11-2018

Mass Spectrometry and Nuclear Magnetic Resonance in the Chemometric Analysis of Cellular Metabolism

Eli Riekeberg
University of Nebraska - Lincoln, eriekeberg@huskers.unl.edu

Mass Spectrometry and Nuclear Magnetic Resonance in the Chemometric Analysis of Cellular Metabolism

By

Eli P. Riekeberg

A THESIS

Presented to the Faculty of
The Graduate College at the University of Nebraska

In Partial Fulfillment of Requirements

For the Degree of Master of Science

Major: Chemistry

Under the Supervision of Professor Robert Powers

Lincoln, Nebraska

November 2018

Mass Spectrometry and Nuclear Magnetic Resonance in the Chemometric Analysis of Cellular Metabolism

Eli Riekeberg, M.S.

University of Nebraska, 2018

Advisor: Robert Powers

The development and awareness of Machine Learning and "big data" has led to a growing interest in applying these methods to bioanalytical research. Methods such as Mass Spectrometry (MS), and Nuclear Magnetic Resonance (NMR) can now obtain tens of thousands to millions of data points from a single sample, due to fundamental instrumental advances and ever-increasing resolution. Simple pairwise comparisons on datasets of this magnitude can obfuscate more complex underlying trends, and does a disservice to the richness of information contained within. This necessitates the need for multivariate approaches that can more fully take advantage of the complexity of these datasets.

Performing these multivariate analyses takes high degree of expertise, requiring knowledge of such disparate areas as chemistry, physics, mathematics, statistics, software development and signal processing. As a result, this barrier to entry prevents many investigators from fully utilizing all the tools available to them, instead relying on a mix of commercial and free software, chained together with in-house developed solutions just to perform a single analysis. While there are numerous methods in published literature for statistical analysis of these larger datasets, most are still confined to the realm of theory due to them not being implemented into publicly available software for the research community.

This dissertation outlines the development of routines for handling LC-MS data with freely available tools, including the Octave programming language. This presents, in combination with our previously developed software MVAPACK, a unified platform for metabolomics data analysis that will encourage the wider adoption of multi-instrument investigations and multiblock statistical analyses.

Acknowledgements:

To my Mother and Father: Without you, I never would have gotten this far. Mom, you always encouraged me to follow whatever could hold my interest (sometimes to a fault!), which led to me always thinking I would be a scientist from a young age. Dad, we might not have had the same ideas about my career but you always encouraged the value of hard work over merely talking about it. I have you to thank for an insatiable work ethic and the will to grit my teeth and bear through tough times, which has served me well in every aspect of my life, not merely graduate school.

When considering the heartfelt gratitude I owe my family, I can't help but think about all of the amazing colleagues I consider a second family away from home. I have had the opportunity to work alongside some incredible people over the years. To Darrell Marshall, your friendship and thoughts about what success in graduate school entails continue to inspire me and inform me daily. To Bradley Worley, Jon Catazaro, Shulei Lei and Teklab Grebregiworgis, I am grateful for all the time we spent together, and learning as much as I could from your experience. I only hope that I can follow your example and be as strong of a mentor to someone who comes after me. To the rest of our research group, I can't help but be proud of the community we have built. Whether it's talking about our research, personal endeavors, or intellectual burdens, I've always found that "iron sharpens iron" and we all come away a bit stronger as a result. I truly believe that our prolific collaboration and inter-disciplinarity could not be possible without merging our specialties together so that everyone can learn from our experience. I hope that we maintain, and even improve upon, this culture to help foster an increasingly inviting and thriving research group we can all be proud of.

I also want to thank Justin Borgstede, Terra Willard, Brendan Andres and Drew Timbs: your incredible moral support and confidence in me during this tumultuous period of graduate school is more valuable to me than you will ever know.

Lastly, I want to thank my advisor, Dr. Robert Powers and my supervisory committee, Dr. Pat Dussault, and Dr. Eric Dodds. Thank you for providing me an environment so brimming with opportunity that I had the chance to forge my own path, and providing me the tools to do so even when the way forward wasn't clear. Thank you all especially for your patience and guidance during my time at the University of Nebraska.

Adapted Works

Chapter 2: Chemometrics and Machine Learning in Metabolomics

-Adapted from a previous publication “New frontiers in metabolomics: from measurement to insight” by Eli Riekeberg and Robert Powers

Link: http://bionmr.unl.edu/files/publications/145.pdf

Chapter 3: The role of Normalization in Metabolomics

-Adapted from a previous publication “Comparing normalization methods and the impact of noise” by Thao Vu, Eli Riekeberg, Yumou Qiu, and Robert Powers
Link: http://bionmr.unl.edu/files/publications/152.pdf

Chapter 4: Challenges of Integrating Liquid Chromatography as a Platform in Metabolomics

-Adapted from a publication to be submitted “A Workflow for Integrated Analysis of LC-MS/NMR Data” by Eli Riekeberg, Aline De Lima Leite, and Robert Powers

Table of Contents

Chapter 1: Metabolomics: An Ideal Approach?

1.1 Introduction

Metabolomics is one of the most rapidly growing fields in biological research. Considered downstream from genomics, transcriptomics and proteomics, it forms the base of the "omics" disciplines, its goal is to form a complete analysis of the total set of metabolites with in a living system. As the metabolism of an organism should directly reflect directly observed phenotypes, it has invaluable applications in biomarker discovery[1], toxicology[2] and functional determination of genes and their protein products[3]. Despite all of this success, metabolomics still suffers from a systemic lack of standardization in protocols. The sheer variety of instrumentation, data processing modalities and statistical modeling algorithms results in a wide variety of approaches in the literature, often with little attention paid to which combination is most effective for a given problem.

Nuclear Magnetic Resonance (NMR) and Mass Spectrometry (MS) are the most widely used instruments in metabolomics, and there is a growing awareness that the two methods offer synergistic effects when combined with one another[4–6]. As a result, it has become clear that the best results are obtained when both methods are used together so as to minimize the other's particular disadvantages. Often times, the practical limitation to performing these integrated studies are not acquisition but analysis.

In NMR, a number of software packages including BATMAN[7], Bayesil[8], and TOCCATA[9] and databases such as HMDB, BMRB and Chemspider are all routinely used with varying degrees of success. Packages such as BATMAN require an intimate knowledge of the R programming language in order to use properly, whereas TOCCATA and Bayesil require little to no knowledge of programming environments and be used through a simple web interface. There is a price to this convenience unfortunately, as these online methods are not easily modified or reproduced, and lack any transparency in this "black box" approach. The aforementioned databases are also not without their shortcomings, often failing to have entries for all possible metabolites that could be encountered in a typical study. Some, such as the HMDB aim to correct this by accepting spectra acquired by the scientific community at large, which creates a more complete database but leads to a heterogeneity in sample preparation, solvents utilized and spectral acquisition parameters. MS has a similar variety of tools, including XCMS, MZMine, and MetAlign among others. MZMine for example, is a largely visual data processing tool, with required ranges and parameters being user-determined based on observed spectral properties and is easy to use for anyone with MS background. XCMS however, is much more programmatically driven and offers a steeper learning curve, but offers powerful automation tools and flexibility unavailable in other software suites. The needs of which software to use, as in any field is a tradeoff between the time necessary to devote to learning the tool and what the benefits are of its unique approach.

Both the fields of NMR and MS have independently generated their own solutions for data analysis, but little attention has been given to platform-independent variants. As a result, no single software package offers all of the required tools for metabolomics. This often forces researchers to make use of multiple software packages, utilizing the 1-2 critical functions it provides as part of a larger processing scheme. The burdensome nature of using these multiple packages presents a tremendous barrier to entry, in order to learn how to use each of the packages in an appropriate way. Additionally, this brings even larger concerns of reproducibility; how do we reproduce our results if one of the crucial pieces is updated and no longer compatible with the rest of the pipeline?

The standards that the field of metabolomics demands, uniformity of process, clarity in communicating protocols, reproducibility and comparative benchmarking are stifled by the lack of a single software package to perform all of the necessary steps. For NMR, this need has been met by MVAPACK, which provides a complete set of functions for each of the necessary steps in NMR metabolomics data analysis in an intuitive and easy to use platform. As MVAPACK is an open-source tool, it is extremely transparent in its approach and additional functionality can be added as more advantageous approaches are discovered. While this consolidation of NMR data processing in the context of metabolomics has already taken place, a similar solution has yet to be demonstrated for MS-based investigations. With the addition of functions for processing MS data within MVAPACK, a single software package is created that is capable of handling data regardless of which instrumentation is used, laying the necessary groundwork for future studies that combine both NMR and MS without the need for self-developed processing pipelines utilizing numerous unrelated software packages.

1.2 Summary of Work

This thesis will attempt to summarize the field of metabolomics, with a focus on the issues pertaining to data analysis.

Chapter 2 will introduce the field of metabolomics. Typical use cases, and a general overview of what a metabolomics experiment entails will be presented, along with an exhaustive summary of algorithms that have been applied towards understanding of the data these experiments generate. A few technical advances, primarily from the field of NMR and biomarker discovery, and their potential effects on the field are discussed.

Chapter 3 primarily highlights the burden of choice that is often encountered when processing metabolomics data. It uses a single step in data processing, normalization, and discusses the myriad of issues that can arise from this choice. A number of widely used algorithms are compared to their best-in-class counterparts, on both simulated and real NMR datasets to illustrate how relative signal intensity, noise and overall variance can drastically inform which methods are most appropriate, and perhaps more importantly, detrimental to accurate results.

Chapter 4 will present a demonstrable workflow for processing LC-MS data, using open-source software tools that will run on a typical user workstation. The major hurdles that arise from this process, as well as some possible techniques to address these limitations are discussed, as well as the potential benefits to investigators familiar with NMR that are looking to incorporate MS and vice-versa.

Chapter 5 concludes the thesis, with a few parting thoughts on the state of the field as well as presenting natural extensions to the research described herein. Potential directions of both software development and methodology and their ability to address some of the next logical steps in metabolomics are entertained.

References

1. Kind, T., Tolstikov, V., Fiehn, O. & Weiss, R. H. A comprehensive urinary metabolomic approach for identifying kidney cancer. *Anal. Biochem.* **363,** 185–195 (2007).

2. Lindon, J. C., Holmes, E. & Nicholson, J. K. Metabonomics in pharmaceutical R&D. *FEBS J.* **274,** 1140–51 (2007).

3. Joyce, A. R. & Palsson, B. Ø. The model organism as a system: Integrating ‘omics’ data sets. *Nat. Rev. Mol. Cell Biol.* **7,** 198–210 (2006).

4. Ivanisevic, J. *et al.* Toward ’Omic scale metabolite profiling: A dual separation-mass spectrometry approach for coverage of lipid and central carbon metabolism. *Anal. Chem.* **85,** 6876–6884 (2013).

5. Marshall, D. D. & Powers, R. Beyond the paradigm: Combining mass spectrometry and nuclear magnetic resonance for metabolomics. *Prog. Nucl. Magn. Reson. Spectrosc.* **100,** 1–16 (2017).

6. Deng, L. *et al.* Combining NMR and LC/MS using backward variable elimination: Metabolomics analysis of colorectal cancer, polyps, and healthy controls. *Anal. Chem.* **88,** 7975–7983 (2016).

7. Hao, J., Astle, W., De iorio, M. & Ebbels, T. M. D. Batman-an R package for the automated

quantification of metabolites from nuclear magnetic resonance spectra using a bayesian model. *Bioinformatics* **28,** 2088–2090 (2012).

8. Ravanbakhsh, S. *et al.* Accurate, fully-automated NMR spectral profiling for metabolomics. *PLoS One* **10,** 1–15 (2015).

9. Bingol, K., Bruschweiler-Li, L., Li, D. W. & Bruüschweiler, R. Customized metabolomics database for the analysis of NMR 1H-1H TOCSY and 13C-1H HSQC-TOCSY spectra of complex mixtures. *Anal. Chem.* **86,** 5494–5501 (2014).

Chapter 2: Chemometrics and Machine Learning in Metabolomics

2.1 Introduction

Metabolomics is a rapidly growing field of study that endeavors to measure the complete set of metabolites (generally considered to be the intermediates and products of cellular metabolism less than 1 kDa in size) within a biological sample (that is, the metabolome) in order to achieve a global view of the state of the system[1]. Typically, metabolomics is focused only on characterizing the water-soluble metabolites, whereas lipidomics is a specialized discipline that investigates only lipids[2]. Water-soluble metabolites are part of a mobile, open biological system and as a result can readily interact and communicate with the environment, including the microbiome[3]. This is also true for some lipids but to a much lesser extent. Consequently, metabolomics has become an essential resource for systems biology because of its unique perspective relative to genomics and proteomics. Numerous studies have measured the relative upregulation or downregulation of genes or proteins to infer changes in biological function. However, it has been shown that even for common metabolic processes, such as glycolysis, a change in the cellular concentration of an enzyme does not necessarily lead to a proportional change in metabolic flux[4]. Thus, whereas genomics and proteomics identify what might happen, metabolomics identifies what is actually happening in the system. This realization demands a different perspective and requires the measurement of transcriptional, proteomic, and metabolomic data in order to obtain a complete picture of the system's response to environmental or genetic stress. As another illustration, a "silent mutation" does not produce an observable change in phenotype despite an alteration in a gene or protein product. Therefore, metabolomics profiling can be used to decode the function of silent mutations, such as the Pfk26 and Pfk27 genes in Saccharomyces cerevisiae that both encode the glycolytic/gluconeogenesis regulator phosphofructokinase 2[5] . Via co-response and cluster analysis, these genes were observed to exhibit similar metabolite profiles which differed from other genes impacting energy metabolism. For these reasons, methods to directly measure metabolite concentrations within cells, tissues, organs, or other biological samples are crucial for fully understanding a system when traditional omics studies (for example, genomics, proteomics, and transcriptomics) are deemed insufficient. To date, nuclear magnetic resonance (NMR)[6]and mass spectrometry (MS)[7] have been the primary analytical techniques used to characterize a metabolome. NMR and MS are typically combined with univariate and multivariate statistical methods to identify major metabolite changes and to identify potential biomarkers[8] . Nevertheless, despite the tremendous growth in the field, critical protocols and techniques are still under development. Herein, we present recent advances in methodologies and statistical analysis that are enhancing and improving the performance of metabolomics while extending the applications in which metabolomics can play a significant role.

2.2 Essential Components of a Successful Metabolomics Investigation

Conceptually, an untargeted metabolomics study is quite simple. Biological samples are obtained from two or more experimental groups to be compared (healthy versus diseased, wild-type versus gene knockout, and so on) and the metabolites are extracted. These metabolic extracts then are measured by using numerous instrumental techniques, of which NMR and liquid chromatography (LC)-MS are the most common. The resulting spectra then are subjected to statistical analysis techniques such as principal component analysis (PCA) and orthogonal projection onto latent structures (OPLS) to determine the most significant spectral features that define each group[9-10]. Finally, these spectral features then can be assigned to distinct metabolites and metabolic pathways by using spectral libraries of known metabolites[11-12]. In this manner, untargeted metabolomics is discovery-based since it reveals previously unknown information about how a system responds to environmental or genetic stress. Conversely, targeted metabolomics focuses on analyzing a specific set of metabolites on the basis of some prior knowledge about the system. As a result, targeted metabolomics studies tend to be more sensitive and quantitative and have a higher reproducibility and a lower false-positive rate relative to untargeted metabolomics. Protocols for obtaining and extracting the metabolome have been well developed and exhaustively reviewed for a wide range of biological samples, including cell cultures, urine, blood/serum, and both animal- and plant-derived tissues[13-16]. Although these protocols are readily available, the variable stability of metabolites means that even minor changes in procedure can have a major impact on the observed metabolome. The fast turnover rate of enzymes and the variable temperature and chemical stability of metabolites require that metabolomics samples be collected quickly and handled uniformly and that all enzymatic activity be rapidly quenched in order to minimize biologically irrelevant deviations between samples that may result from the processing protocol[13, 17],. Thus, the optimization of the sample preparation protocol is essential to a successful metabolomics study. Conversely, the most likely source of bias is improper handling of the metabolomics samples. An important consideration is that the complete metabolome cannot be captured in a single extraction protocol. This is in stark contrast to modern genomics, which can reliably cover the entire genome of an organism. A metabolomics extraction protocol will usually focus on only a subset of metabolites (for example, water-soluble metabolites or lipids). Furthermore, an extraction protocol may focus on either a highly reproducible and quantitative extraction of a restricted set of metabolites (that is, targeted metabolomics) or the global collection of all possible metabolites (that is, untargeted metabolomics) with a possible reduction in precision. In general, 200 to 500 metabolites may be observed by targeted metabolomics, whereas upwards of 1,500 metabolites have been detected in untargeted metabolomics studies[18]. Following extraction and subsequent data collection, the final and perhaps most crucial step is metabolite assignment, which typically is accomplished by a comparison with spectral libraries of known metabolites. This is not a trivial task since the number of possible metabolites can be prohibitively large, and a large segment of the metabolome is either unknown or lacking a reference spectrum. For example, the human metabolome is estimated to contain around 150,000 metabolites[18], but the Human Metabolome Database[12] contains only around 74,000 metabolites (as of 1 June 2017). Thus, there are still many unknown metabolites and a true estimate of the size of the human metabolome is challenging[19]. Another complicating factor is that different organisms may have completely unique metabolomes. For instance, plants have over 45,000 known secondary metabolites[20]. Finally, there may be ambiguities in making a metabolite assignment because of chemical shift overlap or identical masses (for example, isomers). As a result, the assignment of metabolites to spectral features may be as low as 4 to 5%[21].

2.3 A Tale of Two Methods: Mass Spectrometry or Nuclear Magnetic Resonance?

Perhaps the most important choice that can be made in a metabolomics study is which instrumental platform is used. Although a wide range of instruments have been used for metabolomics, including capillary electrophoresis, infrared spectroscopy, and Raman spectroscopy, only NMR and MS are routinely used for metabolomics. NMR and MS are often applied in metabolomics investigations because of their inherent complementarity, which results from their distinct advantages and disadvantages[22]. NMR is highly reproducible and quantitative, has simple sample preparation protocols, and is able to measure analytes over a wide range of solvent conditions[23]. Despite these advantages, the main limitation of NMR is its low sensitivity, which restricts its application to measuring the most abundant metabolites in the sample (micromolar to millimolar range). This has been noted as a significant hurdle that has slowed the widespread adoption of NMR by the metabolomics community[6, 22-23]. Conversely, the high sensitivity and low detection limits of MS enable the detection of subtle metabolic changes that are invisible by NMR. With this increase in sensitivity, the detection of thousands of peaks is relatively common[24], but untargeted MS metabolomics studies often are not quantitative in nature. Since MS detectors rely on ionization processes, MS is restricted to detecting metabolites that readily ionize. Correspondingly, a significant reduction in observable metabolites may occur depending on the specifics of the sample being considered[25]. For a detailed overview of the utility of various MS detectors to metabolomics, see the review article by Dunn et al. [26]

MS also suffers from reproducibility problems since contaminants within the sample can change the ionization efficiency of metabolites[2] . Specifically, quantitation is challenging in untargeted MS since peak intensity is dependent on ionization efficiency, which varies between metabolites and also is strongly dependent on experimental conditions that may result in varying ion suppression[27]. One issue of particular relevance to MS is the relatively narrow nominal mass and mass defect distribution of the metabolome which results in significant peak overlap[28]. This can be resolved by coupling MS to a chromatographic method, most commonly LC or gas chromatography (GC), to resolve overlapping peaks and to aid in the metabolite identification based on retention time and the properties of the stationary phase. GC was the first separation technique applied to the analysis of metabolic mixtures; for example, GC-MS was used to identify biomarkers for diagnosing phenylketonuria in 1970[29]. GC-MS is particularly beneficial for the analysis of volatile metabolite mixtures since minimal sample preparation is required; in some cases, samples can be directly analyzed. Furthermore, a number of applications of GC-MS uniquely involve detecting volatile metabolites; two examples are the measurement of exhaled breath condensates for diagnosing lung cancer[30] and the monitoring of volatile paper degradation products from historic books[31]. An obvious disadvantage of GC-MS is its reliance on analyte volatility, where metabolites of low volatility or low temperature stability may be modified or destroyed[17]. Limited metabolite volatility can be overcome through the use of derivatization schemes, but derivatization is time-consuming. More importantly, differences in the efficiency of the derivatization[32] and differences in the stability of the derivatized metabolites[17]may dramatically perturb the apparent concentrations of the metabolites, possibly leading to an erroneous biological conclusion. LC was not widely used for metabolomics until the 1980s[26] and this was due to technical limitations with interfacing LC and mass spectrometers. A main advantage of LC over GC is that most metabolites can be detected intact and without modification from a deravitizing agent. Additionally, LC provides an accurate analysis of thermally unstable or reactive metabolites since the separation typically occurs at room temperature. However, the introduction of a liquid phase does introduce a higher variability in retention times[33], an increase in ion suppression due to matrix effects[34], and a lower resolution relative to GC. NMR and MS tend to observe a distinct set of metabolites from the same metabolomics sample. Consequently, there is a growing trend in metabolomics to perform tandem studies in which the same sample is analyzed by both NMR and MS[35-38]. In this manner, the coverage of the metabolome is significantly increased by taking advantage of the strengths of both methods. NMR identifies trends in metabolic alteration along core metabolic pathways and provides a context for the interpretation of the low-abundance metabolites identified by MS. Of course, the combined use of NMR and MS leads to a proportional increase in data set size with the added complexity of the simultaneous processing, analysis, and interpretation of two dissimilar data types.

2.4 Practical Concerns in Data Processing and Interpretation

Metabolomics experiments generate large data sets that require specialized tools for analysis. Numerous software packages for data pre-processing and statistical analysis are available and have been reviewed elsewhere[39-40]. Unfortunately, no single software exists that can simultaneously perform all of the critical steps needed for an analysis of a combined NMR and MS data set. Although the statistical techniques applied to NMR and MS data sets are largely the same, each technique requires a unique set of pre-processing tools and algorithms prior to modelling. For example, an NMR spectrum has to be Fourier-transformed and phased, whereas centroiding and de-isotoping are required in MS. Owing largely to these data type–specific processing requirements, newly developed software is almost exclusively restricted to one method or the other. Conversely, there has been minimal effort in developing tools capable of working with both NMR and MS data sets[39]. There are two general approaches to integrating NMR and MS data sets into a single coherent study. The first involves samples simply being independently analyzed by each method. The separate data sets then are compared in order to identify consistencies in the metabolic alterations observed by each technique. The main advantage of the approach is simplicity since it does not require any significant protocol changes. Also, the confidence of a metabolite assignment may be significantly increased if it is identified by both methods. Furthermore, a measure of internal consistency may be achieved if metabolite concentrations can be estimated by both methods. However, significant information can be lost during this process since, for example, ambiguity in peak assignments sometimes can be resolved by information from the other method. There is also a lack of statistical correlation since the data sets are independently analyzed. Although the manual curation of independent data sets is the dominant method currently used by metabolomics investigators, it also suffers from reproducibility problems due to potential biases in data interpretation (for example, metabolite assignment methods) among other issues. The second approach to combining NMR and MS data sets is to simultaneously integrate each data set into a single statistical model using a multiblock analysis. Multiblock analysis encompasses a variety of methods that combine multiple data sets prior to conventional multivariate analysis. In addition to combining multiple instrumental data sources42, multiblock analysis has been successfully employed to combine data sets from different omics disciplines[41]. Multiblock methods are preferable to independent analysis since the relative contributions of each data set still can be quantified, but importantly the larger combined data set is likely to result in models with greater predictive ability and resolving power than either method alone[42]. However, the software tools to perform multiblock analyses are crude and often rely on custom sets of pre-processing routines using multiple software packages. The lack of integrated analysis tools and software is a major roadblock in metabolomics, especially in light of the growing interest in combining NMR and MS data sets.

