit and G. C. Turk, "A drift correction procedure," *Analytical Chemistry*, vol. 70, pp. 3184–3190, 1998.

[281] M. R. N. Ranjbar, Y. Zhao, M. G. Tadesse, Y. Wang, and H. W. Ressom, "Evaluation of normalization methods for analysis of LC-MS data," in *Bioinformatics and Biomedicine Workshops (BIBMW), 2012 IEEE International Conference on.* IEEE, Conference Proceedings, pp. 610–617.

[282] H. G. Gika, E. Macpherson, G. A. Theodoridis, and I. D. Wilson, "Evaluation of the repeatability of ultra-performance liquid chromatography-TOF-MS for global metabolic profiling of human urine samples," *Journal of Chromatography B*, vol. 871, pp. 299–305, 2008.

[283] B. D. Prakash and Y. C. Wei, "A fully automated iterative moving averaging (AIMA) technique for baseline correction," *Analyst*, vol. 136, pp. 3130–3135, 2011.

[284] R. D. C. Team, "R." [Online]. Available: https://www.r-project.org/

[285] D. Rodbard, "Statistical quality control and routine data processing for radioimmunoassays and immunoradiometric assays," *Clinical Chemistry*, vol. 20, pp. 1255–1270, 1974.

[286] T. J. Hastie and R. J. Tibshirani, *Generalized additive models*. CRC Press, 1990, vol. 43.

[287] B. D. Ripley and M. Maechler, "Fit a smoothing spline." [Online]. Available: https://cran.r-project.org/web/packages/pspline/

[288] K. Kultima, A. Nilsson, B. Scholz, U. L. Rossbach, M. Fälth, and P. E. Andrén, "Development and evaluation of normalization methods for label-free relative quantification of endogenous peptides," *Molecular & Cellular Proteomics*, vol. 8, pp. 2285–2295, 2009.

[289] M. Calvin, "The photosynthetic carbon cycle," *J. Chem. Soc.*, pp. 1895–1915, 1956. [Online]. Available: http://dx.doi.org/10.1039/JR9560001895

[290] R. Goodacre, S. Vaidyanathan, W. B. Dunn, G. G. Harrigan, and D. B. Kell, "Metabolomics by numbers: acquiring and understanding global metabolite data," *Trends in biotechnology*, vol. 22, pp. 245–252, 2004.

[291] S. Oliver, "Proteomics: guilt-by-association goes global," *Nature*, vol. 403, pp. 601–603, 2000.

[292] J. Gillis and P. Pavlidis, ""Guilt by association" is the exception rather than the rule in gene networks," *PLOS Computational Biology*, vol. 8, p. e1002444, 2012.

[293] C. D. Broeckling, D. V. Huhman, M. A. Farag, J. T. Smith, G. D. May, P. Mendes, R. A. Dixon, and L. W. Sumner, "Metabolic profiling of medicago truncatula cell cultures reveals the effects of biotic and abiotic elicitors on metabolism," *Journal of Experimental Botany*, vol. 56, pp. 323–336, 2005.

[294] H. Kessmann, R. Edwards, P. W. Geno, and R. A. Dixon, "Stress responses in alfalfa (*Medicago sativa* l.) – V. constitutive and elicitor-induced accumulation of isoflavonoid conjugates in cell suspension cultures," *Plant Physiology*, vol. 94, pp. 227–232, 1990.

[295] G. K. Pierens, M. E. Palframan, C. J. Tranter, A. R. Carroll, and R. J. Quinn, "A robust clustering approach for NMR spectra of natural product extracts," *Magnetic Resonance in Chemistry*, vol. 43, pp. 359–365, 2005.

[296] J. Qian, M. Dolled-Filhart, J. Lin, H. Yu, and M. Gerstein, "Beyond synexpression relationships: local clustering of time-shifted and inverted gene expression profiles identifies new, biologically relevant interactions," *Journal of Molecular Biology*, vol. 314, pp. 1053–1066, 2001.

[297] A. K. Gombert and J. Nielsen, "Mathematical modelling of metabolism," *Current Opinion in Biotechnology*, vol. 11, pp. 180–186, 2000.

[298] R. J. Cho, M. J. Campbell, E. A. Winzeler, L. Steinmetz, A. Conway, L. Wodicka, T. G. Wolfsberg, A. E. Gabrielian, D. Landsman, D. J. Lockhart *et al.*, "A genome-wide transcriptional analysis of the mitotic cell cycle," *Molecular cell*, vol. 2, pp. 65–73, 1998.