2.5 Recent Advances in Metabolomics

2.5.1 Dynamic Nuclear Polarization

NMR metabolomics investigations, especially those concerned with achieving a high confidence in metabolite identification, require two-dimensional NMR methods to resolve the overlap present in one-dimensional spectra. In general, this requires isotopic labelling with NMR-active nuclei like 13C and 15N because of their low natural abundance. In the last few years, dynamic nuclear polarization (DNP) has evolved from a structural biology tool in the area of solid-state NMR to have potential applications in solution-state metabolomics[43]. In DNP, a solid, frozen metabolomics sample at about 1.5 K is polarized in the presence of microwave-irradiated free-radicals, which induces a temporary hyperpolarization in spinactive nuclei through a transfer of polarization from electrons to nuclei. The sample then needs to be rapidly melted and transferred to an NMR spectrometer to take advantage of the greatly enhanced sensitivity (>10,000-fold)[44]. The dramatic increase in sensitivity avoids the need for isotopic labelling, especially for in vivo samples, and may permit the detection of low-abundance metabolites. Nevertheless, DNP experiments are limited by T1 relaxation rates, resulting in a short measurement window of the dynamically polarized samples. DNP also requires substantial hardware modifications and accessories (for example, microwave generator) to rapidly thaw and shuttle samples back and forth from the NMR spectrometer. DNP has also been applied to 13C-labeled metabolites that then are used as a tracer compound for in vivo imaging[45]. This requires close proximity of the polarizer and magnetic resonance imaging spectrometer to allow for rapid transfer, dissolution, and injection of the 13C-labeled metabolite given the relatively short T1 of 30 to 40 seconds for a 13C-labeled carboxyl group. Despite these technical obstacles, DNP has been successfully used to monitor a single metabolite (for example, pyruvate) in living tissue (for example, heart) by magnetic resonance imaging[46]. Besides the short measurement time, another challenge with the application of DNP to in vivo imaging is the limited number of 13C-labeled metabolites that can be polarized and tolerated at the concentrations needed for imaging (25 to 80 mM) and that are also a useful biological probe. In addition to pyruvate, bicarbonate, fumarate, urea, glutamine, and dehydroascorbate have been used for in vivo imaging[45]. Despite fundamental issues of reproducibility and limitations in sample preparation, DNP protocols and technology are rapidly advancing and one day could become a routine tool for metabolomics[47].

2.5.2 Disease Profiling and Personalized Medicine

Metabolomics can be used to profile an individual's responses to a drug treatment or other medical therapy by monitoring metabolite changes in readily obtainable biofluids (for example, blood and urine). A unique advantage of metabolites as biomarkers is the likely occurrence of observing a set of multiple metabolites with distinctly different concentration changes that are correlated with a disease state or treatment response. Correspondingly, multiple metabolites, instead of a single biomarker, are expected to yield a higher sensitivity and selectivity. For example, plasma baseline levels of xanthine, 2-hydroxyvaleric acid, succinic acid, stearic acid, and fructose prior to simvastatin treatment were observed to reliably predict a good or poor response in reducing low-density lipoprotein cholesterol[48]. The OPLS model yielded a 70% sensitivity and 79% specificity with a corresponding area under the receiver operating characteristic curve of 0.84. Thus, metabolomics can be used to predict whether a patient will respond to a drug in addition to being used as a semi-quantitative prognosis of disease progression. For example, in a recent study of patients with tuberculosis (TB), urine samples that were collected over the course of a 6-month period became more similar to those of a non-TB control group during the course of first-line anti-TB therapy (for example, isoniazid, ethambutol, or pyrazinamide). Metabolomics has also been successfully employed to identify serum metabolic alterations associated with psoriasis[49]. Importantly, the metabolomics results were consistent with trends previously observed in genomics and proteomics studies. The metabolome changes were observed to reverse following successful corticosteroid treatment[50]. Interestingly, the authors identified an increased demand for glutamine, which had not been previously reported in psoriasis.[50-51]

Glutamine demand is directly associated with diseases characterized by increased cellular proliferation, such as in cancers. A significant alteration in β-isosterol, which is a commonly employed herbal remedy, was also observed. Thus, metabolomics may also be used to identify a patient's use of alternative treatments outside of his or her physician's knowledge or recommendation. In this manner, metabolomics may assist in determining whether co-administration of a complementary treatment was beneficial or detrimental to a patient's therapeutic outcome.

2.6 New Trends in Data Analysis

Much of the data analysis approach in metabolomics has been largely borrowed from the field of chemometrics, which pioneered the application of PCA and PLS to chemical systems[52]. Although these are powerful statistical tools, the current trend in metabolomics data analysis is evaluating the efficacy of new algorithms and statistical methods to improve group separation and metabolite identification. PCA, PLS, and OPLS are all commonly employed by metabolomics investigators, but newer approaches, including support vector machine (SVM[53]), random forest (RF)[54], and self-organizing map (SOM)[55] algorithms, have all been recently applied to metabolomics data sets.

Despite having been formalized since 1992[56], SVM has been used extensively only in the analysis of gene microarray data, particularly due to its performance on data sets characterized by a large number of variables and few samples[57][mukerjee]. SVM is also able to identify nonlinear relationships that violate the linearity assumptions of PCA and OPLS, making it easily generalizable. SVM has been recently applied in biomarker discovery for ovarian cancer, and a model using serum-derived LC-MS spectra was able to predict disease onset with higher accuracy than the currently accepted method of CA-125 serum monitoring[58]. A major caveat of SVM is its restriction to binary classification problems: it is able to discriminate between only two experimental groups. Simply, spectra belonging to two experimental groups are represented as points in n-dimensional space, where n corresponds to the number of observed metabolites. A hyperplane then is calculated that best separates the points from the two groups. The coefficients of the calculated hyperplane are used to determine which metabolites are most important for discriminating between the two groups. Although methods have been proposed to extend SVM to multi-class problems, they are often done by breaking down the data set into an ensemble of binary groups that oversimplifies the problem and leads to uninformative models[59].

The RF algorithm is a decision tree–based method that uses random subsets of the data to construct multiple models, which then are combined to create an average model in a process known as bootstrap aggregation. In the decision tree method, samples are mapped to a target value (that is, which experimental class the sample belongs to) using a set of variable-based decision rules that separate the samples into groups corresponding to the target value. These newly formed groups can be further subdivided according to new variables, and each “branch” of separation is repeated until the samples can be fully differentiated. The major advantage of the decision tree is its imperviousness to scaling and variable normalization, an extremely common problem in metabolomics data[60]. The disadvantages include an extreme propensity for overfitting and having extremely poor generalizability that severely limits its utility. RF addresses this limitation by creating an ensemble of partial decision trees that, when combined into an overall model, reduces variance and overfitting[61]. In particular, the RF algorithm, being relatively unaffected by scaling and normalization and easily handling both large data sets and missing values, is highly adaptable to the realities of real-world data sets. A major disadvantage of RF is that the method requires extensive "tuning" of default parameters by the investigator in order to obtain the best model. Also, the resulting decision tree can be hard to visualize for large data sets[62]. RFs have shown clinical value: they have been used to determine a set of serum protein and metabolic biomarkers in prostate cancer with higher predictive accuracy than the current prostate-specific antigen biomarker[63-64]. See Gromski et al. (2015) for an excellent review of the SVM and RF algorithms that also includes comparisons with other mainstream techniques[62]. SOM is an approach similar to PCA that reduces multi-dimensional problems to a more easily interpretable low-dimensional grid to visualize natural clustering trends and groupings within a data set. SOM can be applied to the same tasks as PCA but without the biases toward high-variance metabolites. SOMs, like SVM, have the ability to detect non-linear relationships between detected metabolites68. SOMs have been successfully applied to develop biomarkers for early-stage renal cell carcinoma as well as to predict patient response to surgical intervention with a predictive accuracy of 94.74%[65]. In comparison with the other statistical methods, SOM has been severely limited in metabolomics because of a computationally intensive algorithm and the lack of a pre-packaged software, which has significantly diminished its accessibility to the wider research community[66]. Nevertheless, the usage of SOMs in metabolomics is steadily rising, and comparative analyses are beginning to demonstrate that SOMs are an acceptable alternative to more traditional clustering algorithms[67].

In addition to statistical methods applied directly to spectral profiles, identified metabolites can be used with pathway analysis[68] to understand metabolite interactions with known pathways or to discover mechanisms of action in pharmaceutical natural product research73. Metabolomics data sets can generate an overwhelming and seemingly disjointed list of metabolites, which pathway analysis aims to place into a broader biological context by assigning metabolites to relevant metabolic pathways. This is done through a number of software tools that integrate putatively identified metabolites with pathway information from various databases. For example, MetaboAnalyst 3.0 is a suite of metabolomics tools (http://www.metaboanalyst.ca), which includes modules for metabolite enrichment analysis (MSEA), metabolite pathway analysis (MetPA), and an integrated pathway analysis. The user input is typically a list of metabolites (with or without concentrations) or genes or both. MSEA provides a ranked list of potentially key metabolic pathways based on the observed number of metabolites associated with that pathway (that is, metabolite set enrichment)[69]. MetPA combines MSEA with a pathway topology analysis to provide an overall pathway analysis to identify the metabolic pathways primarily impacted in the study[68]. The integrated pathway analysis combines both metabolomics and genomics data with enrichment analysis and topology analysis to again identify the pathways (in rank order) that were primarily impacted in the study[70]. MetaboSignal (https://bioconductor.org/) is an alternative approach to pathway analysis which employs directed graphs with network topology approaches to compute centrality measures to correlate genemetabolite relationships through shortest-path distances[71]. Thus, unlike the MetaboAnalyst 3.0 tools, the output of MetaboSignal is a network map of gene-metabolite connectivities. Cytoscape (http://www.cytoscape.org/) is a generalized network interaction and visualization tool that works with a variety of data sets, including metabolomics data. Cytoscape combined with MetScape 3 (http://metscape.ncibi.org/)[72] can generate network maps similar to those of MetaboSignal from metabolomics or genomics data or both78. MetScape uses known pathways from Kyoto Encyclopedia of Genes and Genomes (KEGG)[73] and Edinburgh Human Metabolic Network (EHMN)[74] databases and gene set enrichment analysis to generate these networks in order to visualize the impacted metabolic pathways. In essence, there is some significant overlap in the capabilities of MetaboAnalyst 3.0, MetaboSignal, and Cytoscape/MetScape 3. Importantly, pathway analysis provides an interaction network that may identify centralized hubs where metabolic pathways coincide or where bottlenecks may occur. The limited connectivity or altered flow through specific metabolic nodes (that is, change in metabolic flux) may identify functionally essential biological processes[75]. These essential pathways then can be selectively targeted. By genetically or chemically restricting a potentially essential metabolic pathway, it is possible to ascertain the relevance of the pathway to a systems response to an environmental stress (that is, drug resistance) and potentially reverse or negate the response[76]. Pathway analysis also allows for the integration of multi-modal omics data, such as combining gene-expression and metabolomics data to uncover gene and protein functions. For example, metabolite profiles were integrated with genome-wide screening of single-nucleotide polymorphisms (SNPs) to identify the molecular mechanism of the NAT8 and PYROXD2 genes. Briefly, SNPs were ranked according to the strength of an association with observed metabolites. The regions where these SNPs occur on the chromosomes then were screened to determine at what position in the genome the gene/protein product responsible for the mediation is stored. With this approach, it was suggested that the NAT8 and PYROXD2 genes were responsible for mediation of serum diethylamine levels[77], a novel insight for these previously under-annotated genes.

As another illustration, transcriptomic and metabolomic data from Arabidopsis thaliana were integrated to characterize the biological response resulting from the over-expression of PAP1, a gene known to cause profound accumulation of anthocyanins and to encode a MYB transcription factor regulating flavonoid biosynthesis. The authors were able to correlate the biosynthesis of cyanidin and quercetin derivatives with a specific set of upregulated genes that enabled them to identify the function of two uncharacterized proteins: a flavonoid 3-O-glucosyltransferase and anthocyanin 5-O-glucosyltransferase[78]. Numerous tools are now available for pathway analysis of metabolomics data[73, 79-80], which will significantly improve data interpretation and simplify our understanding of biological relevance. Thus, pathway analysis is becoming a routine component of a detailed metabolomics analysis.

2.7 Concluding Thoughts: What does the future hold?

The recent advancements in metabolomics outlined herein have been shown to enhance its utility in systems biology research and to have a beneficial impact on medical research and personalized medicine. The measurement of metabolomics profiles has been shown to be useful for monitoring treatment efficacies from both pharmaceutical and surgical interventions. As our understanding of the relationship between disease state and the chemical profile of biofluids grows, metabolomics is expected to become a routine approach for monitoring disease development and progression, as a tool for disease diagnosis, and for understanding the underlying molecular mechanisms of drug resistance. Metabolite profiles could be obtained at regular intervals and screened for changes over a patient's lifetime as a diagnostic tool and a means to monitor a patient's overall health status. Some of this work has already begun; an ongoing Alphabet (parent company of Google) "moon-shot project" is a baseline study attempting to determine the inherent level of variability in human medical data that is not associated with a disease. Though still in its infancy, a similar approach using metabolomic profiles may be used to determine the inherent variability in biofluid profiles for healthy individuals. In this manner, metabolic profiles associated with disease onset and progression can be easily distinguished from the known variance in healthy individuals Some of the biggest challenges remaining in the field of metabolomics involve fundamental limits in experimental methodology. Metabolomics requires relatively high-cost instrumentation and complex data analysis and still suffers from issues of sample-to-sample variability. Although great strides in each of these areas have been made, there is still more work to be done before metabolomics can become a key and routine part of a clinical practice. Nevertheless, metabolomics continues to make important contributions to both medical research and general systems biology studies. In fact, the ability to directly measure metabolite concentration changes by using a targeted NMR or MS approach would greatly benefit investigations into a range of research areas that often are overlooked by other methods. In this manner, a metabolomics assay that targets a select and specific set of metabolites can be used to develop a highly reproducible and quantifiable assay that can be translated into a validated clinical assay.

2.8 References

1. Johnson, C. H.; Ivanisevic, J.; Siuzdak, G., Metabolomics: beyond biomarkers and towards mechanisms. (1471-0080 (Electronic)).

2. Antignac, J.-P.; de Wasch, K.; Monteau, F.; De Brabander, H.; Andre, F.; Le Bizec, B., The ion suppression phenomenon in liquid chromatography–mass spectrometry and its consequences in the field of residue analysis. *Analytica Chimica Acta* **2005,** *529* (1), 129-136.

3. Turnbaugh, P. J.; Ley, R. E.; Hamady, M.; Fraser-Liggett, C. M.; Knight, R.; Gordon, J. I., The Human Microbiome Project. *Nature* **2007,** *449*, 804.

4. ter Kuile, B. H.; Westerhoff, H. V., Transcriptome meets metabolome: hierarchical and metabolic regulation of the glycolytic pathway. *FEBS Letters* **2001,** *500* (3), 169-171.

5. Raamsdonk, L. M.; Teusink, B.; Broadhurst, D.; Zhang, N.; Hayes, A.; Walsh, M. C.; Berden, J. A.; Brindle, K. M.; Kell, D. B.; Rowland, J. J.; Westerhoff, H. V.; van Dam, K.; Oliver, S. G., A functional genomics strategy that uses metabolome data to reveal the phenotype of silent mutations. *Nature Biotechnology* **2001,** *19*, 45.

6. Gowda, G. A.; Zhang, S.; Gu, H.; Asiago, V.; Shanaiah, N.; Raftery, D., Metabolomics-based methods for early disease diagnostics. *Expert review of molecular diagnostics* **2008,** *8* (5), 617-33.

7. Theodoridis, G. A.; Gika, H. G.; Want, E. J.; Wilson, I. D., Liquid chromatography–mass spectrometry based global metabolite profiling: A review. *Analytica Chimica Acta* **2012,** *711*, 7-16.

8. Griffiths, W. J.; Koal, T.; Wang, Y.; Kohl, M.; Enot, D. P.; Deigner, H.-P., Targeted Metabolomics for Biomarker Discovery. *Angewandte Chemie International Edition* **2010,** *49* (32), 5426-5445.

9. Dai, H.; Xiao, C.; Liu, H.; Hao, F.; Tang, H., Combined NMR and LC−DAD-MS Analysis Reveals Comprehensive Metabonomic Variations for Three Phenotypic Cultivars of Salvia Miltiorrhiza Bunge. *Journal of Proteome Research* **2010,** *9* (3), 1565-1578.

10. Das, M. K.; Bishwal, S. C.; Das, A.; Dabral, D.; Badireddy, V. K.; Pandit, B.; Varghese, G. M.; Nanda, R. K., Deregulated Tyrosine–Phenylalanine Metabolism in Pulmonary Tuberculosis Patients. *Journal of Proteome Research* **2015,** *14* (4), 1947-1956.

11. Go, E. P., Database resources in metabolomics: an overview. (1557-1904 (Electronic)).

12. Wishart, D. S.; Jewison, T.; Guo, A. C.; Wilson, M.; Knox, C.; Liu, Y.; Djoumbou, Y.; Mandal, R.; Aziat, F.; Dong, E.; Bouatra, S.; Sinelnikov, I.; Arndt, D.; Xia, J.; Liu, P.; Yallou, F.; Bjorndahl, T.; Perez-Pineiro, R.; Eisner, R.; Allen, F.; Neveu, V.; Greiner, R.; Scalbert, A., HMDB 3.0—The Human Metabolome Database in 2013. *Nucleic Acids Research* **2013,** *41* (D1), D801-D807.

13. Breier, M.; Wahl, S.; Prehn, C.; Fugmann, M.; Ferrari, U.; Weise, M.; Banning, F.; Seissler, J.; Grallert, H.; Adamski, J.; Lechner, A., Targeted metabolomics identifies reliable and stable metabolites in human serum and plasma samples. (1932-6203 (Electronic)).

14. Kind, T.; Tolstikov, V.; Fiehn, O.; Weiss, R. H., A comprehensive urinary metabolomic approach for identifying kidney cancer. *Analytical Biochemistry* **2007,** *363* (2), 185-195.

15. Krishnan, P.; Kruger, N. J.; Ratcliffe, R. G., Metabolite fingerprinting and profiling in plants using NMR. *Journal of Experimental Botany* **2005,** *56* (410), 255-265.

16. Wu, H.; Southam, A. D.; Hines, A.; Viant, M. R., High-throughput tissue extraction protocol for NMR- and MS-based metabolomics. *Analytical Biochemistry* **2008,** *372* (2), 204-212.

17. Fang, M.; Ivanisevic, J.; Benton, H. P.; Johnson, C. H.; Patti, G. J.; Hoang, L. T.; Uritboonthai, W.; Kurczy, M. E.; Siuzdak, G., Thermal Degradation of Small Molecules: A Global Metabolomic Investigation. (1520-6882 (Electronic)).

18. Markley, J. L.; Bruschweiler, R.; Edison, A. S.; Eghbalnia, H. R.; Powers, R.; Raftery, D.; Wishart, D. S., The future of NMR-based metabolomics. (1879-0429 (Electronic)).

19. Wishart, D. S.; Lewis Mj Fau - Morrissey, J. A.; Morrissey Ja Fau - Flegel, M. D.; Flegel Md Fau - Jeroncic, K.; Jeroncic K Fau - Xiong, Y.; Xiong Y Fau - Cheng, D.; Cheng D Fau - Eisner, R.; Eisner R Fau - Gautam, B.; Gautam B Fau - Tzur, D.; Tzur D Fau - Sawhney, S.; Sawhney S Fau - Bamforth, F.;

Bamforth F Fau - Greiner, R.; Greiner R Fau - Li, L.; Li, L., The human cerebrospinal fluid metabolome. (1570-0232 (Print)).

20. De Luca, V.; St Pierre, B., The cell and developmental biology of alkaloid biosynthesis. (1360-1385 (Print)).

21. Dias, D. A.; Jones, O. A.; Beale, D. J.; Boughton, B. A.; Benheim, D.; Kouremenos, K. A.; Wolfender, J. L.; Wishart, D. S., Current and Future Perspectives on the Structural Identification of Small Molecules in Biological Systems. *Metabolites* **2016,** *6* (4).

22. Marshall, D. D.; Powers, R., Beyond the paradigm: Combining mass spectrometry and nuclear magnetic resonance for metabolomics. *Progress in nuclear magnetic resonance spectroscopy* **2017,** *100*, 1-16.

23. Pan, Z.; Raftery, D., Comparing and combining NMR spectroscopy and mass spectrometry in metabolomics. *Analytical and bioanalytical chemistry* **2007,** *387* (2), 525-7.

24. Dettmer, K.; Aronov, P. A.; Hammock, B. D., Mass spectrometry-based metabolomics. *Mass spectrometry reviews* **2007,** *26* (1), 51-78.

25. Copeland, J. C.; Zehr, L. J.; Cerny, R. L.; Powers, R., The applicability of molecular descriptors for designing an electrospray ionization mass spectrometry compatible library for drug discovery. *Combinatorial chemistry & high throughput screening* **2012,** *15* (10), 806-15.

26. Dunn, W. B.; Bailey, N. J.; Johnson, H. E., Measuring the metabolome: current analytical technologies. *The Analyst* **2005,** *130* (5), 606-25.

27. Annesley, T. M., Ion suppression in mass spectrometry. *Clinical chemistry* **2003,** *49* (7), 1041-4.

28. Kell, D. B., Metabolomics and systems biology: making sense of the soup. *Current opinion in microbiology* **2004,** *7* (3), 296-307.

29. Blau, K.; Cameron, H. H.; Summer, G. K., Diagnosis of phenylketonuria by gas chromatography. *Methods in medical research* **1970,** *12*, 100-5.

30. Fuchs, P.; Loeseken, C.; Schubert, J. K.; Miekisch, W., Breath gas aldehydes as biomarkers of lung cancer. *International journal of cancer* **2010,** *126* (11), 2663-70.

31. Strlic, M.; Thomas, J.; Trafela, T.; Csefalvayova, L.; Kralj Cigic, I.; Kolar, J.; Cassar, M., Material degradomics: on the smell of old books. *Anal Chem* **2009,** *81* (20), 8617-22.

32. Villas-Boas, S. G.; Smart, K. F.; Sivakumaran, S.; Lane, G. A., Alkylation or Silylation for Analysis of Amino and Non-Amino Organic Acids by GC-MS? *Metabolites* **2011,** *1* (1), 3-20.

33. Barwick, V. J., Sources of uncertainty in gas chromatography and high-performance liquid chromatography. *Journal of Chromatography A* **1999,** *849* (1), 13-33.

34. Matuszewski, B. K.; Constanzer, M. L.; Chavez-Eng, C. M., Strategies for the assessment of matrix effect in quantitative bioanalytical methods based on HPLC-MS/MS. *Anal Chem* **2003,** *75* (13), 3019-30.

35. Yanshole, V. V.; Snytnikova, O. A.; Kiryutin, A. S.; Yanshole, L. V.; Sagdeev, R. Z.; Tsentalovich, Y. P., Metabolomics of the rat lens: a combined LC-MS and NMR study. *Experimental eye research* **2014,** *125*, 71-8.

36. Bingol, K.; Bruschweiler-Li, L.; Yu, C.; Somogyi, A.; Zhang, F.; Bruschweiler, R., Metabolomics beyond spectroscopic databases: a combined MS/NMR strategy for the rapid identification of new metabolites in complex mixtures. *Anal Chem* **2015,** *87* (7), 3864-70.

37. Bingol, K.; Bruschweiler, R., NMR/MS Translator for the Enhanced Simultaneous Analysis of Metabolomics Mixtures by NMR Spectroscopy and Mass Spectrometry: Application to Human Urine. *J Proteome Res* **2015,** *14* (6), 2642-8.

38. Bingol, K.; Bruschweiler, R., Two elephants in the room: new hybrid nuclear magnetic resonance and mass spectrometry approaches for metabolomics. *Current opinion in clinical nutrition and metabolic care* **2015,** *18* (5), 471-7.

39. Misra, B. B.; van der Hooft, J. J., Updates in metabolomics tools and resources: 2014-2015. *Electrophoresis* **2016,** *37* (1), 86-110.

40. Castillo, S.; Gopalacharyulu, P.; Yetukuri, L.; Orešič, M., Algorithms and tools for the preprocessing of LC–MS metabolomics data. *Chemometrics and Intelligent Laboratory Systems* **2011,** *108* (1), 23-32.

41. Boccard, J.; Rutledge, D. N., A consensus orthogonal partial least squares discriminant analysis (OPLS-DA) strategy for multiblock Omics data fusion. *Anal Chim Acta* **2013,** *769*, 30-9.

42. Marshall, D. D.; Lei, S.; Worley, B.; Huang, Y.; Garcia-Garcia, A.; Franco, R.; Dodds, E. D.; Powers, R., Combining DI-ESI-MS and NMR datasets for metabolic profiling. *Metabolomics* **2015,** *11* (2), 391-402.

43. Dumez, J. N.; Milani, J.; Vuichoud, B.; Bornet, A.; Lalande-Martin, J.; Tea, I.; Yon, M.; Maucourt, M.; Deborde, C.; Moing, A.; Frydman, L.; Bodenhausen, G.; Jannin, S.; Giraudeau, P., Hyperpolarized NMR of plant and cancer cell extracts at natural abundance. *The Analyst* **2015,** *140* (17), 5860-3.

44. Ardenkjaer-Larsen, J. H.; Fridlund, B.; Gram, A.; Hansson, G.; Hansson, L.; Lerche, M. H.; Servin, R.; Thaning, M.; Golman, K., Increase in signal-to-noise ratio of > 10,000 times in liquid-state NMR. *Proceedings of the National Academy of Sciences of the United States of America* **2003,** *100* (18), 10158-63.

45. Dutta, P.; Martinez, G. V.; Gillies, R. J., A New Horizon of DNP technology: Application to In-vivo (13)C Magnetic Resonance Spectroscopy and Imaging. *Biophysical reviews* **2013,** *5* (3), 271-281.

46. Schroeder, M. A.; Cochlin, L. E.; Heather, L. C.; Clarke, K.; Radda, G. K.; Tyler, D. J., In vivo assessment of pyruvate dehydrogenase flux in the heart using hyperpolarized carbon-13 magnetic resonance. *Proceedings of the National Academy of Sciences of the United States of America* **2008,** *105* (33), 12051-6.

47. Bornet, A.; Maucourt, M.; Deborde, C.; Jacob, D.; Milani, J.; Vuichoud, B.; Ji, X.; Dumez, J. N.; Moing, A.; Bodenhausen, G.; Jannin, S.; Giraudeau, P., Highly Repeatable Dissolution Dynamic Nuclear Polarization for Heteronuclear NMR Metabolomics. *Anal Chem* **2016,** *88* (12), 6179-83.

48. Trupp, M.; Zhu, H.; Wikoff, W. R.; Baillie, R. A.; Zeng, Z. B.; Karp, P. D.; Fiehn, O.; Krauss, R. M.; Kaddurah-Daouk, R., Metabolomics reveals amino acids contribute to variation in response to simvastatin treatment. *PloS one* **2012,** *7* (7), e38386.

49. Armstrong, A. W.; Wu, J.; Johnson, M. A.; Grapov, D.; Azizi, B.; Dhillon, J.; Fiehn, O., Metabolomics in psoriatic disease: pilot study reveals metabolite differences in psoriasis and psoriatic arthritis. *F1000Res* **2014,** *3*, 248.

50. Sitter, B.; Johnsson, M. K.; Halgunset, J.; Bathen, T. F., Metabolic changes in psoriatic skin under topical corticosteroid treatment. *BMC dermatology* **2013,** *13*, 8.

51. Medina, M. A., Glutamine and cancer. *The Journal of nutrition* **2001,** *131* (9 Suppl), 2539S-42S; discussion 2550S-1S.

52. Wold, S.; Sjöström, M.; Eriksson, L., PLS-regression: a basic tool of chemometrics. *Chemometrics and Intelligent Laboratory Systems* **2001,** *58* (2), 109-130.

53. Heinemann, J.; Mazurie, A.; Tokmina-Lukaszewska, M.; Beilman, G.; Bothner, B., *Application of support vector machines to metabolomics experiments with limited replicates*. 2014; Vol. 10.

54. Chen, T.; Cao, Y.; Zhang, Y.; Liu, J.; Bao, Y.; Wang, C.; Jia, W.; Zhao, A., Random forest in clinical metabolomics for phenotypic discrimination and biomarker selection. *Evidence-based complementary and alternative medicine : eCAM* **2013,** *2013*, 298183.

55. Patterson, A. D.; Li, H.; Eichler, G. S.; Krausz, K. W.; Weinstein, J. N.; Fornace, A. J., Jr.; Gonzalez, F. J.; Idle, J. R., UPLC-ESI-TOFMS-based metabolomics and gene expression dynamics inspector self-organizing metabolomic maps as tools for understanding the cellular response to ionizing radiation. *Anal Chem* **2008,** *80* (3), 665-74.

56. Boser, B. E.; Guyon, I. M.; Vapnik, V. N., A training algorithm for optimal margin classifiers. In *Proceedings of the fifth annual workshop on Computational learning theory*, ACM: Pittsburgh, Pennsylvania, USA, 1992; pp 144-152.

57. Furey, T. S.; Cristianini, N.; Duffy, N.; Bednarski, D. W.; Schummer, M.; Haussler, D., Support vector machine classification and validation of cancer tissue samples using microarray expression data. *Bioinformatics (Oxford, England)* **2000,** *16* (10), 906-14.

58. Gaul, D. A.; Mezencev, R.; Long, T. Q.; Jones, C. M.; Benigno, B. B.; Gray, A.; Fernandez, F. M.; McDonald, J. F., Highly-accurate metabolomic detection of early-stage ovarian cancer. *Scientific reports* **2015,** *5*, 16351.

59. Hsu, C. W.; Lin, C. J., A comparison of methods for multiclass support vector machines. *IEEE transactions on neural networks* **2002,** *13* (2), 415-25.

60. Bryan, K.; Brennan, L.; Cunningham, P., MetaFIND: a feature analysis tool for metabolomics data. *BMC bioinformatics* **2008,** *9*, 470.

61. Breiman, L., Random Forests. *Machine Learning* **2001,** *45* (1), 5-32.

62. Gromski, P. S.; Muhamadali, H.; Ellis, D. I.; Xu, Y.; Correa, E.; Turner, M. L.; Goodacre, R., A tutorial review: Metabolomics and partial least squares-discriminant analysis--a marriage of convenience or a shotgun wedding. *Anal Chim Acta* **2015,** *879*, 10-23.

63. Fan, Y.; Murphy, T. B.; Byrne, J. C.; Brennan, L.; Fitzpatrick, J. M.; Watson, R. W., Applying random forests to identify biomarker panels in serum 2D-DIGE data for the detection and staging of prostate cancer. *J Proteome Res* **2011,** *10* (3), 1361-73.

64. Thompson, I. M.; Pauler, D. K.; Goodman, P. J.; Tangen, C. M.; Lucia, M. S.; Parnes, H. L.; Minasian, L. M.; Ford, L. G.; Lippman, S. M.; Crawford, E. D.; Crowley, J. J.; Coltman, C. A., Jr., Prevalence of prostate cancer among men with a prostate-specific antigen level < or =4.0 ng per milliliter. *The New England journal of medicine* **2004,** *350* (22), 2239-46.

65. Zheng, H.; Ji, J.; Zhao, L.; Chen, M.; Shi, A.; Pan, L.; Huang, Y.; Zhang, H.; Dong, B.; Gao, H., Prediction and diagnosis of renal cell carcinoma using nuclear magnetic resonance-based serum metabolomics and self-organizing maps. *Oncotarget* **2016,** *7* (37), 59189-59198.

66. Lloyd, G. R.; Wongravee, K.; Silwood, C. J. L.; Grootveld, M.; Brereton, R. G., Self Organising Maps for variable selection: Application to human saliva analysed by nuclear magnetic resonance spectroscopy to investigate the effect of an oral healthcare product. *Chemometrics and Intelligent Laboratory Systems* **2009,** *98* (2), 149-161.

67. Wijetunge, C. D.; Li, Z.; Saeed, I.; Bowne, J.; Hsu, A. L.; Roessner, U.; Bacic, A.; Halgamuge, S. K., Exploratory analysis of high-throughput metabolomic data. *Metabolomics* **2013,** *9* (6), 1311-1320.

68. Xia, J.; Wishart, D. S., MetPA: a web-based metabolomics tool for pathway analysis and visualization. *Bioinformatics (Oxford, England)* **2010,** *26* (18), 2342-4.

69. Xia, J.; Wishart, D. S., MSEA: a web-based tool to identify biologically meaningful patterns in quantitative metabolomic data. *Nucleic Acids Res* **2010,** *38* (Web Server issue), W71-7.

70. Xia, J.; Sinelnikov, I. V.; Han, B.; Wishart, D. S., MetaboAnalyst 3.0--making metabolomics more meaningful. *Nucleic Acids Res* **2015,** *43* (W1), W251-7.

71. Rodriguez-Martinez, A.; Ayala, R.; Posma, J. M.; Neves, A. L.; Gauguier, D.; Nicholson, J. K.; Dumas, M. E., MetaboSignal: a network-based approach for topological analysis of metabotype regulation via metabolic and signaling pathways. *Bioinformatics (Oxford, England)* **2017,** *33* (5), 773-775.

72. Basu, S.; Duren, W.; Evans, C. R.; Burant, C. F.; Michailidis, G.; Karnovsky, A., Sparse network modeling and metscape-based visualization methods for the analysis of large-scale metabolomics data. *Bioinformatics (Oxford, England)* **2017,** *33* (10), 1545-1553.

73. Kanehisa, M.; Goto, S.; Sato, Y.; Kawashima, M.; Furumichi, M.; Tanabe, M., Data, information, knowledge and principle: back to metabolism in KEGG. *Nucleic Acids Res* **2014,** *42* (Database issue), D199-205.

74. Ma, H.; Sorokin, A.; Mazein, A.; Selkov, A.; Selkov, E.; Demin, O.; Goryanin, I., The Edinburgh human metabolic network reconstruction and its functional analysis. *Molecular systems biology* **2007,** *3*, 135.

75. Jeong, H.; Mason, S. P.; Barabasi, A. L.; Oltvai, Z. N., Lethality and centrality in protein networks. *Nature* **2001,** *411* (6833), 41-2.

76. Gaupp, R.; Lei, S.; Reed, J. M.; Peisker, H.; Boyle-Vavra, S.; Bayer, A. S.; Bischoff, M.; Herrmann, M.; Daum, R. S.; Powers, R.; Somerville, G. A., Staphylococcus aureus metabolic adaptations

during the transition from a daptomycin susceptibility phenotype to a daptomycin nonsusceptibility phenotype. *Antimicrobial agents and chemotherapy* **2015,** *59* (7), 4226-38.

77. Nicholson, G.; Rantalainen, M.; Li, J. V.; Maher, A. D.; Malmodin, D.; Ahmadi, K. R.; Faber, J. H.; Barrett, A.; Min, J. L.; Rayner, N. W.; Toft, H.; Krestyaninova, M.; Viksna, J.; Neogi, S. G.; Dumas, M. E.; Sarkans, U.; Donnelly, P.; Illig, T.; Adamski, J.; Suhre, K.; Allen, M.; Zondervan, K. T.; Spector, T. D.; Nicholson, J. K.; Lindon, J. C.; Baunsgaard, D.; Holmes, E.; McCarthy, M. I.; Holmes, C. C., A genome-wide metabolic QTL analysis in Europeans implicates two loci shaped by recent positive selection. *PLoS genetics* **2011,** *7* (9), e1002270.

78. Tohge, T.; Nishiyama, Y.; Hirai, M. Y.; Yano, M.; Nakajima, J.; Awazuhara, M.; Inoue, E.; Takahashi, H.; Goodenowe, D. B.; Kitayama, M.; Noji, M.; Yamazaki, M.; Saito, K., Functional genomics by integrated analysis of metabolome and transcriptome of Arabidopsis plants over-expressing an MYB transcription factor. *The Plant journal : for cell and molecular biology* **2005,** *42* (2), 218-35.

79. Vastrik, I.; D'Eustachio, P.; Schmidt, E.; Gopinath, G.; Croft, D.; de Bono, B.; Gillespie, M.; Jassal, B.; Lewis, S.; Matthews, L.; Wu, G.; Birney, E.; Stein, L., Reactome: a knowledge base of biologic pathways and processes. *Genome Biol* **2007,** *8* (3), R39.

80. Caspi, R.; Billington, R.; Ferrer, L.; Foerster, H.; Fulcher, C. A.; Keseler, I. M.; Kothari, A.; Krummenacker, M.; Latendresse, M.; Mueller, L. A.; Ong, Q.; Paley, S.; Subhraveti, P.; Weaver, D. S.; Karp, P. D., The MetaCyc database of metabolic pathways and enzymes and the BioCyc collection of pathway/genome databases. *Nucleic Acids Res* **2016,** *44* (D1), D471-80.

81. Berger, B.; Peng J Fau - Singh, M.; Singh, M., Computational solutions for omics data. (1471-0064 (Electronic)).

82. Chawade, A.; Alexandersson, E.; Levander, F., Normalyzer: A Tool for Rapid Evaluation of Normalization Methods for Omics Data Sets. *Journal of Proteome Research* **2014,** *13* (6), 3114-3120.

Chapter 3: The role of Normalization in Metabolomics

3.1 Introduction

High-throughput facilities continue to improve the acquisition and throughput of OMICS experiments (e.g., genomics, transcriptomics, proteomics, and metabolomics), which has resulted in the rapid accumulation of large amounts of data. These massive datasets have enabled the detection and quantification of thousands of genes, proteins, and metabolites across various biological samples. Accordingly, OMICs data has significantly contributed to a variety of fields including drug discovery[1], personalized medicine[2], nutrition[3] and environmental studies[4]. Perturbations or variance are inherent to all experimental datasets and come from a variety of sources such as biological variability, instrument instability, and inconsistency in sample handling and preparation. For example, the number of cells harvested, the mass of tissue collected, or the amount of urine produced may vary significantly across all of the biological replicates. These unavoidable variations may mask the real biological signals present in the samples, which, in turn, complicates the reliability and accuracies of all downstream quantitative analyses[5]. Accordingly, the preprocessing of OMICs data is a critical step and involves minimizing undesirable noise to make all subsequent analyses more robust, accurate, and precise[6]. One crucial preprocessing step is the normalization of data, which has been shown to effectively reduce systematic noise in OMICs datasets[7]. Normalization of OMICS datasets can be accomplished using a variety of methods[8,9]. But, the proper choice depends on data characteristics and the sources of variation that needs correcting. How well a specific normalization technique performs in reducing these extraneous biases is still an open question. Accordingly, identifying an optimal normalization technique is still a common issue encountered throughout the OMICs fields. For example, in genomics, differences in sequencing length (library size), gene length, or guanine–cytosine content may lead to data variance and a false interpretation of gene expression variability[10]Thus, an appropriate normalization method needs to eliminate these sources of variance to ensure an accurate measure of gene expression levels. To address this issue, Choe et al. examined four popular normalization methods routinely used in genomics that included: constant sum, rank-invariant, LOcally Estimated Scatterplot Smoothing (LOESS), and quantile[11].The normalization algorithms were compared using RNA-microarray data. The LOESS normalization algorithm assumes a non-linear relationship and uses a local regression approach to adjust signal intensity and noise. Incorporating LOESS normalization into the analysis of the RNA-microarray data yielded superior results relative to the other normalization techniques. LOESS improved the detection of true differentially expressed genes as evident by the largest area under the receiver operating characteristic (ROC) curve. Similarly, Callister et al. evaluated four normalization techniques routinely used in proteomics[12]. Central tendency, linear regression, locally weighted regression, and quantile normalization algorithms were compared using three sets of samples representing different levels of data complexity. The linear regression normalization algorithm was identified as the top performer since it exhibited the largest reduction in extraneous variability while also maintaining the highest reproducibility as measured by both pooled estimate of variance and a median coefficient of variance. Metabolomics characterizes both the identity and the quantity of metabolites present in a biological sample[5]. Since metabolites are a direct product of cellular processes, the metabolome is able to accurately capture the current state of the system. Thus, even subtle changes in metabolite concentrations may provide important insights into disease progression[13], drug resistance[14], or a response to numerous stress factors (e.g., environmental toxins, nutrient limitation, genetic mutation, etc.)[15–17]. Unfortunately, like genomics and proteomics, these metabolite differences are easily obscured by the natural variance that occurs between biological replicates or by inconsistencies in sample sizes. Furthermore, since nuclear magnetic resonance (NMR) spectroscopy[5] is routinely used to monitor the metabolome, instrument instability and experimental factors such as changes in pH, temperature, ionic strength or even sample composition may lead to unintended signal variance[6]. Such non-biologically induced perturbations are likely to mask the true biological signals in the data and complicate the data analysis process. Again, normalization is a necessary requirement to minimize these undesirable variations and to increase the accuracy and reliability of all subsequent data analyses. A variety of procedures are currently available to normalize NMR metabolomics data[8,16]. Since each algorithm addresses systematic variations in a different manner, the correct choice of a normalization scheme can be challenging. For example, some normalization algorithms aim to remove unwanted noise by minimizing inter-sample variation such as probabilistic quotient[6] and cubic splines methods[18], while others such as unit variance or Pareto (often referred to as scaling), aim to adjust the variance of spectral features so that all peaks are equally weighted when used to construct multivariate models such as principal components analysis (PCA). Since these algorithms were developed with different underlying assumptions, each method confers a unique set of advantages and disadvantages. For example, Craig et al. , demonstrated that while constant sum normalization adequately preserves signal quality, it can change the underlying correlations between peaks and generate artifacts[19]. Thus, constant sum may confound interpretations when used incorrectly. A comparative analysis of normalization schemes by Kohl et al. determined that quantile normalization significantly outperforms other approaches in both minimizing inter-sample standard deviation and accurately preserving fold change information[5]. However, it was also noted that the performance of quantile normalization was only truly realized for large datasets ($n \geq 50$) and offers no significant performance benefits on more modestly sized datasets. The diversity of normalization algorithms and the lack of a clear consensus has provided the motivation to conduct a thorough and quantitative evaluation of normalizing methods currently available to the metabolomics community through our MVAPACK software package[20]. MVAPACK is open source software (http:// bionmr.unl.edu/mvapack.php) that includes a complete set of functions for data loading, preprocessing, modeling, and validation of NMR metabolomics datasets. MVAPACK also includes the following normalization methods: probabilistic quotient (PQ)[6] , histogram matching (HM)[21], standard normal variate (SNV)[22], multiplicative scatter correction (MSC)[23], quantile (Q)[5], natural cubic splines (CSpline)[18], smoothing splines (SSpline)[24], constant sum (CS) and region of interest (ROI)[6]. Our phase-scatter correction (PSC) algorithm is also available in MVAPACK, but was not included in this comparison since PSC was previously discussed in detail [25]. The normalization methods were compared using simulated and experimental NMR datasets with various levels of added noise and dilution factors[26]. Their performances were evaluated based on an ability to recover the intensities of the true spectral peaks and the reproducibility of true classifying features from orthogonal projections to latent structures—discriminant analysis (OPLS-DA) model[27]. In this manner, the normalization methods were evaluated based upon expected outcomes for routine metabolomics study: (i) the ability to eliminate irrelevant signal variance due to dilution factors and noise; and (ii) the ability to produce a predictive model that correctly identifies the real group-dependent variants. Our analysis indicates that of the normalization algorithms evaluated, PQ and CS performed the best in the analysis of noisy one-dimensional (1D) NMR metabolomics datasets.

3.2 Methodology

The performance of each normalization method was assessed using two distinct datasets: (i) simulated spectral data and (ii) a previously described experimental data set of 1D 1 H NMR spectra of various coffee samples[26].All of the analyses were conducted using our MVAPACK[20] software package. All of the figures were generated using the R software package[28].

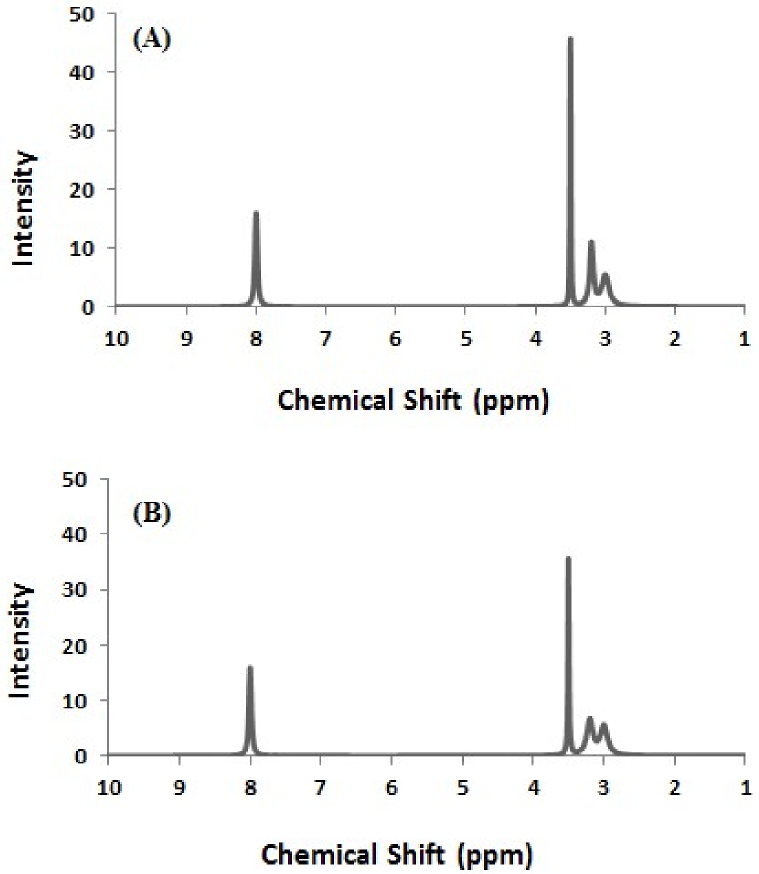


Figure 1. The simulated reference spectrum used for (A) group 1 and (B) for group 2. The two spectra contain the same number of peaks at the same chemical shifts. The only difference between the spectra is the relative peak intensities.