[299] S. J. Kiddle, O. P. Windram, S. McHattie, A. Mead, J. Beynon, V. Buchanan-Wollaston, K. J. Denby, and S. Mukherjee, "Temporal clustering by affinity propagation reveals transcriptional modules in *Arabidopsis thaliana*," *Bioinformatics*, vol. 26, pp. 355–362, 2010.

[300] B. J. Frey and D. Dueck, "Clustering by passing messages between data points," *Science*, vol. 315, pp. 972–976, 2007.

[301] M.-E. Dumas, E. C. Maibaum, C. Teague, H. Ueshima, B. Zhou, J. C. Lindon, J. K. Nicholson, J. Stamler, P. Elliott, and Q. Chan, "Assessment of analytical reproducibility of ^{1}H NMR spectroscopy based metabonomics for large-scale epidemiological research: the intermap study," *Analytical Chemistry*, vol. 78, pp. 2199–2208, 2006.

[302] B. Pagano, I. Lauri, S. De Tito, G. Persico, M. G. Chini, A. Malmendal, E. Novellino, and A. Randazzo, "Use of NMR in profiling of cocaine seizures," *Forensic Science International*, vol. 231, pp. 120–124, 2013.

[303] S. P. Lloyd, "Least squares quantization in PCM," *Information Theory, IEEE Transactions on*, vol. 28, pp. 129–137, 1982.

[304] A. S. Gajadhar, H. Johnson, R. J. Slebos, K. Shaddox, K. Wiles, M. K. Washington, A. J. Herline, D. A. Levine, D. C. Liebler, and F. M. White, "Phosphotyrosine signaling analysis in human tumors is confounded by systemic ischemia-driven artifacts and intra-specimen heterogeneity," *Cancer Research*, vol. 75, pp. 1495–1503, 2015.

[305] C. J. Jeffery, "Moonlighting proteins: old proteins learning new tricks," *TRENDS in Genetics*, vol. 19, pp. 415–417, 2003.

[306] D. Dembélé and P. Kastner, "Fuzzy C-means method for clustering microarray data," *Bioinformatics*, vol. 19, pp. 973–980, 2003.

[307] M. Čuperlović Culf, N. Belacel, A. S. Culf, I. C. Chute, R. J. Ouellette, I. W. Burton, T. K. Karakach, and J. A. Walter, "NMR metabolic analysis of samples using fuzzy K-means clustering," *Magnetic Resonance in Chemistry*, vol. 47, pp. S96–S104, 2009.

[308] J. V. Olsen, B. Blagoev, F. Gnad, B. Macek, C. Kumar, P. Mortensen, and M. Mann, "Global, in vivo, and site-specific phosphorylation dynamics in signaling networks," *Cell*, vol. 127, pp. 635–648, 2006.

[309] C. Ahn, U. Park, and P. B. Park, "Increased salt and drought tolerance by D-ononitol production in transgenic *Arabidopsis thaliana*," *Biochemical and Biophysical Research Communications*, vol. 415, pp. 669–674, 2011.

[310] E. Sheveleva, W. Chmara, H. J. Bohnert, and R. G. Jensen, "Increased salt and drought tolerance by D-ononitol production in transgenic *Nicotiana tabacum* l," *Plant Physiology*, vol. 115, pp. 1211–1219, 1997.

[311] D. H. Wolpert and W. G. Macready, "No free lunch theorems for optimization," *IEEE Transactions on Evolutionary Computation*, vol. 1, pp. 67–82, 1997.

[312] R. Tautenhahn, G. J. Patti, D. Rinehart, and G. Siuzdak, "XCMS Online: a web-based platform to process untargeted metabolomic data," *Analytical Chemistry*, vol. 84, pp. 5035–5039, 2012.

[313] J. Xia, R. Mandal, I. V. Sinelnikov, D. Broadhurst, and D. S. Wishart, "MetaboAnalyst 2.0—a comprehensive server for metabolomic data analysis," *Nucleic Acids Research*, vol. 40, pp. W127–W133, 2012.

[314] J. Xia, N. Psychogios, N. Young, and D. S. Wishart, "MetaboAnalyst: a web server for metabolomic data analysis and interpretation," *Nucleic Acids Research*, vol. 37, pp. W652–W660, 2009.

[315] K. Abe and J.-M. Perraud, "R.net," 2015. [Online]. Available: https://rdotnet.codeplex.com/

[316] C. Rüegg, "Math.net numerics," 2015. [Online]. Available: https://numerics.mathdotnet.com/

[317] BioPAX.org, "Biopax : Biological pathways exchange," 2014. [Online]. Available: http://www.biopax.org/

[318] M. Corporation, "[ms-nrbf]: .net remoting: Binary format data structure," 2016. [Online]. Available: https://msdn.microsoft.com/en-us/library/cc236844.aspx

[319] S.-I. Ao and L. Gelman, *Electronic Engineering and Computing Technology*. Springer, 2010, vol. 60.