3.3 Simulation of a 1D 1H NMR Metabolomics Dataset

The simulated dataset consisted of 50 spectra in which each spectrum contained 901 spectral features. The set of spectra were divided into two separate groups. Each group consisted of 25 spectra that were randomly generated from a reference spectrum. The reference spectrum for each group was independently simulated from the Cauchy distribution[29], but with different parameters. Each reference spectrum contains four peaks located at chemical shifts of 3, 3.2, 3.5, and 8 ppm, respectively. The peak intensities differ between the four peaks and between the two reference spectra as illustrated in Fig. 1. The 25 spectra per group were generated from the reference spectrum by the addition of a minimal amount of Gaussian noise (Mean = 0, SD = 0.001). These two sets of 25 spectra, which correspond to group 1 and group 2, were combined to define the simulated reference dataset X0 (N=50, K=901). The simulated reference dataset X0 was then used to generate eight noise-added simulated sets (Xi) (Fig. S1) with i = 1, 2, …, 8 (Table 1) according to Eq. (1):

$$\boldsymbol{X_i} = \boldsymbol{F_i} * (\boldsymbol{X}_0 + \boldsymbol{E_i}) \tag{1}$$

where Fi is a 50×1 vector of dilution factors generated from a uniform distribution for the ith set, Ei is a matrix of independent Gaussian noise distributed with mean 0 and standard deviation σ i for the ith set, and * presents element-wise multiplication. The value of σ i ranged from 0.1 to 5 which produced a systematic increase in noise for the dataset. The CS, PQ, HM, SNV, MSC, ROI, Q, CSpline, and SSpline normalization methods were then separately applied to each noise-added set (Xi) to obtain normalized set (X̃ i). An OPLS-DA model was then generated from each normalized set (X̃ i). Two-component OPLS-DA models were calculated to obtain the first component loadings to compare the performance of the normalization approaches.

Table 1: Parameters used to generate the noise-added simulated spectra

Set	Dilution Factors ($\boldsymbol{F}$)[a]	Standard Deviation (σ)[b]	Percent Added Noise
S1	$\sim Unif(0.9, 1.1)$	0.1	5%
S2	$\sim Unif(0.9, 1.1)$	0.2	10%
S3	$\sim Unif(0.8, 1.2)$	0.4	20%
S4	$\sim Unif(0.5, 1.5)$	1	50%
S5	$\sim Unif(0.3, 1.7)$	1.4	70%
S6	$\sim Unif(0.1,1.9)$	1.8	90%
S7	$\sim Unif(0.01, 2.5)$	2.5	100%
S8	$\sim Unif(0.001, 5)$	4	200%

[a] A dilution factor was randomly selected from the indicated range of values.

[b] The value of standard deviation used to generate a Gaussian distribution of noise.

3.4 Experimental 1D 1H NMR Metabolomics Dataset

A data matrix of 32 1D 1 H NMR spectra from a publicly available coffees dataset was used to further evaluate the normalization algorithms[26].The coffees dataset contains two groups defined as light and medium decaffeinated coffee consisting of 16 1D 1 H NMR spectra per group. Each spectrum contains 284 spectral features. We applied the same procedures as described above to generate the noise-added experimental dataset. Specifically, the original coffees dataset of 32 1D 1 H NMR experimental spectra was designated as the reference data set Y0 (N=32, K=284). The reference data set Y0 was then used to generate seven simulated sets (Yi) with i = 1, 2, …, 7 (Table 2) according to Eq. 2:

$$\boldsymbol{Y_i} = \boldsymbol{F_i} * (\boldsymbol{Y}_0 + \boldsymbol{E_i}) \qquad (2)$$

where Fi is a 32×1 vector of dilution factors generated from a uniform distribution for the ith set, Ei (N=32, K=284) is a matrix of independent Gaussian noise distributed with mean 0 and standard deviation σ i for the ith set, and * presents element-wise multiplication. The value of σ i ranged from 2.3 × 10−7 to 10−5 which produced a systematic increase in noise while also mimicking the relative variance in the noise present in the coffees dataset.

Table 2. Parameters used to generate the noise-added coffees dataset

Set	Dilution Factors ($\boldsymbol{F}$)[a]	Standard Deviation (σ)[b]	Percent Added Noise
C1	$\sim Unif(0.9, 1.1)$	2.3×10^{-7}	5%
C2	$\sim Unif(0.8, 1.2)$	4.6×10^{-7}	10%
C3	$\sim Unif(0.5, 1.5)$	9.3×10^{-7}	20%
C4	$\sim Unif(0.3, 1.7)$	2.3×10^{-6}	50%
C5	$\sim Unif(0.1, 1.9)$	5×10^{-6}	100%
C6	$\sim Unif(0.01,2.5)$	8×10^{-6}	170%
C7	$\sim Unif(0.001, 5)$	10^{-5}	200%

[a] A dilution factor was randomly selected from the indicated range of values.

[b] The value of standard deviation used to generate a Gaussian distribution of noise.

3.5 Summaries of Employed Normalization Procedures

Constant sum: Each spectrum of the data matrix was divided by its own integral[6].

Probabilistic quotient The normalization factor was the most probable quotient between the signals of the corresponding spectrum and the reference spectrum[6]. The reference spectrum was chosen as the median spectrum of the spectral set. Each spectrum in the dataset was divided by this normalization factor to obtain the normalized spectrum.

Histogram matching: Raw spectra were log transformed prior to normalization. Similar to PQ, the target reference spectrum was the median spectrum of the dataset. Histograms for each sample spectrum and target spectrum were obtained on prespecified intensity intervals. A dilution factor was then chosen to minimize the differences between each sample spectrum histogram and the target histogram[21].The new normalized spectrum was generated by multiplying each original spectrum by the corresponding dilution factor.

Standard normal variate: Each sample spectrum in the dataset was centered prior to normalization. The standard deviation of each spectrum was calculated as a normalization factor[22]. A new normalized dataset was then obtained by dividing each original spectrum by its corresponding normalization factor.

Multiplicative scatter correction: The normalization factors were least squares estimates obtained by regressing each sample spectrum onto the reference spectrum[23].The reference spectrum was the mean spectrum. The ordinary least squares of the regression parameters were used to correct the spectral intensities.

Region of interest: Each sample spectrum of the dataset was normalized to a specified spectral region where its integral was set to one. Each sample spectrum was then normalized relative to the most intense peak in the spectrum.

Quantile: The goal of this quantile normalization method was to obtain an identical distribution of intensities for all of the spectral features[5].First, the mean spectrum was calculated for the data set. The intensities of all features in each sample spectrum were then replaced by the mean intensities in accordance with their quantile orders.

Natural cubic splines: The CSpline method normalized each sample spectrum to the target spectrum. The target spectrum was calculated using the non-linear arithmetic mean of the data set. Depending on the type of data, a geometric mean may also be used [5].A set of 100 quantiles was taken from both the sample spectrum and the target spectrum. The quantiles were then fitted to a natural cubic spline to obtain parameter estimates, which were used for interpolations. The process was repeated five times. For each iteration, a small offset was added to the quantiles before refitting with a natural cubic spline to obtain new interpolations. The set of interpolations were averaged to obtain the normalized spectrum.

Smoothing splines: SSpline is similar to CSpline, but the SSpline algorithm adds more quantiles toward the tail end of the spectrum. The most intense spectral features are located in this region of the spectrum. Moreover, the quantiles are fitted with a smoothing spline that includes a penalty parameter to avoid overfitting. The predicted feature intensities were then used as the normalized intensities.

3.6 Evaluation criteria

Regardless of the type of approach used to address dataset bias or variance, an optimal normalization procedure should reduce any unwanted noise while still preserving the true biological signals. In other words, a necessary condition to retain the true signals is the ability to recover the original peak intensities after removing noise. In this regards, it should be possible to evaluate the relative performance of normalization methods based on how well the algorithms handle increasingly noisy spectra. As the reference set is exposed to increasing amounts of noise, some (or all) of the normalization algorithms would be expected to fail to recover the original peaks intensities. Thus, the peak recovery criteria served as a means to filter-out poorly performing normalization procedures prior to proceeding with the second evaluation criteria. A multivariate statistical model, such as PCA or OPLS, is typically employed to identify spectral features that separate the different groups in the dataset. These spectral features are intrinsic to the dataset. Accordingly, any properly normalized dataset should reproduce these true set of features. The first component loadings extracted from an OPLS-DA model contains the weights of the spectral features that contribute the most to separating the groups. Simply, the first component loadings identify the most-important group-dependent features. Thus, an OPLS-DA model was generated to obtain the first component loadings associated with each normalization method. Only the top performing normalization methods were used to generate an OPLS-DA model. The top performing normalization methods were identified based on the peak recovery criteria. Pearson correlation coefficients were calculated between the loadings of each normalized dataset and the true loadings set. The Pearson correlation coefficients provide a means to measure the reproducibility of the true classifying spectral features produced by each normalization algorithm.

3.6.1 Peak recovery

After sequentially normalizing each noisy data matrix using the nine normalization methods, the intensity of each peak in each spectrum of the normalized set (X̃ i) was compared T. Vu et al. 1 3 108 Page 6 of 10 to the true original spectrum (X0) to measure the recovery of peak intensities (rpi j i). For each spectrum from the normalized data matrix (X̃ i), the recovery of the jth peak was calculated according to this Eq. 3:

$$rpi_i^j = \left(1 - \frac{\left|I_i^j - I_0^j\right|}{\max(|I_0^j|,\left|I_i^j\right|)}\right) \quad (3)$$

where I j i and I j 0 are the intensities of the jth peak from X̃ i and X0, respectively. In this manner, rpij i will range from 0 to 1 regardless of the relative magnitudes of I j i and I j 0. This process was repeated for every peak in each spectrum. The mean recovery and standard error were calculated and reported for each normalized set.

3.6.2 Pearson correlation coefficients

The coffees noisy data matrix (Yi) was only normalized using the top performing algorithms identified from the peak recovery criteria. An OPLS-DA model was generated for each normalized coffees data matrix (Ỹi) and also the original coffees data set (Yo). The datasets were scaled with Pareto scaling prior to calculating the OPLS-DA models. The first component loadings from each OPLS-DA model were then used to calculate a Pearson correlation coefficient between the true backscale loadings vector (po) from the original coffees data set (Yo) and the backscale loadings vector (pi) from each normalized coffees noisy data matric (Ỹi). The Pearson correlation coefficients were calculated according to Eq. 4:

$$r_i = \frac{\sum_{k=1}^{K}(p_i^k-\bar{p}_i)(p_0^k-\bar{p}_0)}{\sqrt{\sum_{k=1}^{K}(p_i^k-\bar{p}_i)^2\ \sum_{k=1}^{K}(p_0^k-\bar{p}_0)^2}} \tag{4}$$

where K denotes the number of spectral features; p̄i is the mean loading of vector pi ; pk i is the kth loading of vector pi ; p̄ 0 is the mean loading of vector p0; and pk 0 is the kth loading of vector p0. This process was repeated 100 times. The mean correlation coefficients and standard error were calculated for each normalized set.

3.7 Results and discussion

The two reference NMR spectra displayed in Fig. 1 were used to generate eight noise-added simulated metabolomics datasets consisting of 25 spectra for each of the two groups (Fig. S1). Accordingly, each simulated dataset contained a total of 50 spectra. The total signal variance in each dataset was defined by the amount of Gaussian noise added and by the dilution factors listed in Table 1. The simulated NMR metabolomics datasets were then normalized using each of the nine normalization methods (i.e., CS, CSpline, HM, MSC, PQ, Q, ROI, SNV, and SSpline). A peak recovery was calculated for each dataset according to Eq. 3. The peak recovery compares each of the normalized dataset to the original reference NMR spectra (Fig. 1). The peak recoveries for each normalized dataset are plotted in Figs. 2 and 3. As expected, the efficiency of peak recovery decreases with increasing signal variance regardless of the normalization method. As illustrated in Fig. 2, most of the normalization methods achieve nearly 100% peak recovery (96 to 99%) under conditions of modest signal variance (S1 and S2). The most notable outlier is HM, which achieved a peak recovery of only 20–28%. This extremely poor performance suggests that HM should be avoided and not used for the normalization of NMR metabolomics data.

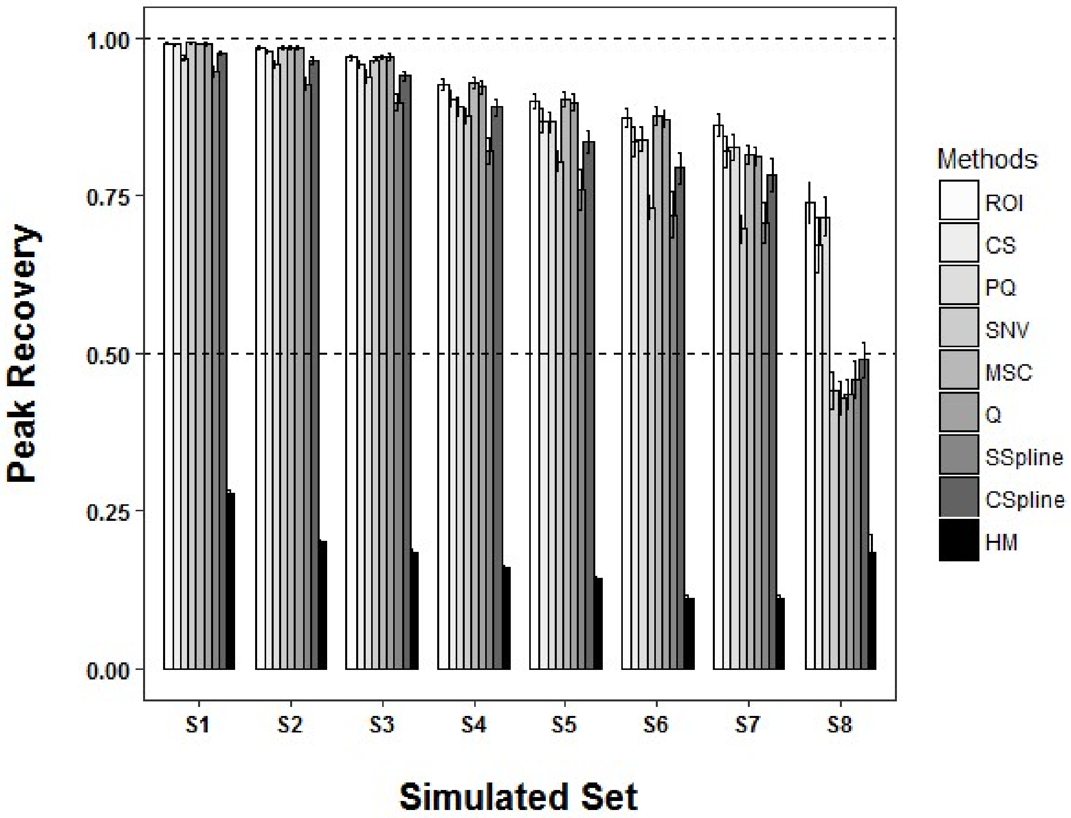


Figure 2. A plot of the recovery of peak intensities (**eqn. 3**) for the 9 normalization methods after being applied to the 8 (**S1 to S8**) simulated datasets listed in **Table 1**. The total signal variance due to the amount of added Gaussian noise and the magnitude of the dilution factor increases from **S1** to **S8**. The horizontal dashed lines represent a full recovery at 100% and partial recovery at 50%. Each bar represents the mean peak recovery and the error bars represent ±2*standard error of the mean.

While significantly better than HM, SSpline also performed consistently below average with a peak recovery range of 93–95%. PQ was modestly below the best performers with a peak recovery range of 96–97%. Conversely, ROI, CS, SNV, MSC, and Q, recovered at least 98% of the peak intensities under conditions of modest signal variance. A further separation in algorithm performance was apparent as the signal variance was progressively increased. SSpline continued to perform worse than average, but from simulated set S5 forward the performance of SNV had also significantly declined to match SSpline. Similarly, from simulated set S6, CSpline had fallen below the average performance of the other normalization methods. In fact, as the amount of signal variance was increased to the highest level (S8), the peak recoveries for CSpline, HM, MSC, Q, and SSpline all fell below 50%. Conversely, CS, PQ and ROI maintained a peak recovery of around 70% (67–74%).

Accordingly, the peak recovery results suggest that the CS, PQ and ROI were the most robust normalization methods and were able to maintain a maximal peak recovery as a function of signal variance (Fig. 3). Pairwise Student's t tests of the mean peak recovery values at the highest signal variance level (S8) yield a maximum p-value of <2.8×10−13 between the CS, PQ, ROI algorithms and the other normalization methods. To further investigate the individual impact of Gaussian noise and dilution factors on peak recovery, the simulation was repeated for the three top performing normalization methods (i.e., CS, PQ and ROI). Instead of simultaneously varying both Gaussian noise and the dilution factors as listed in Table 1, the simulation was repeated with either Gaussian noise or the dilution factor held constant at S1 values. The combined average peak recovery values for CS, PQ and ROI normalized datasets are plotted as a function of added Gaussian noise or dilution factor in Fig. 4. This simulation yielded an unexpected result. The performance of the normalization method was essentially unaffected by the dilution factor. Near perfect peak recovery was obtained even for the highest dilution factor. Instead, the normalization performance was strictly dependent on the level of Gaussian noise added to the spectra.

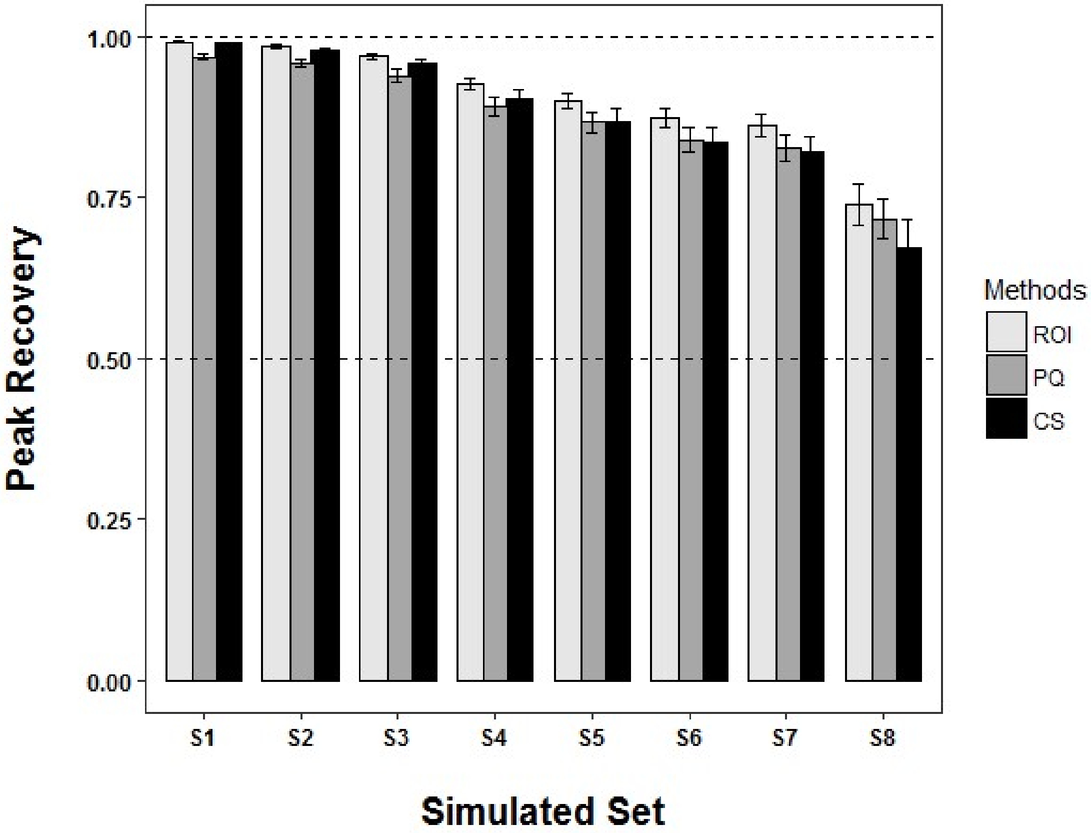


Figure 3. A plot of the recovery of peak intensities (**eqn. 3**) for the three top performing normalization methods after being applied to the 8 (**S1 to S8**) simulated datasets listed in **Table 1**. The total signal variance due to the amount of added Gaussian noise and the magnitude of the dilution factor increases from **S1** to **S8**. The horizontal dashed lines represent a full recovery at 100% and partial recovery at 50%. Each bar represents the mean peak recover and the error bars represent ±2*standard error of the mean.

However, it is important to note that normalization methods also rely on good peak alignment, spectral phasing, baseline correction and solvent suppression in order to perform well. Accordingly, the simulations reported herein were restricted to well-behaved datasets. While being able to accurately reconstitute peak intensity is an important attribute of a normalization algorithm, the proper identification of group-defining spectral features is still a vital necessity. In essence, are biologically-relevant metabolic differences still being correctly identified regardless of the natural signal variance? Does a PCA or OPLS scores plot yield statistically relevant group separations and do the loadings identify the "true" metabolic differences between the groups? To address this issue, the CS, PQ and ROI normalization methods were further evaluated based on the reproducibility of OPLS-DA models as a function of increasing signal variance. An experimental coffees dataset previously used to investigate PCA and OPLS model stability[26] was employed to generate OPLS-DA models using the CS, PQ and ROI normalization methods. Specifically, the coffee dataset consists of 32 1D 1 H NMR spectra for two groups of observations (light and medium decaffeinated coffees). The coffees dataset was modified with Gaussian noise and a dilution factor (Fig. S2) as outlined in Table 2. Consistent with our prior observations[26] the two coffee groups become indistinguishable with an increase in signal variance. Importantly, the estimated loadings from the corresponding OPLS-DA model are less correlated to the true loadings (Fig. 5) with increasing signal variance. Notably, at minimal to moderate signal variance levels (C1 to C3), the PQ and ROI normalization methods perform almost identically and significantly better than CS. But, as the amount of signal variance increased significantly (C4 to C7), the OPLS-DA model was no longer valid with the ROI normalization technique; and the loadings correlation, not surprisingly, decreased dramatically. Similarly, the standard errors of mean loadings correlation coefficients increased significantly for ROI compared to the negligible values observed for CS and PQ (ranged from 0.0003 to 0.008). Interestingly, despite CS initially performing worse than PQ, there was no difference in the loadings correlation between PQ and CS at C4. Furthermore, CS outperformed PQ at the highest signal variance levels (C5 to C7). But, the loadings correlations still decreased linearly with increasing signal variance following CS or PQ normalization. The loss of a correlation to the true loadings was still substantial and would likely lead to incorrect biological interpretations. A similar set of results was obtained for the simulated dataset (Fig. S3). In total, our analysis suggest that CS and PQ are the most robust normalization techniques and are able to compensate, at least partly, for large signal variance. Both CS and PQ maintained the highest level of peak recovery and the highest correlation between backscaled loadings. Notably, PQ was the most robust normalization technique at low to moderate noise levels while CS was slightly better at compensating for larger signal variance. A combined analysis of the peak recovery and OPLS-DA backscaled loadings data provides some further guidance for designing and executing an NMR metabolomics study. As we have noted previously[26,30,31], noise is detrimental to the accurate and reliable analysis of metabolomics data using multivariate statistical techniques such as PCA and OPLS. The results reported herein further support the negative impact of noise on the analysis of NMR metabolomics data.

As evident in Fig. 4, a dilution factor had no appreciable impact on the performance of a normalization method. Instead, all variance in the performance of the normalization methods was due to noise. Furthermore, most of the normalization methods performed equally-well in regards to peak recovery and loadings correlation for added noise levels up to about 20%. The lone exception is HM, which should be avoided. A significant decay in performance occurred when >20% of noise was added to either the simulated or experimental dataset. Accordingly, an experimental NMR dataset that exhibits greater than 20% noise is a serious concern and the resulting chemometrics model is highly suspect. In essence, our analysis sets a minimum criterion for maintaining noise (defined by a standard Gaussian distribution) at below 20% for a valid metabolomics dataset.