[320] E. Urbanczyk-Wochniak and L. W. Sumner, "Mediccyc: a biochemical pathway database for *Medicago truncatula*," *Bioinformatics*, vol. 23, pp. 1418–1423, 2007.

[321] T. Kind, "Mass spectrometry adduct calculator," 2010. [Online]. Available: http://fiehnlab.ucdavis.edu/staff/kind/Metabolomics/MS-Adduct-Calculator

[322] M. Rusilowicz, M. Dickinson, A. Charlton, S. O'Keefe, and J. Wilson, "A batch correction method for liquid chromatography–mass spectrometry data that does not depend on quality control samples," *Metabolomics*, vol. 12, pp. 1–11, 2016.

[323] A. G. Good and S. T. Zaplachinski, "The effects of drought stress on free amino acid accumulation and protein synthesis in brassica napus," *Physiologia Plantarum*, vol. 90, pp. 9–14, 1994.

[324] S. Ramanjulu and C. Sudhakar, "Drought tolerance is partly related to amino acid accumulation and ammonia assimilation: a comparative study in two mulberry genotypes differing in drought sensitivity," *Journal of Plant Physiology*, vol. 150, pp. 345–350, 1997.

[325] S. A. Hammad and O. A. Ali, "Physiological and biochemical studies on drought tolerance of wheat plants by application of amino acids and yeast extract," *Annals of Agricultural Sciences*, vol. 59, pp. 133–145, 2014.

[326] "Plant metabolic pathway database (pnm/plantcyc)," 2016. [Online]. Available: http://www.plantcyc.org

[327] S. Clouse, "Brassinosteroids," *Current Biology*, vol. 11, p. R904, 2001.

[328] M. Rusilowicz, "Martin rusilowicz," 2015. [Online]. Available: https://bitbucket.org/mjr129

[329] J. S. McKenzie, J. A. Donarski, J. C. Wilson, and A. J. Charlton, "Analysis of complex mixtures using high-resolution nuclear magnetic resonance spectroscopy and chemometrics," *Progress in Nuclear Magnetic Resonance Spectroscopy*, vol. 59, pp. 336–359, 2011.

[330] E. Holmes, A. W. Nicholls, J. C. Lindon, S. Ramos, M. Spraul, P. Neidig, S. C. Connor, J. Connelly, S. J. P. Damment, J. Haselden, and J. K. Nicholson, "Development of a model for classification of toxin-induced lesions using ^{1}H NMR spectroscopy of urine combined with pattern recognition," *NMR Biomed*, vol. 11, pp. 235–244, 1998.

[331] G. G. Harrigan and R. Goodacre, *Metabolic profiling: its role in biomarker discovery and gene function analysis*. Springer, 2003.

[332] R. Goodacre, B. Shann, R. J. Gilbert, E. M. Timmins, A. C. McGovern, B. K. Alsberg, D. B. Kell, and N. A. Logan, "Detection of the dipicolinic acid biomarker in bacillus spores using curie-point pyrolysis mass spectrometry and Fourier transform infrared spectroscopy," *Anal. Chem.*, vol. 72, pp. 119–127, 2000.

[333] A. Friedlander, K. Neshatian, and M. Zhang, "Meta-learning and feature ranking using genetic programming for classification: Variable

terminal weighting," in *Evolutionary Computation (CEC), 2011 IEEE Congress on*, Conference Proceedings, pp. 941–948.

[334] J. R. Koza, "Human-competitive results produced by genetic programming," *Genetic Programming and Evolvable Machines*, vol. 11, pp. 251–284, 2010.

[335] J. H. Holland, *Adaptation in natural and artificial systems: An introductory analysis with applications to biology, control, and artificial intelligence.* U Michigan Press, 1975.

[336] R. Forsyth, "Beagle - a darwinian approach to pattern recognition," *Kybernetes*, vol. 10, pp. 159–166, 1981.

[337] M. O'Neil and C. Ryan, *Grammatical evolution.* Springer, 2003, pp. 33–47.

[338] F. Rothlauf and M. Oetzel, *On the Locality of Grammatical Evolution*, ser. Lecture Notes in Computer Science. Springer Berlin Heidelberg, 2006, vol. 3905, book section 29, pp. 320–330.