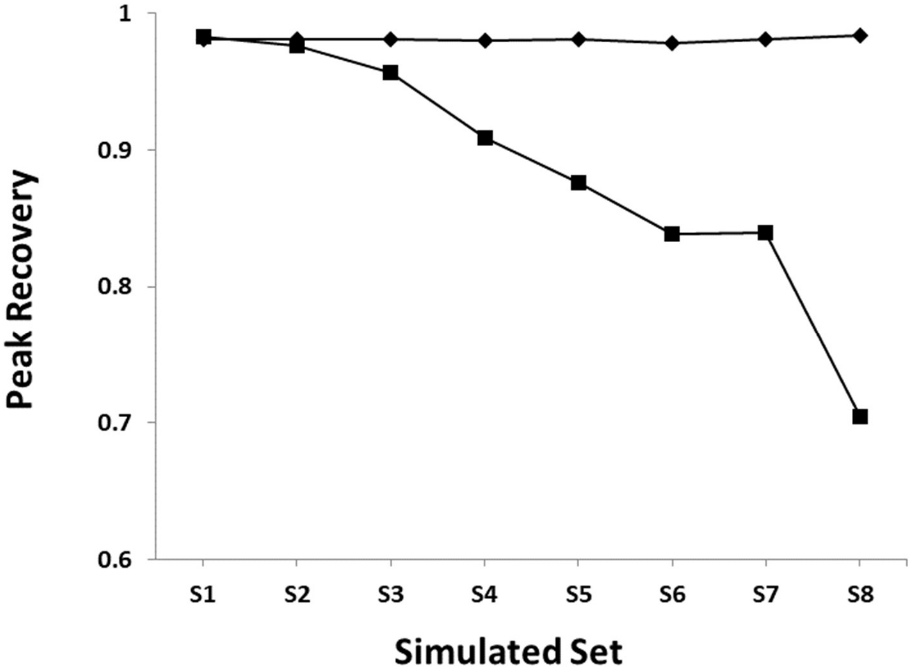


Figure 4. A plot of the average peak recovery calculated from the three top-performing normalization methods (CS, PQ, and ROI). Datasets were regenerated according to the scheme described in **Table 1** but containing only a dilution factor (♦) or the addition of Gaussian noise (■). The dilution factor or added Gaussian noise was held constant at **S1** values when the other parameter was varied. The peak recovery decreases with additive noise, but is unaffected by dilution factor.

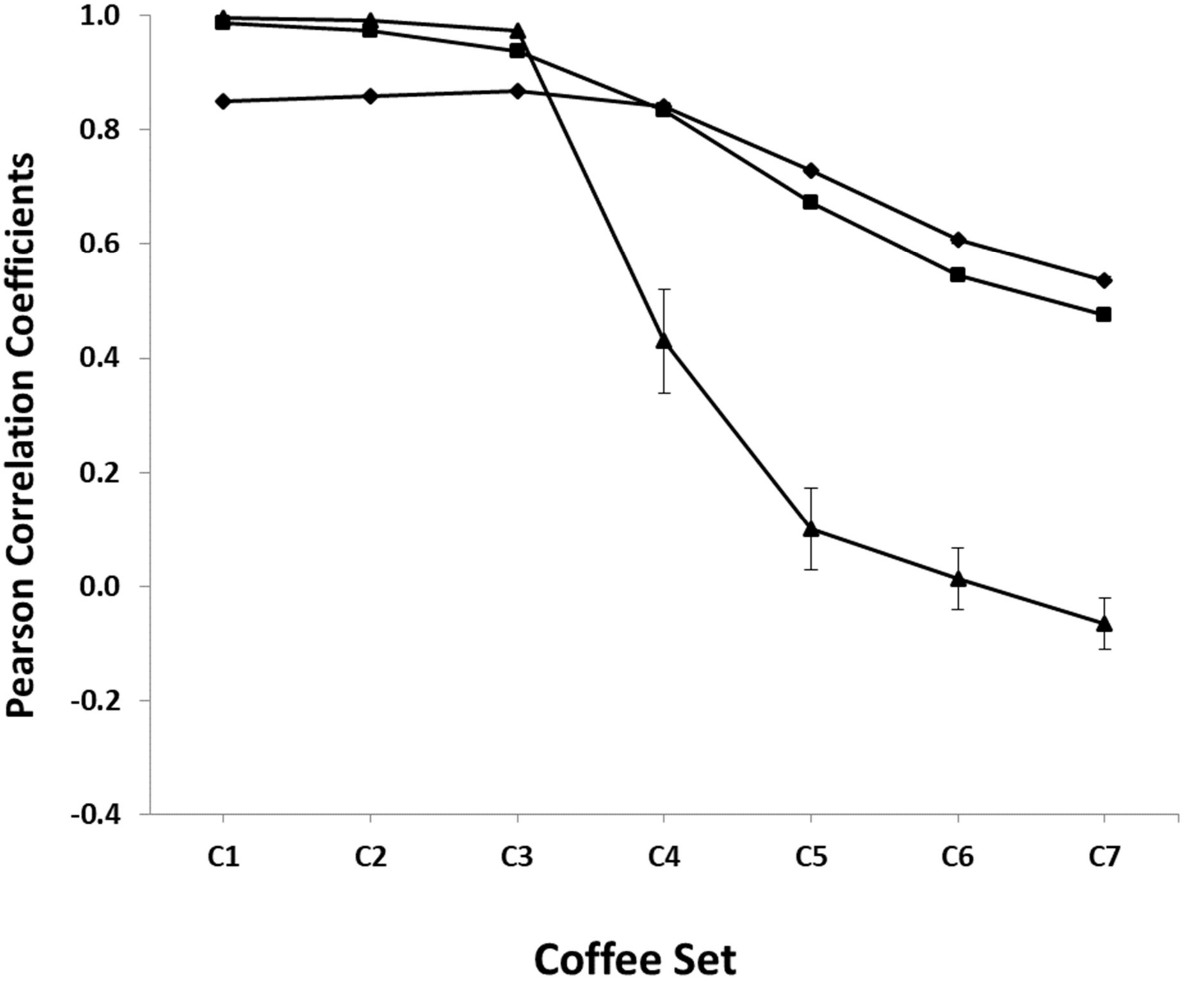


Figure 5. A plot of the average Pearson correlation coefficients (**eqn. 4**) calculated by comparing the true backscaled loadings from the original coffee dataset OPLS-DA model relative to the backscaled loadings from the CS (♦), PQ (■), and ROI (▲) normalized coffees noisy dataset OPLS-DA model. The amount of signal variance introduced into the coffees dataset is described in **Table 2**. The error bars represent ±2*standard error of the mean. Please note that most of the error bars are smaller than the size of the symbols.

3.8 Conclusion

The nine normalization methods available in our MVAPACK software package were evaluated for their ability to compensate for increasing signal variance. The performance of the normalization techniques were tested on simulated and experimental 1D 1 H NMR datasets with the addition of Gaussian noise and dilution factors. However, it is important to keep in mind that the Gaussian noise and dilution factors used in model construction are only an approximation of non-biological variance. At low to moderate noise levels, all of the normalization methods, except HM, performed well in terms of peak recovery. Accordingly, HM should be avoided as a normalization technique for NMR. Notably, peak recovery performance was only dependent on added Gaussian noise, and independent of dilution factor. At high signal variance, most normalization procedures failed to recover true peak intensities except for CS, PQ, and ROI. Again, PQ and ROI normalization algorithms performed equally-well and significantly better than CS at low to moderate noise levels in reproducing the backscaled loadings from an OPLS-DA model. But, ROI generated statistically invalid OPLS-DA models and poor backscaled loadings correlations at higherlevels of noise. Interestingly, CS performed slightly better than PQ in reproducing the backscaled loadings at high noise levels. Thus, our results suggest that CS and PQ perform the best in regards to maintaining the true signal in noisy datasets. Consistent with our prior observations, groups become indistinguishable with increasing noise; and correlations to the true loadings are lost. In other words, an increasing level of additive Gaussian noise masks the true signals in the datasets. Accordingly, if this noise is not handled properly, it will lead to false conclusions and biologically irrelevant observations. In this regards, our analysis suggests that, at a minimum, noise needs to remain below 20% in order for an NMR metabolomics dataset to provide an accurate and biologically-relevant chemometrics model.

3.9 References

1. Butcher, E. C., Berg, E. L. & Kunkel, E. J. Systems biology in drug discovery. *Nature Biotechnology* (2004). doi:10.1038/nbt1017

2. Chen, R. *et al.* Personal omics profiling reveals dynamic molecular and medical phenotypes. *Cell* (2012). doi:10.1016/j.cell.2012.02.009

3. Wishart, D. S. Metabolomics: applications to food science and nutrition research. *Trends in Food Science and Technology* (2008). doi:10.1016/j.tifs.2008.03.003

4. Aardema, M. J. & MacGregor, J. T. Toxicology and genetic toxicology in the new era of 'toxicogenomics': Impact of '-omics' technologies. *Mutat. Res. - Fundam. Mol. Mech. Mutagen.* (2002). doi:10.1016/S0027-5107(01)00292-5

5. Kohl, S. M. *et al.* State-of-the art data normalization methods improve NMR-based metabolomic analysis. *Metabolomics* **8,** 146–160 (2012).

6. Dieterle, F., Ross, A., Schlotterbeck, G. & Senn, H. Probabilistic quotient normalization as robust method to account for dilution of complex biological mixtures. Application in1H NMR metabonomics. *Anal. Chem.* (2006). doi:10.1021/ac051632c

7. Chawade, A., Alexandersson, E. & Levander, F. Normalyzer: A tool for rapid evaluation of normalization methods for omics data sets. *J. Proteome Res.* (2014). doi:10.1021/pr401264n

8. Hochrein, J. *et al.* Data Normalization of ^{1}H NMR Metabolite Fingerprinting Data Sets in the Presence of Unbalanced Metabolite Regulation. *J. Proteome Res.* (2015). doi:10.1021/acs.jproteome.5b00192

9. Giraudeau, P., Tea, I., Remaud, G. S. & Akoka, S. Reference and normalization methods: Essential tools for the intercomparison of NMR spectra. *Journal of Pharmaceutical and Biomedical Analysis* (2014). doi:10.1016/j.jpba.2013.07.020

10. Zyprych-Walczak, J. *et al.* The Impact of Normalization Methods on RNA-Seq Data Analysis.

Biomed Res. Int. (2015). doi:10.1155/2015/621690

11. Choe, S. E., Boutros, M., Michelson, A. M., Church, G. M. & Halfon, M. S. Preferred analysis methods for Affymetrix GeneChips revealed by a wholly defined control dataset. *Genome Biol.* (2005). doi:10.1186/gb-2005-6-2-r16

12. Callister, S. J. *et al.* Normalization approaches for removing systematic biases associated with mass spectrometry and label-free proteomics. *J. Proteome Res.* (2006). doi:10.1021/pr050300l

13. Cuykx, M., Claes, L., Rodrigues, R. M., Vanhaecke, T. & Covaci, A. Metabolomics profiling of steatosis progression in HepaRG® cells using sodium valproate. *Toxicol. Lett.* (2018). doi:10.1016/j.toxlet.2017.12.015

14. Thulin, E., Thulin, M. & Andersson, D. I. Reversion of High-level Mecillinam Resistance to Susceptibility in Escherichia coli During Growth in Urine. *EBioMedicine* (2017). doi:10.1016/j.ebiom.2017.08.021

15. Doran, M. L. *et al.* Metabolomic analysis of oxidative stress: Superoxide dismutase mutation and paraquat induced stress in Drosophila melanogaster. *Free Radic. Biol. Med.* (2017). doi:10.1016/j.freeradbiomed.2017.10.011

16. Fukushima, A. *et al.* Effects of Combined Low Glutathione with Mild Oxidative and Low Phosphorus Stress on the Metabolism of Arabidopsis thaliana. *Front. Plant Sci.* (2017). doi:10.3389/fpls.2017.01464

17. Jung, Y. S., Lee, J., Seo, J. & Hwang, G. S. Metabolite profiling study on the toxicological effects of polybrominated diphenyl ether in a rat model. *Environ. Toxicol.* (2017). doi:10.1002/tox.22322

18. Workman, C. *et al.* A new non-linear normalization method for reducing variability in DNA microarray experiments. *Genome Biol.* (2002). doi:10.1186/gb-2002-3-9-research0048

19. Craig, A., Cloarec, O., Holmes, E., Nicholson, J. K. & Lindon, J. C. Scaling and normalization

effects in NMR spectroscopic metabonomic data sets. *Anal. Chem.* (2006). doi:10.1021/ac0519312

20. Worley, B. & Powers, R. MVAPACK: A complete data handling package for NMR metabolomics. *ACS Chem. Biol.* (2014). doi:10.1021/cb4008937

21. Torgrip, R. J. O., Åberg, K. M., Alm, E., Schuppe-Koistinen, I. & Lindberg, J. A note on normalization of biofluid 1D 1H-NMR data. *Metabolomics* (2008). doi:10.1007/s11306-007-0102-2

22. Barnes, R. J., Dhanoa, M. S. & Lister, S. J. Standard normal variate transformation and de-trending of near-infrared diffuse reflectance spectra. *Appl. Spectrosc.* (1989). doi:10.1366/0003702894202201

23. Windig, W., Shaver, J. & Bro, R. Loopy MSC: A simple way to improve multiplicative scatter correction. *Appl. Spectrosc.* (2008). doi:10.1366/000370208786049097

24. Fujioka, H., Kano, H., Egerstedt, M. & Martin, C. F. *SMOOTHING SPLINE CURVES AND SURFACES FOR SAMPLED DATA*. *International Journal of Innovative Computing, Information and Control ICIC International c* (2005).

25. Worley, B. & Powers, R. Simultaneous phase and scatter correction for NMR datasets. *Chemom. Intell. Lab. Syst.* (2014). doi:10.1016/j.chemolab.2013.11.005

26. Worley, B. & Powers, R. PCA as a Practical Indicator of OPLS-DA Model Reliability. *Curr. Metabolomics* (2016). doi:10.2174/2213235X04666160613122429

27. Worley, B. & Powers, R. Multivariate Analysis in Metabolomics. *Curr. Metabolomics* **1,** 92–107 (2013).

28. Team, R. D. C. & R Development Core Team, R. R: A Language and Environment for Statistical Computing. *R Found. Stat. Comput.* (2016). doi:10.1007/978-3-540-74686-7

29. Weisstein, E. W. Cauchy Distribution. *MathWorld* (2017).

30. Halouska, S. & Powers, R. Negative impact of noise on the principal component analysis of NMR data. *J. Magn. Reson.* (2006). doi:10.1016/j.jmr.2005.08.016

31. Halouska, S. *et al.* Revisiting Protocols for the NMR Analysis of Bacterial Metabolomes. *J. Integr. OMICS* **3,** 120–137 (2013).

Chapter 4: The Challenges of Mass Spectrometry in Metabolomics

4.1 Introduction

Metabolomics is an increasingly important avenue of biological research. The ideal goal of metabolomics is to be able to obtain both accurate and precise measurements on as many metabolites (broadly defined as the small molecule components of cellular metabolism) as possible within a biological system[1–3]

Theoretically, any analytical method capable of measuring these cellular components can be used; fourier transform infrared spectroscopy (FT-IR), nuclear magnetic resonance (NMR), capillary electrophoresis (CE) and mass spectrometry (MS) have all been successfully employed in metabolomics investigations, particularly in combination with separation techniques such as gas and liquid chromatography (GC/LC respectively).However, the vast majority of studies are performed using either NMR or MS[1]. This is in large part due to their amenability to a wide range of metabolites (unlike CE, which performs best on charged species) and one-to-one correspondence between a known metabolite and an observed spectral signal (in contrast with FT-IR methods, which can often only assign a signal to a specific chemical moiety within a larger structure, rather than identifying individual metabolites on its own). Additionally, NMR and MS have spatially resolved signals (especially so when using 2-Dimensional NMR or LC-MS), making them much more amenable to the analysis of complex mixtures than more non-specific methods such as 1D-NMR or Infrared/UV which lack the resolving power to distinguish individual components as the mixture complexity increases[4].

Both NMR and LC-MS offer their own unique advantages. NMR is highly reproducible, with simple sample preparation and amenability to analytes over a wide range of solvent conditions, as well as definitive structural characterization. Despite these strengths, NMR still struggles with a low sensitivity which is a major roadblock that is slowing its wider adoption as an analytical technique for metabolomics. The lower sensitivity of NMR contrasts with MS, which boasts both high sensitivity and low detection limits that can detect more subtle metabolite changes that are “NMR-invisible” due to limited concentrations. However, the nature of MS detectors (and a reliance on ionization processes) necessarily restricts analysis to metabolites which can be readily ionized. Additionally, ~~it is known that~~ minor contaminants are known to reduce ionization efficiency, which negatively impacts the reproducibility of heterogeneous metabolomics samples. Accordingly, NMR and mass spectrometry are often viewed as complementary techniques that broaden the coverage and improve the accuracy of metabolomics studies[5].

. Given the inherent biases of NMR towards high abundance metabolites and MS inability to capture poorly ionizing compounds, each method observes a unique subset of the total metabolites present in a sample, with little overlap between the two sets. A growing recognition of this reality suggests that simply performing both methods simultaneously when it is appropriate and feasible to do is the best approach to cover the largest possible set of metabolites present. The complementarity of NMR and mass spectrometry has resulted in an expanding number of studies that are routinely using both NMR and MS to characterize metabolomics samples in this way such that each method can compensate somewhat for the weaknesses of the other. In order to understand this further, an overview of how each method is commonly used in metabolomics is presented to illustrate the key differences between NMR and MS and what considerations need to be made when choosing the most appropriate platform for a given study.

NMR, being one of the most fundamental spectroscopic techniques, is able to simultaneously identify and quantify a large number of metabolites in the micromolar range. As a measurement method, it is incredibly straightforward and easily automated making it the ideal platform for high-throughput. An additional benefit is that it is non-destructive, and samples can be recovered and analyzed further with other methods. It has been used extensively for biomarker discovery [6,7], as well as determining genetic function[8].The major hurdle of NMR is it’s low sensitivity, which places a practical restriction on the number of metabolites that can be observed. [9]NMR spectra require processes such as zero-filling, fourier transform, phasing, referencing, alignment, normalization, scaling and binning in order to be analyzed by common statistical processes such as PCA and OPLS in metabolomics. As a result of all of these necessary tools, a number of software packages that offer these functionalities have been developed. However, each of these tools was designed to address a subset of these tasks extremely well, with little attention given to the rest of the data processing pipeline.

For example, NMRPipe[10] mainly focuses on tools for the processing of raw, spectral acquisition free induction decays into interpretable spectra while packages such as WavPeak focus mainly on intelligent peaklist generation from previously processed spectra[11].As a result of this, within the NMR community it has become common to utilize multiple software packages, with each step of spectral analysis being performed by the tool best suited to the task. This introduces many potential point of failure, as not all software developed independently of each other can necessarily be guaranteed to cooperate with one another and offer redundant capabilities.

To address this need, MVAPACK was developed. MVAPACK[12] offers a complete set of tools for NMR spectral analysis all the way from initial preprocessing such as fourier transform to statistical analysis and model validation. As a result, metabolomics analysis in NMR can be performed in a single platform, which supports full transparency, uses only open-source tools in an intuitive and interactive scripting environment. MVPACK scripts are also easily documented for future use, allowing analyses to be repeated at any time or extended to include additional steps.

MS in metabolomics has been actively developed over the previous decades, and has been an incredibly active area of research. Unlike NMR, its high sensitivity allows it to potentially measure hundreds of metabolites, including lower concentration compounds that are considered NMR-invisible at physiologically relevant levels[13]. MS can be used either as a standalone instrument (direct injection) or coupled to chromatrography. While direct injection can provide a rapid platform suitable for high-throughput, it suffers from issues of reproducibility that arise from differences in ionization efficiency. Additionally, chromatography is often used in order to reduce the complexity of a sample, conceptually breaking down the problem into a series of simpler mixtures, each measured back to back as they elute from a column. This aids in both reducing spectral overlap due to having a large number of components, as well as providing a secondary measurement based on separation time that can be used to assign particularly ambiguous metabolite signals.The processing needs of MS data are quite unique from that of NMR, requiring the additional tools to address the non-uniformity of collected data such as gap-filling, and the need to eliminate redundant information with deisotoping. Another major point of contrast to NMR is the variety of mass selectors and ionization approaches available, which complicates the ability to perform data analysis with a single software platform. For this reason, MS data is often analyzed using vendor provided software that has been designed to work with the instrument they are bundled with. A reliance on this proprietary software makes the task of standardization much more difficult, and is especially problematic due to it's lack of transparency and inability to be upgraded to include additional tools that might be required by an analyst. In much the same way as NMR, MS also developed a wide range of open-source software tools, each of which was best designed to suit particular steps of the data processing pipeline. There are two main approaches to using both NMR and MS in a single, integrated metabolomics study. The first involves samples being *independently* acquired, extracted and then separately analyzed by either NMR or mass spectrometry. The final results are then compared to each other to search for general consistencies in metabolic alteration that are both present and observable by each method alone. This method is almost always performed as a collaboration, with two (or more) research groups each focusing on one instrumental method, due to the amount of domain expertise that is required to perform a high-quality analysis using either NMR or MS. The advantage of performing an analysis in this way is that it allows each branch of the study to be performed with the best in-class tools available, and having domain experts available to ensure that all necessary steps of experimental design, data collection and statistical analysis are being performed appropriately. This also allows for a comparison of which metabolites and associated metabolic pathways are seen in each method alone. If the same metabolites are identified by both methods, their relative concentration ranges can be compared and used to provide a higher degree of confidence that the metabolite assignment is correct than either method could do alone. However, this may not always be possible due to the unique subsets of metabolites that are often obtained by each method. The chief disadvantage of this method is the need for multiple investigators to perform the data analysis, often with domain specific processing software. This makes it nearly impossible for a single investigator to perform such integrated studies, and introduces multiple points of failure as a result of the large variety of potentially applicable tools.

A less popular, but perhaps more beneficial approach, would be to analyze the two methods simultaneously, using integrated tools that allow a study to be done in a uniform fashion from sample preparation through to statistical analysis. A number of tools have been developed for combining NMR and MS, most notably SUMMIT[14]. Unfortunately, summit does not as of yet include many of the essential processing tools required to deal with raw MS data, and instead relies on pre-processing and generation of peaklists by other programs before it performs it’s own methods of statistical integration.

Despite the demonstrated advantages of using both NMR and MS in combination, there are far fewer tools for performing these integrated analyses than there are for performing either one alone. As a result, proper multiblock analyses require the development of in-house programs and statistical methods that present a significant hurdle to researchers wanting to perform these types of analyses who need not be demanded to possess knowledge of such disparate fields as statistics, computer programming, linear algebra and machine learning in addition to the experimental peculiarities of their particular domain. One example of such integrated studies is multiblock analysis. Briefly, multiblock analysis takes measured variables from multiple discrete “classes” of measurements and exploits the underlying structure of these variable groupings to determine both overall sample differences as well as the relative contributions of these variable classes. When applied to metabolomics, each variable grouping would consist of a set of measurements arising from a given method (*e.g.,* an NMR group, and LC-MS group, etc.). Multiblock analysis has a major advantage over single methods of analysis, since it is able to make use of data from both instrumental techniques simultaneously. This has been shown in at least one paper by Marshall et al. where the integration of NMR and direct-injection mass spectrometry through multiblock modeling within MVAPACK was able to generate models with greater predictive ability and resolving power than models generated from either method alone. While this demonstrated the ability of MVAPACK to perform data analyses derived from multiple sources, direct-injection MS derived spectra were treated as pseudo 1D-NMR spectra, and no functions specific to the realm of MS data processing were utilized. Herein we present a workflow using MVAPACK, in conjunction with freely available tools and the Octave programming language, capable of working with LC-MS data. This in combination with our already reported tools for processing NMR, results in a single software environment capable of robust, easy to perform analyses independent of instrumental platform, and addresses the need for software that can perform these integrated analyses that the field demands.

4.2 Methods and Software Design

4.2.1 Software design for MVAPACK LC-MS data processing pipeline.

MVAPACK (http://bionmr.unl.edu/mvapack.php) was originally developed in GNU Octave version 3.6.0, but the recent Octave upgrade to version 4.2.0 required a complete revision to MVAPACK to remove deprecated functionality used by the loadnmr(), classes(), and addclasses() functions. MVAPACK is comprised of a set of individual, independent functions that can be readily interchanged or combined to form the desired processing pipeline. In this regards, the output of one function becomes the input for the subsequent function in the data handling process, with entire data workflows being simply represented as a sequence of brief functions called in the order in which they are to be performed. MVAPACK currently contains the following broad classes of functions: (i) data input , (ii) instrument-specific preprocessing, (iii) instrument-agnostic preprocessing, (iv) multivariate modeling and visualization, and (v) model validation.

An LC-MS dataset requires the following processing steps that differ from the NMR processing and are not currently available in MVAPACK: (i) functions to import LC-MS data, (ii) visualization utilities for LC-MS data, (iii) and functions to pre-process LC-MS data.   Accordingly, in order to implement an LC-MS data processing pipeline into MVAPACK the following new functions as outlined in Figure 4.1 were required: mzxmlinfo.m, which loads and stores sample acquisition parameters, sparse_matrix_load.m and a number of internal functions including get_line, get_mz_axis and get_rt_axis, which load an LC-MS sample into a memory efficient structure for storage and manipulation, array_converter.m, which removes unnecessary metadata and prepares for statistical analysis, threshold.m which allows for the generation of peaklists, plotscan.m and plotscan3d.m, which allow for visualization of imported data.

For file import, a number of vendor formats are commonly available. In order to make this process as general as possible, an open-source data format was required. Specifically, a common file-format used to represent mass spectral data is mzXML mzXML, first proposed by the HUPO initiative as a way to represent MS data, has become an accepted, vendor-independent MS storage format. Briefly, mzXML is based on extensible markup language (XML) which is used to represent data in a hierarchical fashion using tags elements and attributes. The large size of LC-MS datasets relative to NMR is a fundamental challenge that requires data reduction. This was addressed with a combination of sparse matrices and thresholding to restrict analysis to non-zero values above a user-defined intensity level. Uninformative spectral regions were then removed, and peaklists generated from binned regions of the thresholded data, and multivariate analysis performed on the subsequent peaklists.

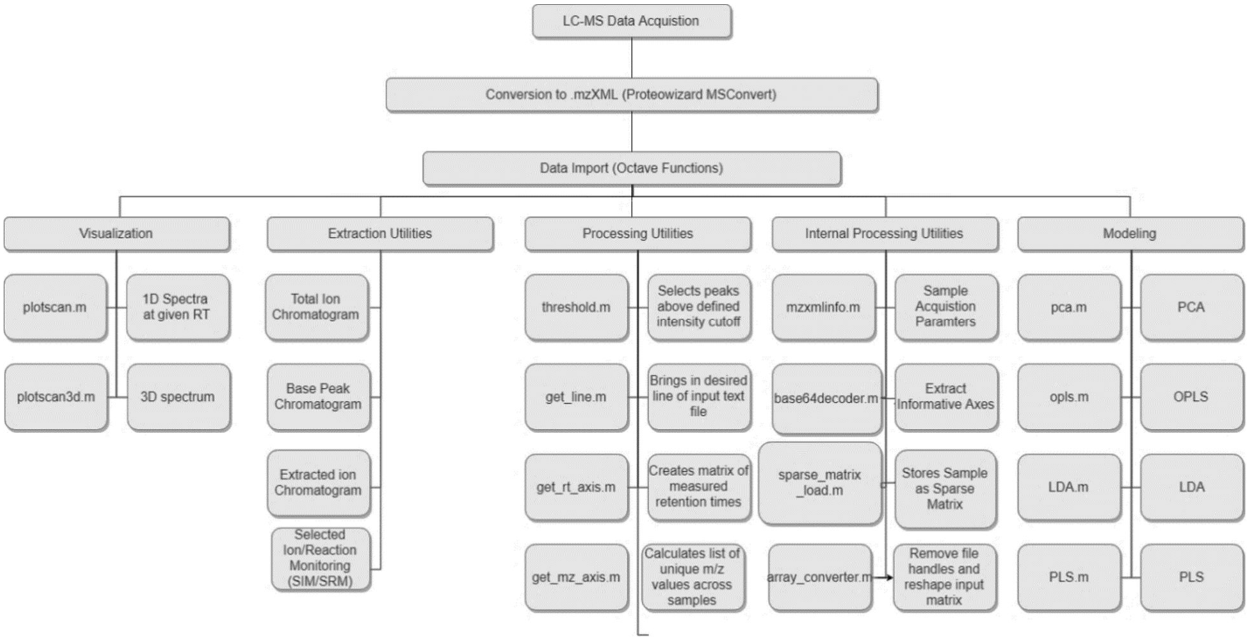

LC-MS Data Acquistion
Conversion to .mzXML (Proteowizard MSConvert)
Data Import (Octave Functions)
Visualization
Extraction Utilities
Processing Utilities
Internal Processing Utilities
Modeling
plotscan.m
1D Spectra at given RT
plotscan3d.m
3D spectrum
Total Ion Chromatogram
Base Peak Chromatogram
Extracted ion Chromatogram
Selected Ion/Reaction Monitoring (SIM/SRM)
threshold.m
Selects peaks above defined intensity cutoff
get_line.m
Brings in desired line of input text file
get_rt_axis.m
Creates matrix of measured retention times
get_mz_axis.m
Calculates list of unique m/z values across samples
mzxmlinfo.m
Sample Acquistion Paramters
base64decoder.m
Extract Informative Axes
sparse_matrix _load.m
Stores Sample as Sparse Matrix
array_converter.m
Remove file handles and reshape input matrix
pca.m
PCA
opls.m
OPLS
LDA.m
LDA
PLS.m
PLS

Figure 4.1 Outline Of Functions For Processing LC-MS Data In MVAPACK. Newly developed functions included plotscan.m, plotscan3d.m, threshold.m, get_line.m, get_rt_axis.m, get_mz_axis.m. mzxmlinfo.m, base64decoder.m, sparse_matrix_load.m, and array_converter.m.

4.2.2 Standard LC-MS dataset.

A representative dataset (7MIX_STD LC-MS) from the Sashimi Project Data Repository (http://sashimi.sourceforge.net/repository.html) was used to evaluate the performance of the LC-MS data handling pipeline implemented into our MVAPACK software (http://bionmr.unl.edu/mvapack.php). The dataset is centroided. A single spectrum in the dataset contains 7161 retention times with 2,194,813 unique *m/z* values. A two-dimensional (2D) array of retention times and *m/z* values would result in a staggering ~15 billion intensity entries. Thus, a single spectrum would require over 120 Gb of RAM if each intensity is stored as a double precision floating point value (a requirement for most interactive languages such as python to ensure values are not artificially truncated). Handling a dataset of this size is computationally intractable for typical compute configurations. Accordingly, the large 7MIX_STD LC-MS dataset presents a fundamental problem with processing LC-MS data that was resolved with our MVAPACK software.

4.2.3 Preparation of experimental metabolomics samples.

The LC-MS data handling pipeline implemented in MVAPACK was further evaluated using a typical LC-MS dataset comprised of cell lysates. 10 Replicates each of two unique cellular species (*E. coli* MG1655 *and* Wild Type *S. aureus)* were grown in a preferred media of Luria-Bertania (LB) and Tryptic Soy Broth (TSB), respectively. 250 mL of sterile media was inoculated to an initial OD_{600} of 0.05 and allowed to incubate at 250 rpm at 37 ºC. A portion of the cells were harvested at an OD_{600} of 0.8 and separated from the growth medium by centrifugation at 12,000 rpm for 60 seconds and then quenched in ice-cold methanol. Cell pellets were then stored at -80 ºC until further processing. Cell pellets were extracted with 2 ml aliquots of HPLC-grade methanol while vortexing three times for 30 seconds and resting in dry ice between rounds. Samples were then centrifuged at 12,000 rpm to pellet the cell debris and 1 mL of supernatant was removed and dried using a Savant Speed-vac system with a labconco freeze dryer. Dried extracts were reconstituted in 1 mL of 0.1% formic acid and centrifuged at 10000 rpm for 5 minutes to remove any insoluble material.

4.2.4 Acquisition of LC-MS spectra.

The LC-MS analyses were performed using an ACQUITY Ultra-Performance Liquid Chromatography (UPLC) system (Waters, Milford, MA, USA) coupled to a Waters Xevo G2-XS Q-TOF mass spectrometer (Waters Co., Milford, MA, USA.) with an electrospray ionization (ESI) source. The column used was an ACQUITY UPLC HSS T3 C18 (1.0 × 50 mm, 1.8 µm, Waters Co., Milford, MA, USA). Column and autosampler temperature were set to 40 °C and 5 °C respectively, and the flow rate was set at 95 µL/min. The mobile phase was composed of 0.1% formic acid in water (A1) and 0.1% formic acid in acetonitrile (B1). Two microliters of sample were injected and separated with a linear gradient program from 1% to 95% B in 7.30 min. The ESI source was set to positive mode with a scan range of *m/z* 50 to 1,200. The voltage of the capillary and cone were set to 3.2 kV and 40 V, respectively. The gas flow for desolvation and cone was 800 and 50 L/h. The source temperature and desolvation gas temperature were 120 and 400°C, respectively.

4.2.5 Processing of LC-MS spectra.

Waters ".Raw" vendor-formatted data files were converted to an uncompressed text format and filtered with a continuous wavelet transform using Proteowizard's MSConvert tool. Data was imported with the Data analyses were performed in Octave 4.4.0, using a series of functions developed for this purpose. All analyses were performed on an Intel Pentium 2.6Ghz with 8Gb of RAM, installed with Windows 10 and Opensuse "Tumbleweed" rolling release.

4.3 Results and Discussion

4.3.1 MVAPACK upgrade to be GNU Octave complaint

The underlying structure of MVAPACK is as a set of functions that run within the Octave programming language. This occasionally requires updating functions to remove any references to deprecated functionality whenever a language undergoes a major update. In this latest update, Octave changed the behavior of the inbuilt fileread() command, used to import plaintext files into Octave.

Previously, the fileread command returned a column vector containing each individual character of the text file in the order that it appeared whereas in 4.0 onwards, it returns a new structure that maintains spacing, tabs and other formatting information so files can be viewed in the same format they appear in a regular text editor. Each of the functions that utilized this function for reading information from plaintext files was updated to get the required information from this new structure using a series of regular expressions.

4.3.2 LC-MS data Input

A particular challenge of MS instruments is the number of vendor proprietary data formats. It would not be feasible to develop functions that work with each vendor format, especially when considering that these functions could cease working entirely if an update to the format is made by the vendor without prior notice. Using an open source format, such as mzXML, alleviates this issue by creating a universal starting point that all open-source approaches can use. This approach is supported by both vendors and the community, with mzXML export supported by all MS vendors. The highly structured style of mzXML makes it simple to identify the location of desired information, with each section being clearly delineated with start and end tags that represent what is contained between them. This is accomplished using the loadmzxml() command, which makes use of built-in fileread() commands of Octave along with regular expressions that pattern match the desired binary data arrays conforming to the mzXML specification.

4.3.3Data Size and Dimensionality

One of the major hurdles of dealing with MS data (particularly when coupled with a chromatographic method), is the sheer number of data points that are obtained. In order to process this data in an interactive way, the size of the dataset must be compressed or reduced significantly, while minimizing the effects that this compression will have on the final analysis.

A potential approach to data reduction is to take advantage of the inherent property of an LC-MS data, which is data sparsity. Simply, not every pair of *m/z* values and retention times are necessarily observed in a given sample. ~~Not every m/z value will be seen in every retention time, and vice versa.~~ A traditional tabular format of data storage would simply represent these unobserved pairs as zeros. Even though these values are "zero", the data point still occupies as much disk space or memory as non-zero values. In essence, the abundance of zero data points waste precious computational resources. An alternative route would be to denote only the values which are non-zero, implicating that anything not stored would be zero. An example of such a storage scheme, herein referred to as a sparse matrix, is presented in Scheme 1. While this is a modest memory savings of 8 bits in this example, it's important to remember that ***A*** scales exponentially. The increase in memory or disk allocation occurs regardless of whether meaningful data is present. Notably, the component vectors only scale with the number of values that must be stored. This is essential for LC-MS data, since a single column (for example, an observed *m/z* value) might only be present at one point in the chromatogram and is most efficiently stored as a set of positional indices and their associated intensity value. As can be seen in figure 3, the number of non-zero values (denoted as ***nnz***), comprises less than 1% of the total data structure using a typical storage scheme. Thus, a sparse matrix would lead to a >99% reduction in memory or disk utilization. The exact memory savings is expected to vary depending on instrument resolution and acquisition parameters. Nevertheless, a sparse matrix potentially resolves the inherent challenge of manipulating large LC-MS datasets a computer language such as Octave or MVAPACK. Importantly, a sparse matrix approach may allow large LC-MS datasets to be processed and manipulated transparently or to be easily ported to another software environment.

An overview of the functions used for processing LC-MS data is shown in figure 4.1. A diagram explaining the rationale of a sparse matrix is presented in figure 4.2. The 7MIX_STD LC-MS dataset was used to demonstrate the application of sparse data matrix. Figure 4.3 illustrates how the LC-MS data is accessed from a sparse matrix. Each intensity-peak ID pair from the 7MIX_STD LC-MS dataset is mapped to its experimentally observed intensity. The peak ID is used instead of a definite *m/z* value in order to account for discontinuities between groups and across biological replicates. Conversely, a traditional “dense” matrix approach assumes that a non-zero intensity is obtained for each *m/z* value. This occurs even if the intensity is zero across the entire data set. Instead, a peak ID allows for assigning an *m/z* value, or a range of m/z values, to an index. In this regards, each LC-MS peak is correctly represented in the data matrix even it does not occur at the *exact* retention time or *m/z* value for all replicates in the data set.

Scheme 1. Sparse Matrix Formulation and Rationale

$$\begin{vmatrix} 0 & 0 & 5 & 0 \\ 3 & 0 & 0 & 2 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 4 \end{vmatrix} = \mathrm{A}$$

Scheme 1.2 A dense representation of a sparse matrix

The dense version of this matrix requires explicit storage of the zero values. When the relative proportion of zero values is incredibly high, this becomes extremely wasteful. Instead, it can be decomposed into three component vectors containing the indices and non-zero values only.

I=[1 2 2 3 4] (row indices)

J=[3 1 4 2 4] (column indices)

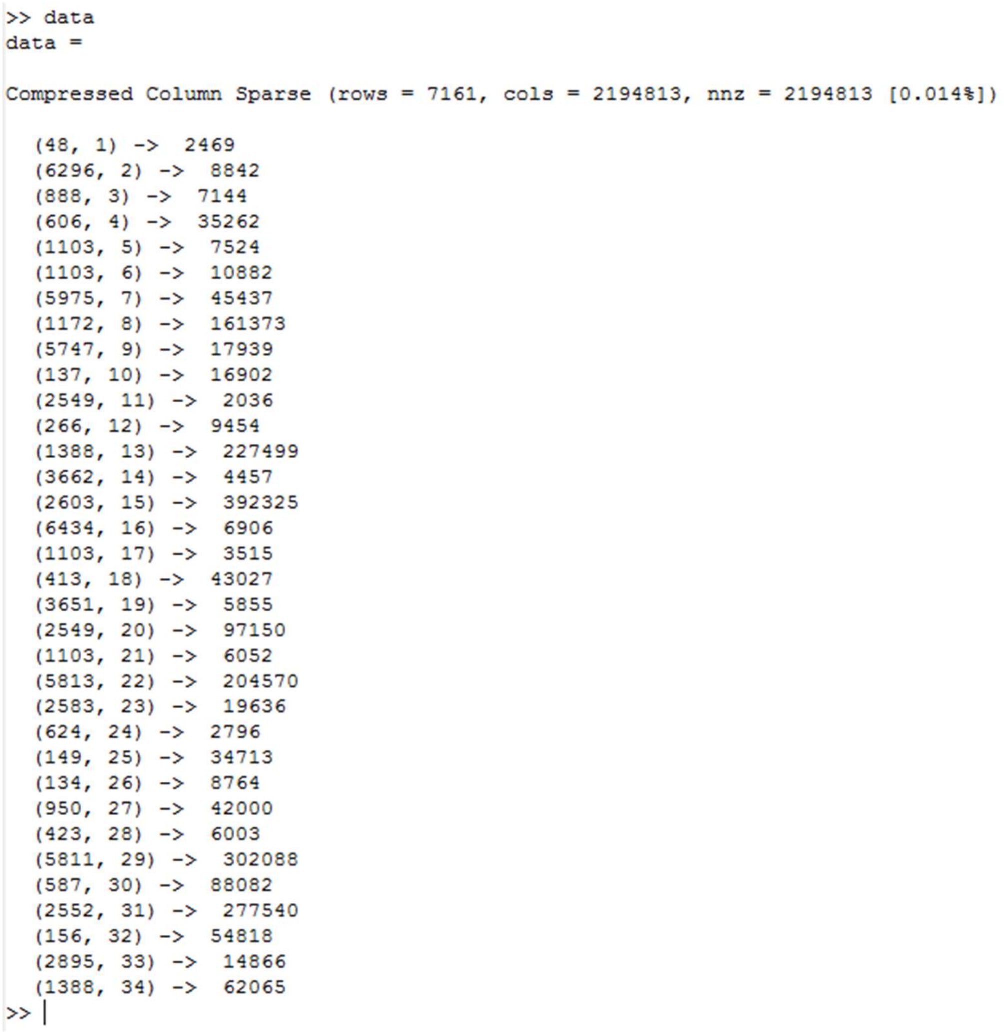
K=[5 3 2 1 4] (non-zero values)

Figure 4.3 Component Vector Decomposition of A

Example memory savings:

A= 4x4 (16 values) *8 = 128 bits

I=5values*8bits=40 bits



```
>> data
data =

Compressed Column Sparse (rows = 7161, cols = 2194813, nnz = 2194813 [0.014%])

  (48, 1) ->  2469
  (6296, 2) ->  8842
  (888, 3) ->  7144
  (606, 4) ->  35262
  (1103, 5) ->  7524
  (1103, 6) ->  10882
  (5975, 7) ->  45437
  (1172, 8) ->  161373
  (5747, 9) ->  17939
  (137, 10) ->  16902
  (2549, 11) ->  2036
  (266, 12) ->  9454
  (1388, 13) ->  227499
  (3662, 14) ->  4457
  (2603, 15) ->  392325
  (6434, 16) ->  6906
  (1103, 17) ->  3515
  (413, 18) ->  43027
  (3651, 19) ->  5855
  (2549, 20) ->  97150
  (1103, 21) ->  6052
  (5813, 22) ->  204570
  (2583, 23) ->  19636
  (624, 24) ->  2796
  (149, 25) ->  34713
  (134, 26) ->  8764
  (950, 27) ->  42000
  (423, 28) ->  6003
  (5811, 29) ->  302088
  (587, 30) ->  88082
  (2552, 31) ->  277540
  (156, 32) ->  54818
  (2895, 33) ->  14866
  (1388, 34) ->  62065
>> |
```

Figure 4.4 Interactive viewing of the 7MIX_STD LC-MS dataset stored as a sparse array in octave. Retention times are stored as rows, *m/z* values stored as columns, and sparsity data is displayed along with the variable. Each line corresponds to a detected peak, read as

$$\text{(retention\_time\_index, mz\_index) -> intensity value.} \qquad [4.1]$$

Retention times and *m/z* values are stored in separate axes vectors such that are indexed using the indices stored in the sparse array to retrieve the associated m/z and retention times for a given peak.

4.4 Visualization of LC-MS Data

Figures 4 through 7 demonstrate commonly requested visualizations of LC-MS data. These MVAPACK functions are useful for the visual confirmation of sample and spectral quality that should be performed prior to attempting multivariate modeling and statistical analysis. Figure 4 shows a top-down visualization (or more colloquially, a “dot plot”) of a standard sample from the 7MIX_STD LC-MS data set retrieved from the Sashimi Project Data Repository. The 7MIX_STD LC-MS data set consists of a 7 protein mix, including rabbit glycogen phosphorylase, e. coli beta galactosidase, bovine serum albumin, myosin, chicken ovalbumin and bovine serotransferrin The data set was chosen to illustrate the MVAPACK LC-MS data visualization functions due to the small number of components of high spectral itensity, as well as being an openly-accessible data set that is commonly used for similar benchmarking in other LC-MS programs. A threshold of $5x10^9$ was applied to the LC-MS dataset so only peaks with intensities above this threshold value are observable. This type of visualization is particularly important for identifying regions of the spectrum that either contains peaks or are devoid of data. For example, the 7MIX_STD LC-MS data set lacks high intensity peaks between scans 4000 and 5000, as seen in figure 4.5.

Figure 4.6 shows an expanded view of the chromatographic time points between 1 and 300. A single high intensity peak occurs at multiple time points, which are being mapped to different peak ID values. This is due to the peaks occurring at slightly different *m/z* values. Thus, the visual display of the LC-MS data set allowed for the identification of the mass deviation, which can then be easily corrected through an appropriate choice of binning or other suitable approaches. Furthermore, displaying the LC-MS data in this manner allows for a reasonable estimate of the mass deviations across the entire datasets. The new MVAPACK LC-MS visualization functions also allow for displaying wither a single mass spectrum or chromatogram. A single mass spectrum from the first time point of a chromatographic run is shown in figure 4.7. Similarly, the separation efficiency for a given biological sample from the data set may be assessed by viewing the total ion chromatogram as shown in figure 4.8. Importantly, the total ion chromatogram and may be statistically analyzed like the mass spectral data using the same set of tools implemented into MVAPACK.