[339] T. Castle and C. G. Johnson, *Positional effect of crossover and mutation in grammatical evolution.* Springer, 2010, pp. 26–37.

[340] D. J. Montana, "Strongly typed genetic programming," *Evolutionary computation*, vol. 3, pp. 199–230, 1995.

[341] B. Liskov and S. Zilles, "Programming with abstract data types," in *ACM Sigplan Notices*, vol. 9. ACM, Conference Proceedings, pp. 50–59.

[342] Y. Shichel, E. Ziserman, and M. Sipper, *GP-Robocode: Using Genetic Programming to Evolve Robocode Players*, ser. Lecture Notes in Computer Science. Springer Berlin Heidelberg, 2005, vol. 3447, book section 13, pp. 143–154.

[343] M. D. McKay, R. J. Beckman, and W. J. Conover, "A comparison of three methods for selecting values of input variables in the analysis of output from a computer code," *Technometrics*, vol. 21, pp. 239–245, 1979. [Online]. Available: http://www.jstor.org/stable/1268522

[344] J. Jeronen, "Orthogonal sample," 2010. [Online]. Available: https://yousource.it.jyu.fi/~jumijero

[345] D. E. Knuth, *The Art of Computer Programming, Volume 2: Seminumerical Algorithms. Third edition.* Addison-Wesley, 1997.

[346] Microsoft, "Random class," 2016. [Online]. Available: https://msdn.microsoft.com/en-us/library/system.random.aspx

[347] D. R. White and S. Poulding, *A rigorous evaluation of crossover and mutation in genetic programming.* Springer, 2009, pp. 220–231.

[348] A. Vargha and H. D. Delaney, "A critique and improvement of the CL common language effect size statistics of McGraw and Wong," *Journal of Educational and Behavioral Statistics*, vol. 25, no. 2, pp. 101–132, 2000.

[349] J. M. Daida, D. S. Ampy, M. Ratanasavetavadhana, H. Li, and O. A. Chaudhri, "Challenges with verification, repeatability, and meaningful comparison in genetic programming: Gibson's magic," in *Proceedings of the 1st Annual Conference on Genetic and Evolutionary Computation-Volume 2.* Morgan Kaufmann Publishers Inc., Conference Proceedings, pp. 1851–1858.

[350] W. H. Press, S. A. Teukolsky, W. T. Vetterling, and B. P. Flannery, *Numerical Recipes in C*, 2nd ed. Cambridge Univiersity Press, 1992.

[351] ytosa and M. Corporation, "Visual Studio – System.Random serious bug," 2011. [Online]. Available: https://connect.microsoft.com/VisualStudio/feedback/details/634761/system-random-serious-bug

[352] R. Wehrens, J. A. Hageman, F. van Eeuwijk, R. Kooke, P. J. Flood, E. Wijnker, J. J. Keurentjes, A. Lommen, H. D. van Eekelen, R. D. Hall *et al.*, "Improved batch correction in untargeted MS-based metabolomics," *Metabolomics*, vol. 12, no. 5, pp. 1–12, 2016.

[353] V. Manahov, "The rise of the machines in commodities markets: new evidence obtained using strongly typed genetic programming," *Annals of Operations Research*, pp. 1–32, 2016.

[354] M. Belmadani, "MotifGP: DNA motif discovery using multiobjective evolution," Ph.D. dissertation, University of Ottawa, 2016.

[355] V. Sarpe and D. C. Schriemer, "Supporting metabolomics with adaptable software: design architectures for the end-user," *Current Opinion in Biotechnology*, vol. 43, pp. 110–117, 2017.

[356] H. R. Eghbalnia, P. R. Romero, W. M. Westler, K. Baskaran, E. L. Ulrich, and J. L. Markley, "Increasing rigor in NMR-based metabolomics through validated and open source tools," *Current opinion in biotechnology*, vol. 43, pp. 56–61, 2017.

[357] B. Jagla, B. Wiswedel, and J.-Y. Coppée, "Extending KNIME for next-generation sequencing data analysis," *Bioinformatics*, vol. 27, no. 20, pp. 2907–2909, 2011.

[358] E. Afgan, D. Baker, M. Van den Beek, D. Blankenberg, D. Bouvier, M. Čech, J. Chilton, D. Clements, N. Coraor, C. Eberhard *et al.*, "The galaxy platform for accessible, reproducible and collaborative biomedical analyses: 2016 update," *Nucleic acids research*, p. gkw343, 2016.

[359] M. Katajamaa and M. Orešič, "Data processing for mass spectrometry-based metabolomics," *Journal of Chromatography A*, vol. 1158, pp. 318–328, 2007.