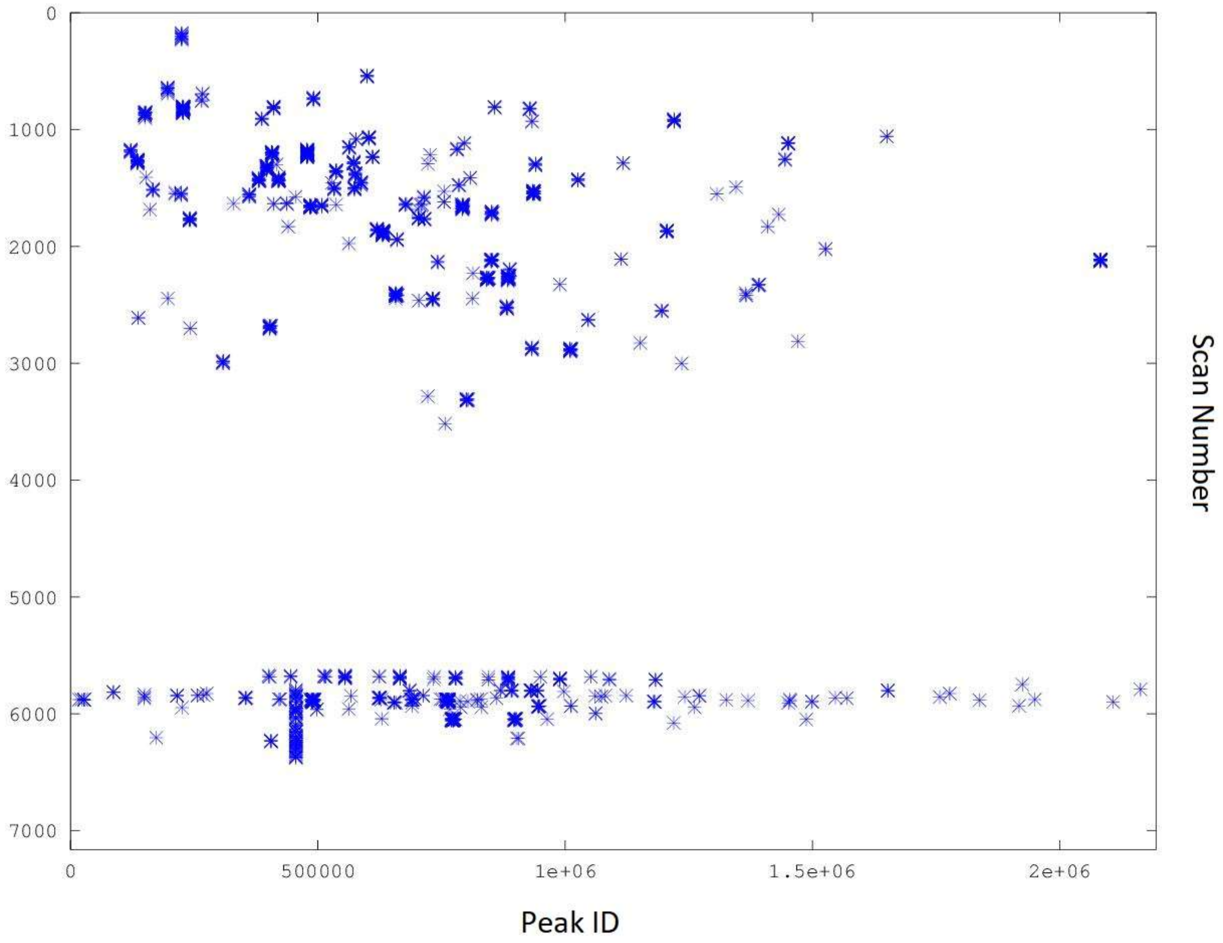


Figure 4.5 2D Visualization of 7MIX-STD LC-MS data set. Peaks below the given threshold ($5x10^9$) are excluded.

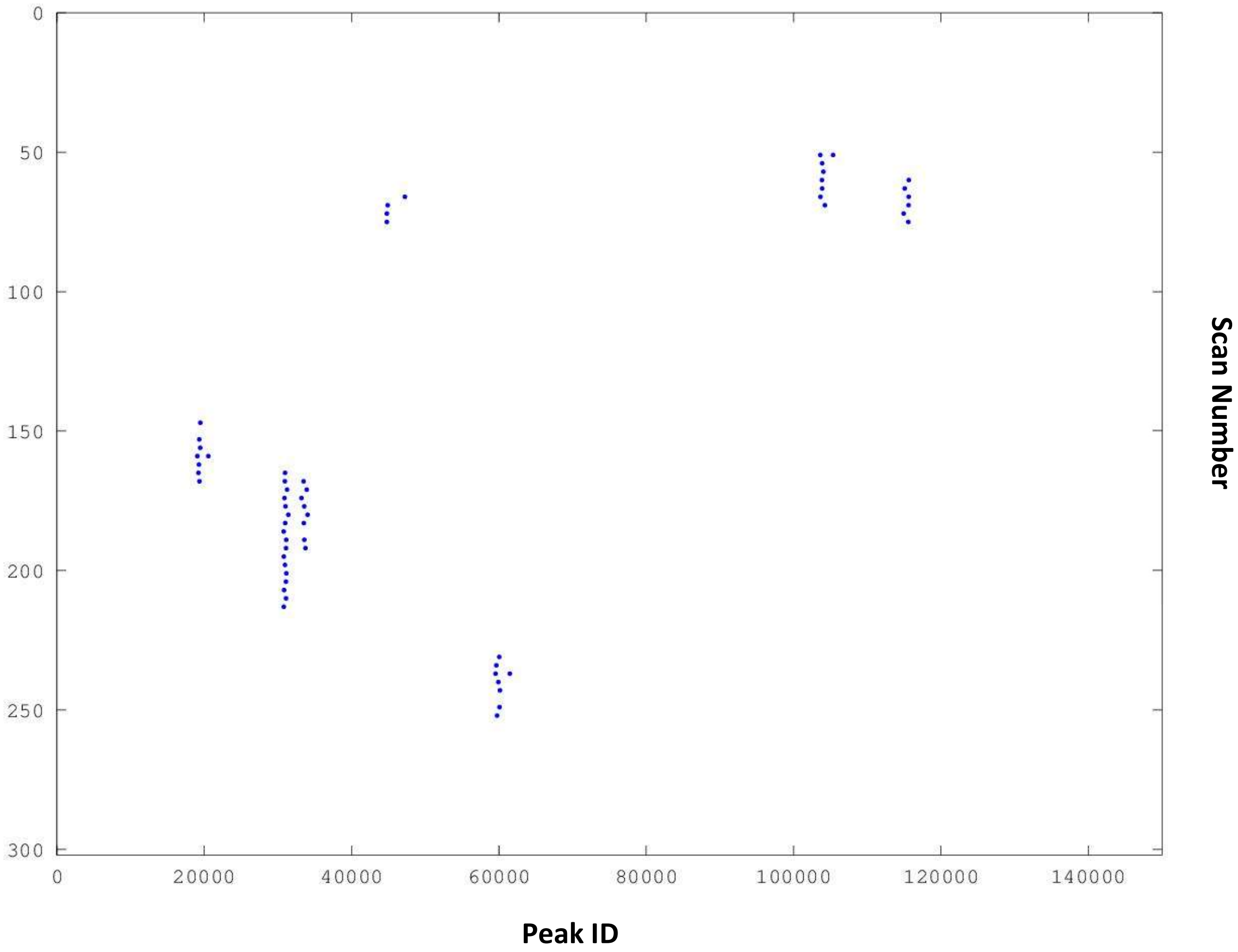


Figure 4.6. Selected region of figure 4 from the first 300 obtained chromatrographic time points. Distinct peaks can be identified and scan variability can be visually estimated in the “peak ID” axis, which consists of m/z values with any observed variance out to 0.000001 m/z. Visual estimates can be used to determine whether a given peak selection and binning routine is applicable to the imported dataset.

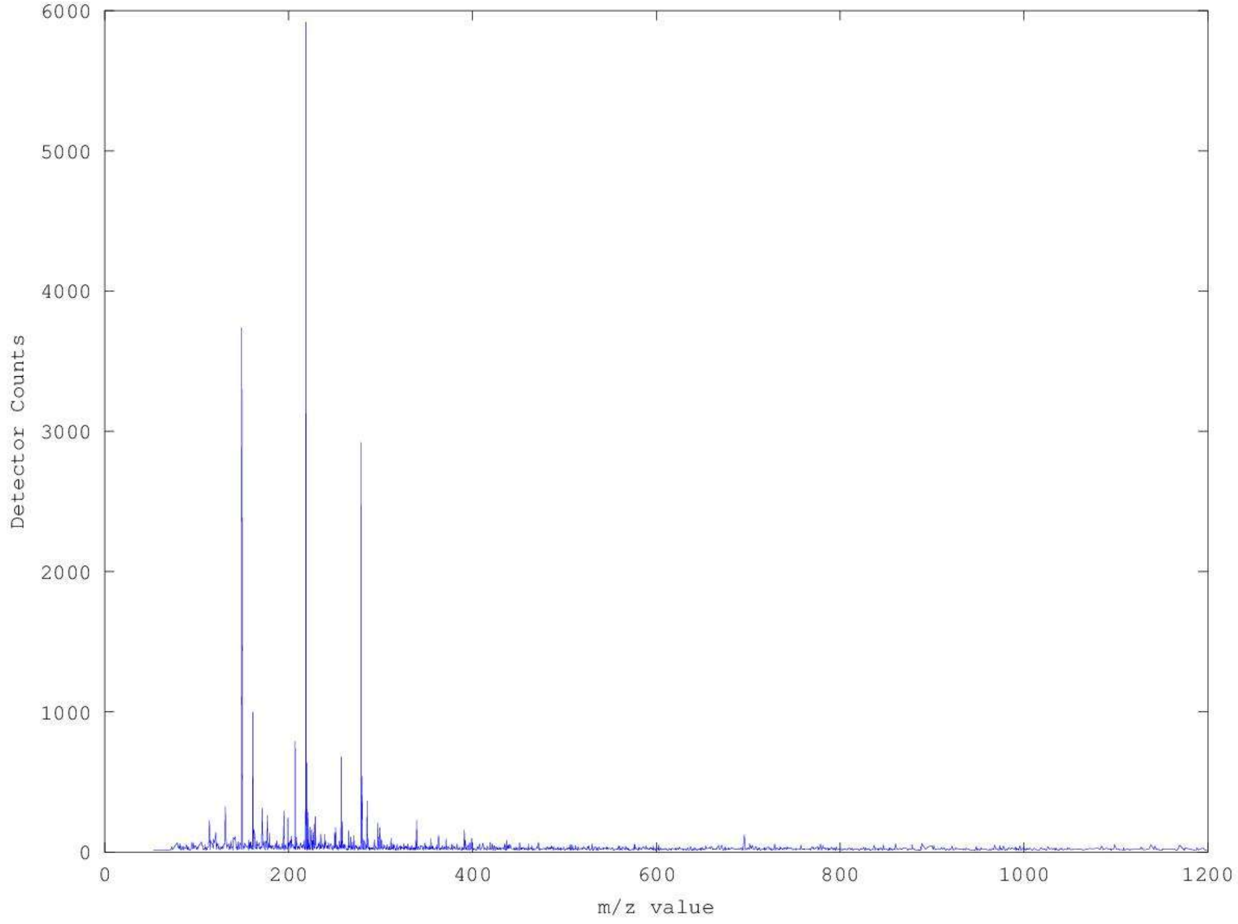


Figure 4.7. Visualization of a single mass spectrum from the LC-MS dataset obtained from *E. coli* cell lysates using the plostcan() function. This can be used to visually estimate an acceptable thresholding intensity and estimate noise levels; this spectrum for example, should have a threshold set no higher than 5000 and no less than 100-200 in order to include sufficient peaks while removing uninformative noise regions.

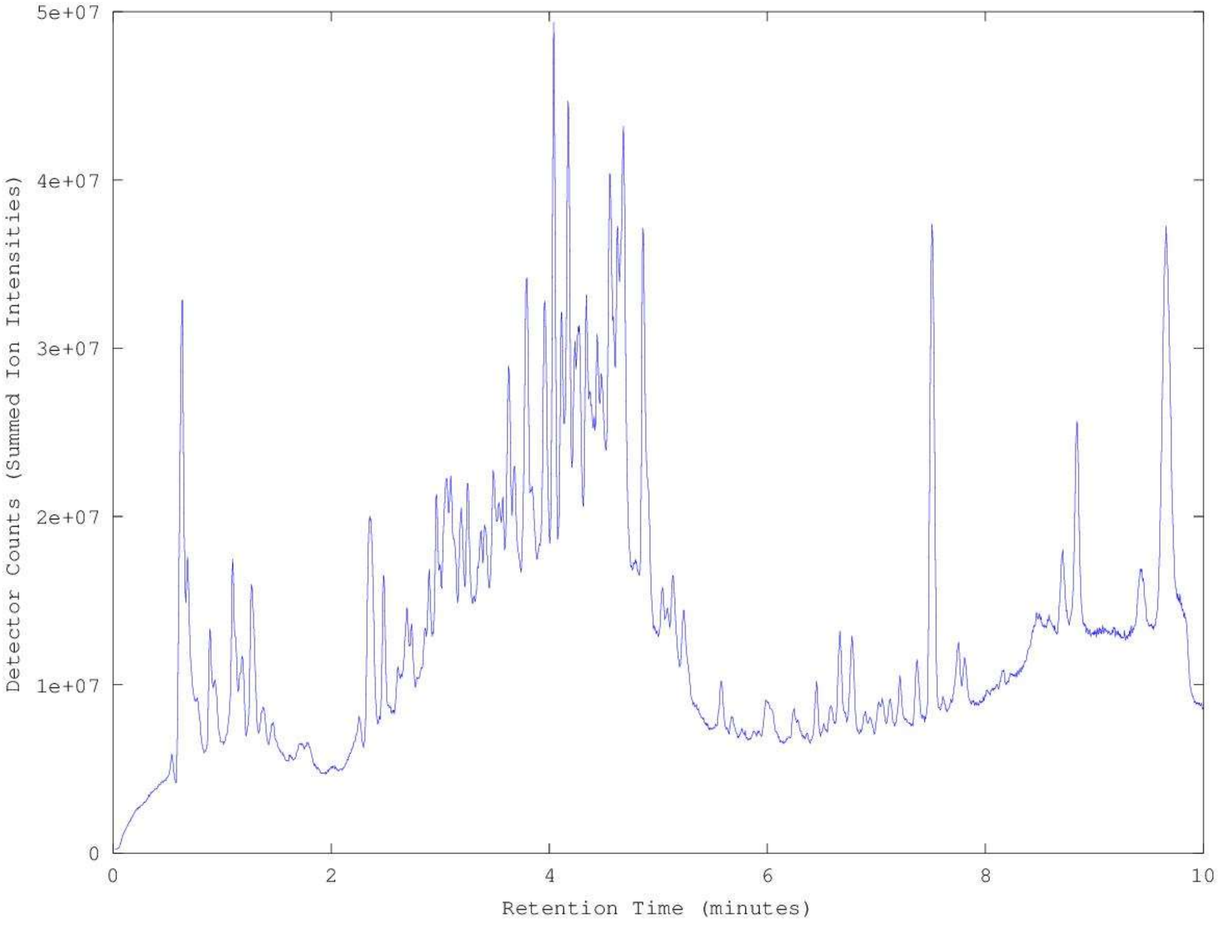


Figure 4.8 Total ion chromatogram (*m/z* projection) from the LC-MS dataset obtained from *E. coli* cell lysates. Generated by summation of m/z intensities within a given scan/chromatrographic time point aand plotted with the plotscan() function. Can be used to inspect the separation efficiency of the LC portion of an LC-MS sample acquisition, and determine regions of potential statistical relevance.

4.5 A Multivariate analyses of LC-MS data

Besides visualization utilities, these samples can also be processed to generate multivariate statistical models. Samples LC-MS spectra from the *E. coli* and *S. aureus* cell lysates were used as a test dataset to demonstrate the LC-MS data processing pipeline implemented into MVAPACK., The LC-MS spectra were collected in continuous mode, and processed using the proteowizard MSConvert[15] tool to convert them into a plantext, mzXML format. The *E. coli*-*S. aureus* LC-MS data set was then imported into MVAPACK using the new loadmzxml() function, and stored as cell arrays of m/z and intensity pairs. These cell arrays were then transformed into sparse arrays using the array_converter() function, and subsequently visualized with the plotscan and plotscan3d commands. An appropriate threshold was set based on the visually determined noise level from these plots, and peaklists generated by this thresholding were stored in a data matrix used for further statistical modeling. A principal component analysis (PCA) was generated from the total ion chromatograms for the *E. coli*-*S. aureus* LC-MS data set. The entire set of replicates and full resolution data was utilized for the PCA model. The total Ion Chromatogram was not subjected to any binning or other processing other than an SNV scaling prior to this PCA analysis. Using the total ion chromatograms allows for a global visualization of the dataset, with a focus on the separation efficiency and the identification of potentially discriminative regions in the chromatogram that can guide subsequent analysis of mass spectra occurring in these regions.

As expected, *E. coli* and *S. aureus* form separate and distinct clusters in the PCA scores plot. Thus, the total ion chromatograms alone are sufficient to distinguish *E. coli* from *S. aureus*. The PCA component weights were projected back onto the original chromatograms to generate a back-scaled loadings plot to identify the key chromatographic features that define the group separations (Figure 4.10). The back-scaled loadings show that the strongest contributing factors to group separation are located in the retention time region corresponding to 7.6 to 7.8 minutes. This chromatographic region alone is sufficient to distinguish different between the two bacterial strains. Furthermore, the group dependent chromatographic region may be used as a tool for variable selection of the mass spectral data.

Using the previous analysis indicating the importance of the retention time range of 7.6 to 7.8 minutes, the analysis of the mass spectral data was restricted to the *m/z* values occurring at a retention time of 7.7 minutes. A PCA model was generated from a subset of the *E. coli*-*S. aureus* LC-MS data set corresponding to a retention time of 7.7 minutes. The PCA scores plot and the back-scaled loadings are shown in figures 4.11 and 4.12. Once again, *E. coli* and *S. aureus* form separate and distinct clusters in the PCA scores plot, but the relative separation is reduced compared to the previous example. This is to be expected since less data was available to construct the PCA model. In effect, a single mass spectrum was used for each replicate, which contains significantly less data points than the corresponding chromatogram. However, the back-scaled loadings based on the mass spectral data are more informative then the results with the total ion chromatogram. Specifically, the pseudo-mass spectrum allows for the ready identification of key metabolites that differentiate the two groups. For example, the large, negative weighting at *m/z* 227.868 indicates that this metabolite significantly contributes to the observed difference between the two bacterial strains. Additionally, the observed mass can be combined with the retention time of 7.7 minutes to assist in assigning a metabolite to the spectral feature.

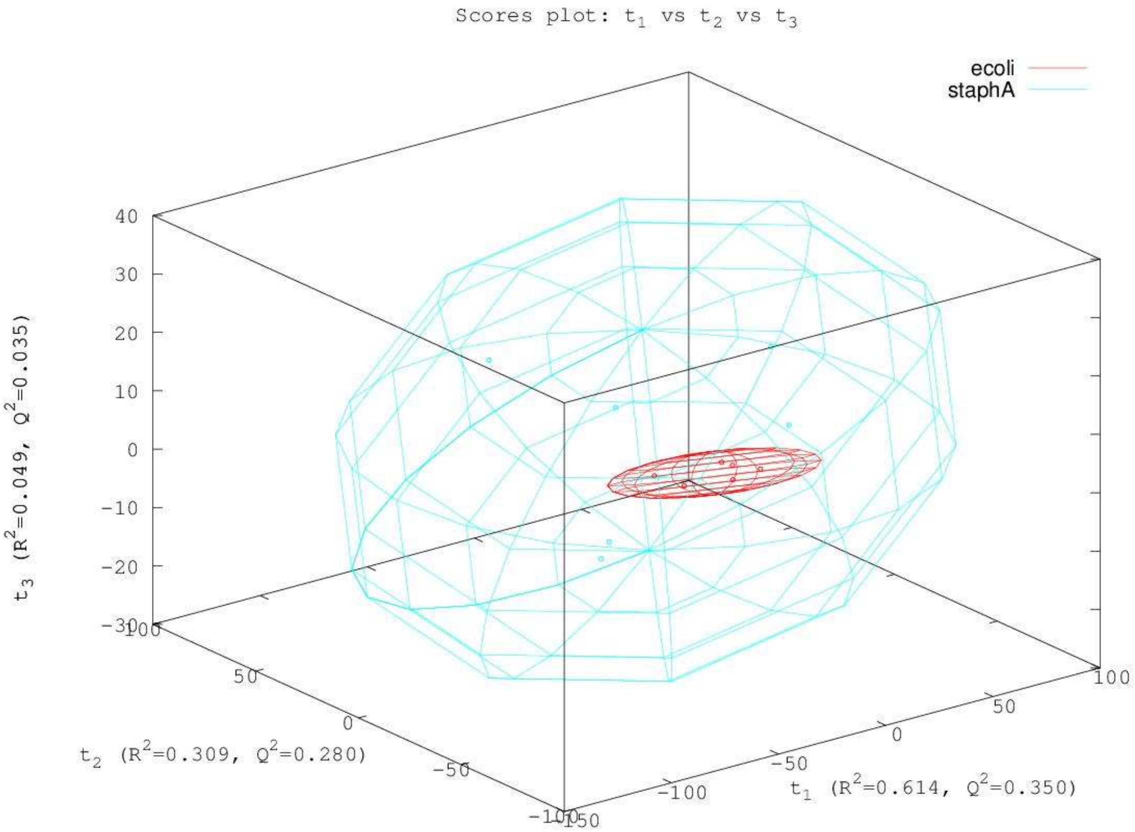


Figure 4.9. PCA scores plot generated from the total ion chromatograms from the LC-MS datasets obtained from *E. coli and S. aureus* cell lysates. A 4 component PCA model was obtained with a cumulative R^2 of 0.988.

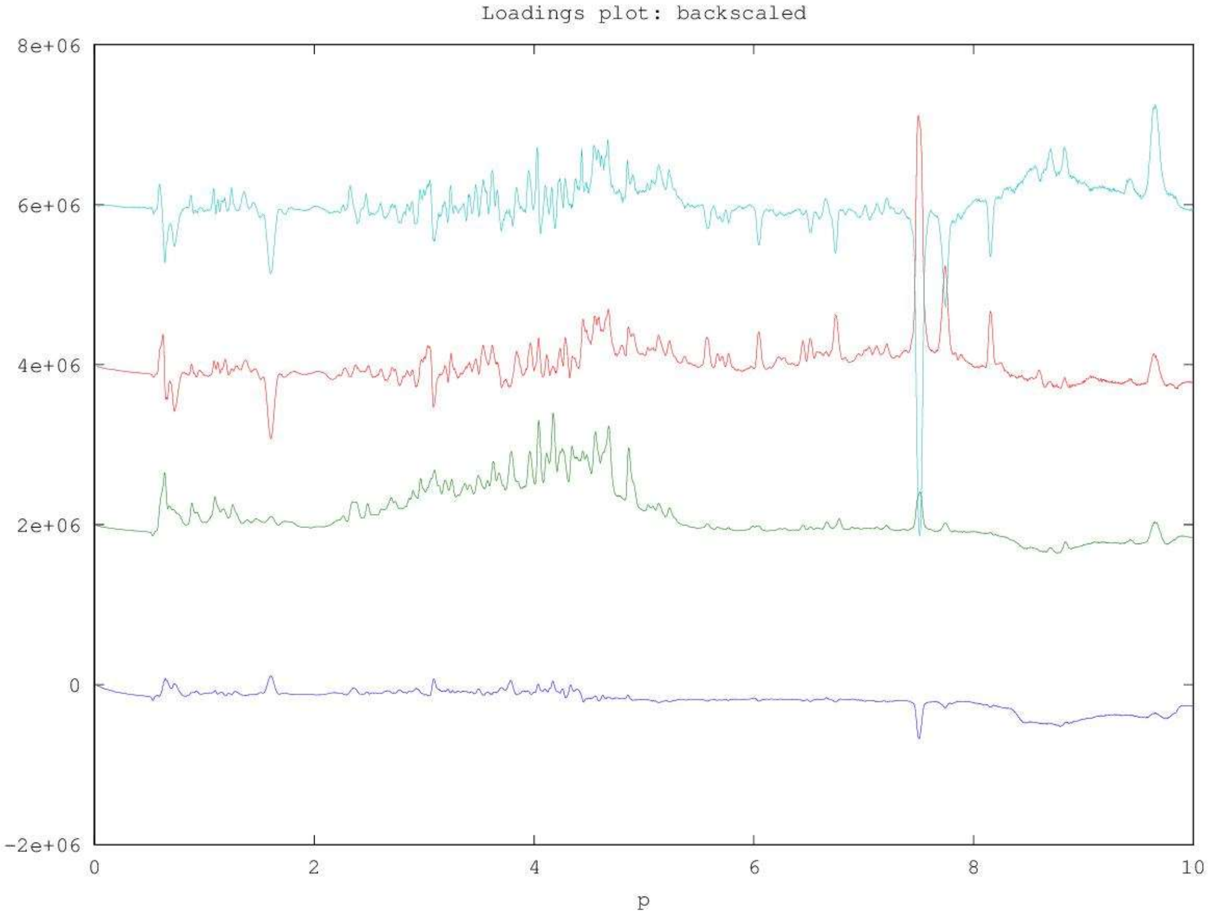


Figure 4.10. A backscaled loadings plot based on the weights obtained from the model in figure 9. A strong downregulation in the region around 7.6-7.8 is clearly evident in the first two components of the model, suggesting the presence of underlying spectral differences in this range.

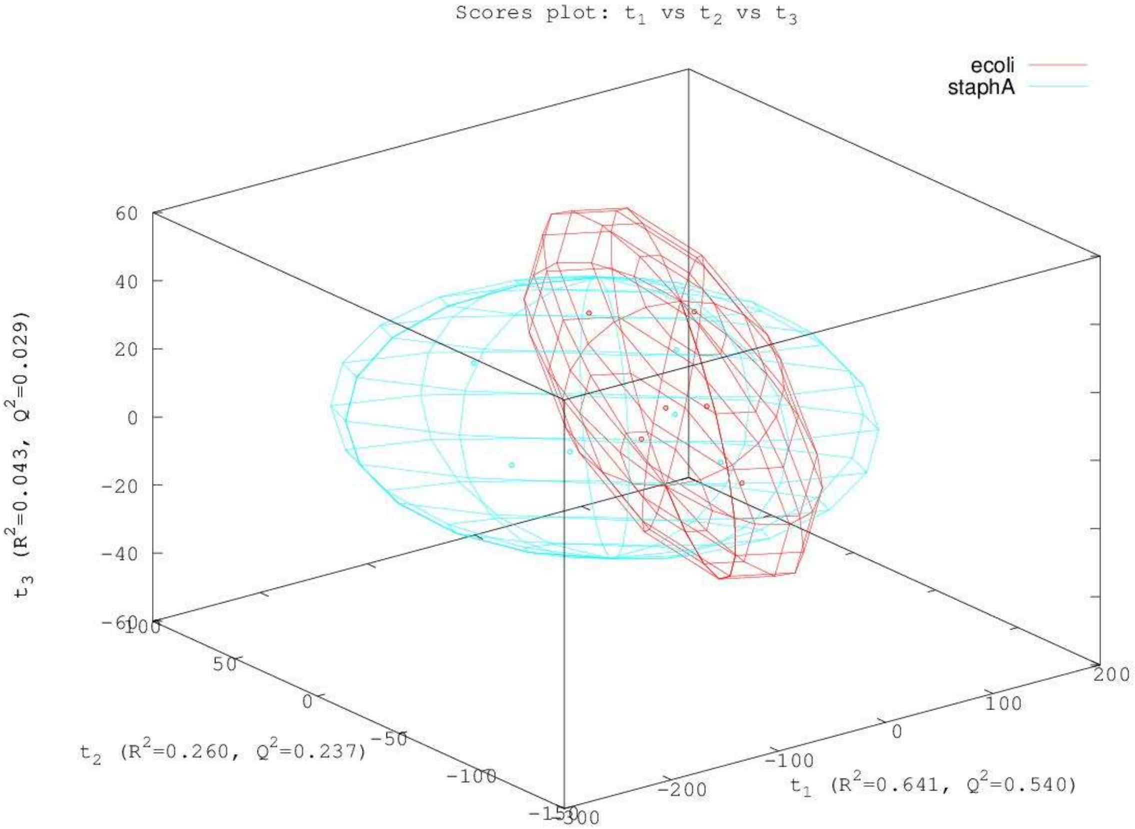


Figure 4.11. PCA scores plot generated from the mass spectral data (7.6-7.8 region) obtained from *E coli and S. aureus* cell lysates. A four component model was obtained with a cumulative R^2 of 0.968.

?

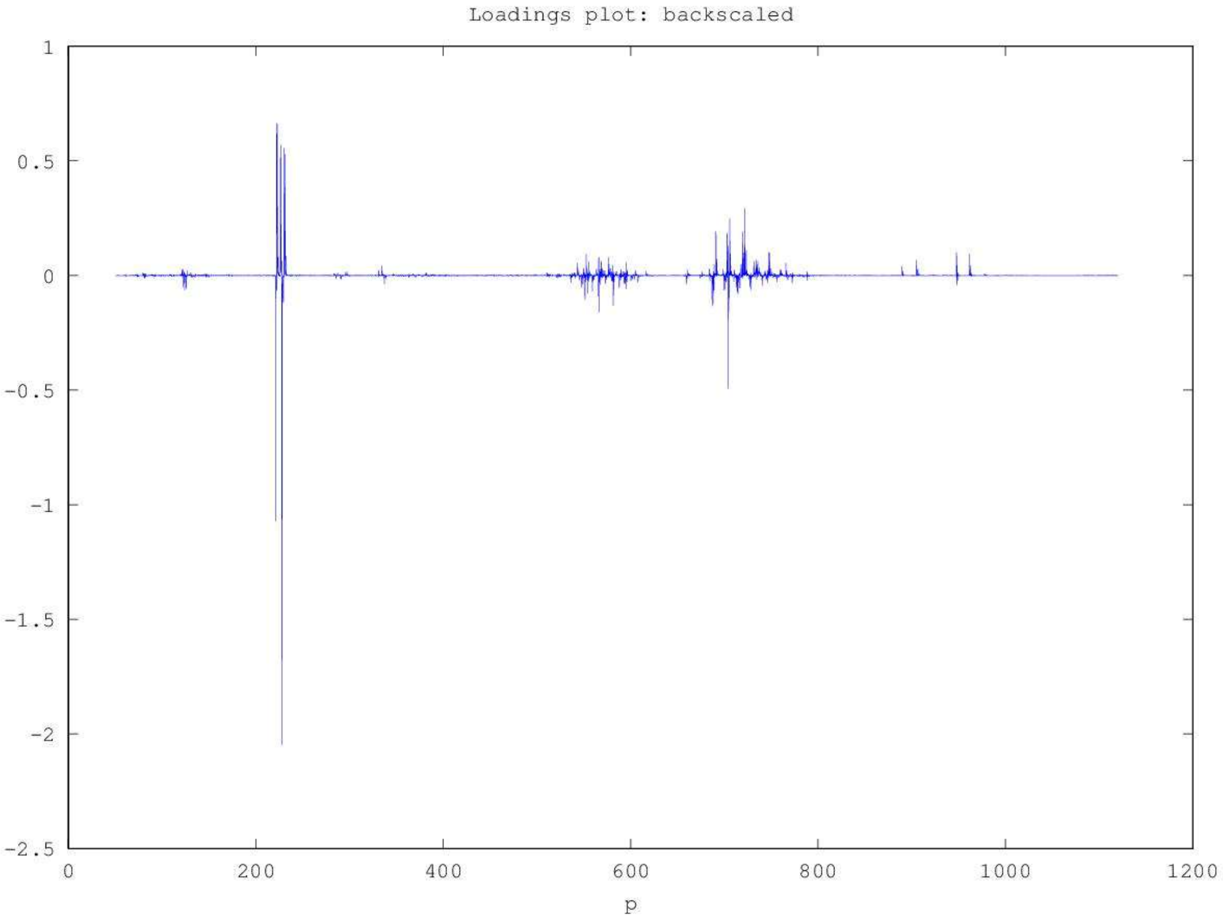


Figure 4.12. A backscaled loadings plot based on the weights obtained from the model in figure 10. It can

4.6 Conclusion

Sparse matrices were successfully employed to reduce the effective size of LC- MS data without any appreciable loss of data quality or information. Sparse matrices were key to the implementation of an LC-MS data processing pipeline within our MVAPACK metabolomics toolkit. Importantly, the data processing utilizes the *entire* LC-MS dataset, which is an important advancement over traditional approaches that rely on thresholding, or peak picking to reduce the size of the data matrix. These simple data reduction methods result in a dramatic reduction in size of the entire LC-MS dataset, allowing it to be processed in an interactive way with the hardware available on a typical user PC. A total of 10 new functions have been added to MVAPACK that enable inputting, visualization, and processing, of LC-MS. Thus, MVAPACK can now uniquely process both NMR and mass spectral data and generate a variety of multivariate models. Importantly, the NMR and mass spectral data can be integrated into a *single* statistical model using multi-block (MB) models (i.e., MB-PCA, MB-PLS, and MB-OPLS). MVAPACK has been developed using GNU Octave, which offers a unique level of customizability that enables a robust and flexible combination of functions that differs dramatically from of online “black-box” processing utilities. MVAPACK is open-source software with a simple syntax that can be easily modified and extended to include additional functions as the metabolomics field continues to evolve.

4.7 References

1. Dunn, W. B., Bailey, N. J. C. & Johnson, H. E. Measuring the metabolome: current analytical technologies. *Analyst* **130,** 606–625 (2005).

2. Griffiths, W. J. *et al.* Targeted metabolomics for biomarker discovery. *Angewandte Chemie - International Edition* **49,** 5426–5445 (2010).

3. Johnson, C. H., Ivanisevic, J. & Siuzdak, G. Metabolomics: beyond biomarkers and towards mechanisms. *Nat Rev Mol Cell Biol* **17,** 451–459 (2016).

4. Kell, D. B. Metabolomics and systems biology: Making sense of the soup. *Curr. Opin. Microbiol.* **7,** 296–307 (2004).

5. Deng, L. *et al.* Combining NMR and LC/MS using backward variable elimination: Metabolomics analysis of colorectal cancer, polyps, and healthy controls. *Anal. Chem.* **88,** 7975–7983 (2016).

6. Heinzmann, S. S., Holmes, E., Kochhar, S., Nicholson, J. K. & Schmitt-Kopplin, P. 2-Furoylglycine as a Candidate Biomarker of Coffee Consumption. *J. Agric. Food Chem.* **63,** 8615–8621 (2015).

7. Fan, Y. *et al.* Applying random forests to identify biomarker panels in serum 2D-DIGE data for the detection and staging of prostate cancer. *J. Proteome Res.* **10,** 1361–1373 (2011).

8. Nicholson, G. *et al.* A genome-wide metabolic QTL analysis in europeans implicates two Loci shaped by recent positive selection. *PLoS Genet.* **7,** (2011).

9. Lloyd, G. R., Wongravee, K., Silwood, C. J. L., Grootveld, M. & Brereton, R. G. Self Organising Maps for variable selection: Application to human saliva analysed by nuclear magnetic resonance spectroscopy to investigate the effect of an oral healthcare product. *Chemom. Intell. Lab. Syst.* **98,**

149–161 (2009).

10. Delaglio, F. *et al.* NMRPipe: A multidimensional spectral processing system based on UNIX pipes. *J. Biomol. NMR* (1995). doi:10.1007/BF00197809

11. Liu, Z., Abbas, A., Jing, B.-Y. & Gao, X. WaVPeak: picking NMR peaks through wavelet-based smoothing and volume-based filtering. *Bioinformatics* **28,** 914–920 (2012).

12. Worley, B. & Powers, R. MVAPACK: A complete data handling package for NMR metabolomics. *ACS Chem. Biol.* (2014). doi:10.1021/cb4008937

13. Katajamaa, M. & Orešič, M. Processing methods for differential analysis of LC/MS profile data. *BMC Bioinformatics* **6,** 179 (2005).

14. Bingol, K. *et al.* Metabolomics beyond Spectroscopic Databases: A Combined MS/NMR Strategy for the Rapid Identification of New Metabolites in Complex Mixtures. *Anal. Chem.* **87,** 3864–3870 (2015).

15. Kessner, D., Chambers, M., Burke, R., Agus, D. & Mallick, P. ProteoWizard: Open source software for rapid proteomics tools development. *Bioinformatics* **24,** 2534–2536 (2008).

Chapter 5. Conclusion

5.1 Future Directions for Data Processing

The field of metabolomics, particularly with respect to data processing, is still in its formative years. While a variety of chemometric tools have been used to interrogate large metabolomics datasets, there are still algorithms, such as Random Forest (RF)[1] and Support Vector Machines (SVM)[2], that have seen limited utilization. This is primarily due to the paucity of software tools available to the scientific community. The same lack of adequate software packages plagues many areas of metabolomics, and the development of software is crucial to advancing the field by evaluating an algorithms utility as well as uniting the experimental results from multiple instrumental methods. New analytical methods such as dynamic nuclear polarization for NMR and ion mobility for MS, along with other instrumental advancements yet to be made, offer new avenues for exploring and investigating the metabolome.

The pre-processing of NMR data, as outlined in chapter 3, has a delicate interplay with the results of statistical modeling. It requires experience in statistics in order to evaluate the proper choices, and how properties inherent to the dataset being considered can completely obfuscate the true underlying factors when employed with poor judgement. Making the community aware of these pitfalls, not only with the process of normalization but will all processing routines, will hopefully generate greater awareness of their importance and generate more comparative assessments and discussion of similar pitfalls in other user-defined approaches such as binning and peak selection in NMR and deisotoping and gap-filling in MS in the context of metabolomics.

LC-MS has cemented itself as a requisite method for performing metabolomics and other chemometric investigations. However, with the growing awareness of the synergistic information contained from both NMR and LC-MS, it is likely that other instrumental methods such as CE and FT-IR will see increased use as well. As the number of instruments that can be used rises, the need for a single data analysis platform becomes increasingly apparent in order to unify the characteristic metabolic signatures that each method is uniquely capable of observing. The ability to process LC-MS data in MVAPACK, in combination with its demonstrated utility for NMR data, represents a fundamental advance in multi-platform metabolic investigations. Utilizing open-source software allows the user to observe the underlying statistical processes, allowing reasearchers to ensure that algorithms are being appropriately applied to their dataset in question. MVAPACK eliminates the "black-box" approach to data analysis and opens the way for investigators to develop their own methods to analyze and interpret their data in whichever ways demonstrate themselves to be best-suited to the task. The methods outlined in this thesis as well as those yet to be developed, not only need to be further refined, but a particular emphasis on their distribution into simple software packages is crucial to their wider adoption by the scientific community. The principal burden halting researchers from performing more advanced analyses is simply not having these tools freely available, verified to work together and with clear instructions on their proper use. An excellent example is multiway analysis, which currently is only being performed routinely be a few research groups, and is almost exclusively restricted to commercial users of MATLAB[3]. Additional examples include other multivariate analysis approaches that have significant followings in other fields, but are largely unproven in the metabolomics community such as the random forest[4] whose applicability to this field will become more apparent once they become available in standardized platforms such as MVAPACK. It is speculated again that this is largely the result of them not being included as options in most software that has been distributed in the metabolomics community to date, and their implementation is not straightforward. The turning point will be when such algorithms are freely shared between all disciplines, and becomes just as simple to apply to metabolomic data as the methods of OPLS and PCA which are already commonplace. Getting to this point will require ongoing effort from chemists, statisticians and computer scientists to ensure that processes are simple, well-documented and evolving to meet the new challenges we encounter.

Lastly, a trend of simply adding another dimension to our instrumentation (2/3-dimensional NMR or LC-LC-MS for example) has become accepted as a method of dealing with the complexity of biologically derived datasets that metabolomics concerns itself with. While this is fine for univariate studies where resolution of a single, known metabolite can be targeted with a high degree of confidence, it only magnifies the issues of data size that were presented in chapter 4. Additionally, it especially complicates analysis due to the exponential increase in data size on complex samples; the unknown composition of metabolomics samples often means that biological insight cannot be used to remove regions of uninformative data. Great strides in chemometrics are being made in the area of multiway analysis[3,5–7], which concerns itself with developing variants of modeling algorithms suitable for high dimensional data. While their utility in application-driven metabolomics is still largely undemonstrated, this may present another promising avenue of research and quantify changes arising from multiple instrumental techniques.

5.2 References

1. Chen, T. *et al.* Random forest in clinical metabolomics for phenotypic discrimination and biomarker selection. *Evid. Based. Complement. Alternat. Med.* **2013,** 298183 (2013).

2. Heinemann, J., Mazurie, A., Tokmina-Lukaszewska, M., Beilman, G. J. & Bothner, B. Application of support vector machines to metabolomics experiments with limited replicates. *Metabolomics* 1121–1128 (2014). doi:10.1007/s11306-014-0651-0

3. Bro, R. Review on multiway analysis in chemistry - 2000-2005. *Critical Reviews in Analytical Chemistry* (2006). doi:10.1080/10408340600969965

4. Breiman, L. Random Forests. *Mach. Learn.* **45,** 5–32 (2001).

5. Bro, R. Multiway calibration. Multilinear PLS. *J. Chemom.* (1996). doi:10.1002/(SICI)1099-128X(199601)10:1<47::AID-CEM400>3.0.CO;2-C

6. Andersson, C. A. & Bro, R. The N-way Toolbox for MATLAB. *Chemom. Intell. Lab. Syst.* (2000). doi:10.1016/S0169-7439(00)00071-X

7. Gurden, S. P., Westerhuis, J. A., Bro, R. & Smilde, A. K. A comparison of multiway regression and scaling methods. *Chemom. Intell. Lab. Syst.* (2001). doi:10.1016/S0169-7439(01)00168-X

Appendix of New MVAPACK Functions

Function 1. loadmzxml.m

```
function [mz,intensity] = loadmzxml(filename)

%Lines of this form are comments to the reader for clarity

file_as_string = fileread(filename);

[ax,bx,cx,dx,ex,fx,gx] = regexp(file_as_string, 'cvParam: m/z array, m/z\s+binary:\s\[\d+\]\s[\d\.\s]+');

[ay,by,cy,dy,ey,fy,gy] = regexp(file_as_string, 'cvParam: intensity array, number of detector
counts\s+binary:\s\[\d+\]\s[\d\.\s]+');

x=dx;

y=dy;

%Remove extraneous header information

[a,b,c,d,e,f,x_stripped] =regexp(x, 'cvParam: m/z array, m/z\s+binary: \[\d+\]');

[a,b,c,d,e,f,y_stripped] =regexp(y, 'cvParam: intensity array, number of detector counts\s+binary: \[\d+\]');

x=cell

y=cell

%Unpack data from hierarchy.

        for i = 1:length(x_stripped)

                temp=x_stripped{i};

                x_unpacked=temp{2};

                x_trim=strtrim(x_unpacked);

                x_split=strsplit(x_trim);

                %results in a cell array of m/z values for each scan.

                mz{i}=x_split;

        endfor

        %Now for the intensities.

        for i = 1:length(y_stripped)

          temp=y_stripped{i};

          y_unpacked=temp{2};

          y_trim=strtrim(y_unpacked);

          y_split=strsplit(y_trim);
```

```
            %results in a cell array of intensity values for each scan.
            intensity{i}=y_split;
        endfor
endfunction
```

Function 2. Plotscan.m

```
function [imputed] =plotscan(mz_axis, mz_array, intensities, scan_number)
if (nargin != 4)
        print_usage();
endif

imputed=zeros(size(mz_axis));
for i = 1:length(mz_array{scan_number})
peak=str2num(mz_array{scan_number}{i});
idx=lookup(mz_axis,peak);
imputed(idx)=str2num(intensities{scan_number}{i});
endfor

#Progress Display
%printf('Size of mz axis %d \n', size(mz_axis));
%printf('Size of imputed %d \n', size(imputed));
%fflush(stdout);
plot(mz_axis, imputed);
endfunction
```

Function 3. Plotscan3d.m

```
function [] = plotscan3D(x,y,scan_number)
#This function will use data stored in cell arrays.


#m/z values (x axis)
        mz =str2double(x{scan_number});


#intensity values (y axis)
        intensity = str2double(y{scan_number});
#Plot the newly formatted results.
        plot(mz, intensity);
endfunction
```

Function 4. Threshold.m

```
function [features,level] = threshold(x,level)
  % check for proper arguments.
  if (!any(nargin == [1 : 2]) || !any(nargout == [1 : 2]))
    % improper arguments. print the usage statement.
    print_usage();
  end


  % see if a threshold value was provided.
  if (nargin < 2 || isempty(level) || !isscalar(level))
    %No thresholding value provided. Print usage.
    print_usage();
  end


  %Decompose sparse matrix into component vectors
  [rt,mz,int]=find(x);

  %modify the intensity values in-situ
  int(int<level)=0;

  %Generate New Sparse matrix
  features=sparse(rt,mz,int);
end
```

Function 5. Get_axes.m

```
function [mz_axis, rt_axis]=get_axes(mz_array, filename)

#Step 0: Load in the file
file_as_string = fileread(filename);




%Build the m/z axis
%Pull the data out the nested cell arrays
mz_cell=cell2mat(mz_array);
mz_axis=cell2mat(mz_cell);

#Sort the m/z values and remove duplicates
mz_axis=unique(mz_axis);

#Now let's get the rt axis.
#refer to the loadmzxml function for a quick example of extracting RT's.
```

```
[~,~,~,rt_instances,~,~,~]=regexp(file_as_string, 'cvParam: scan start time, [\d]+[\.,][\s\d]+[\s\S]');
[~,~,~,rt_axis,~,~,~]=regexp(rt_instances, '[\d\.]+');

for i= 1:length(rt_axis)
        rt_axis{i}{1}=str2num(rt_axis{i}{1});
        printf("Processing Retention Time: %d \n", i);
        fflush(stdout);
endfor
#Build the axis
rt_axis=cell2mat(rt_axis);
rt_axis=cell2mat(rt_axis);




endfunction
```

Function 6. Mzxmlinfo.m

```
function [info]= mzxmlinfo(filename)



%Garbage Test Function
%function [elem,data,att,StartTime] = mzxmlinfo

%Add path to xerces parsing files
javaaddpath('/path/to/xerces /xercesImpl.jar');
javaaddpath('/path/to/xerces/xml-apis.jar');
```

```
%Emulates behavior of xmlread in matlab

parser = javaObject('org.apache.xerces.parsers.DOMParser');

parser.parse(filename);

xDoc = parser.getDocument;


%Examples of Xerces Use

%elem = xDoc.getElementsByTagName('precursorMz').item(0);

%data = elem.getFirstChild.getTextContent;

%att = elem.getAttribute('precursorIntensity');


%FileSize

[file_info, err, msg] = stat(filename);

filesize = file_info.size;



%Number of Scans;stored in NumScans

msRun = xDoc.getElementsByTagName('msRun').item(0);

NumScans = msRun.getAttribute('scanCount');

%if NumScans not empty

%return NumScans

%else (IE, if empty)

%return "Not given; check the file."
```

```
%Start Time of Run
msRunStart = xDoc.getElementsByTagName('msRun').item(0);
StartTime = msRunStart.getAttribute('startTime');

%End Time of Run
msRunEnd = xDoc.getElementsByTagName('msRun').item(0);
EndTime = msRunEnd.getAttribute('endTime');

%Data Pre-processing Information

%Is it centroided?
centroid = xDoc.getElementsByTagName('dataProcessing').item(0);
centroided = centroid.getAttribute('centroided');
if centroided == "1"
   centroided = "Yes";
else
   centroided = "N/A";
endif
%Note, if centroid is ==1, the data has been previously centroided.

%Is it deisotoped?
deisotope = xDoc.getElementsByTagName('dataProcessing').item(0);
deisotoped = deisotope.getAttribute('deisotoped');
if deisotoped == "1"
   deisotoped = "Yes";
else
   deisotoped = "N/A";
endif
%Note, if deisotoped is ==1, the data has been previously centroided.
```

```
%Is it charge deconvoluted?
chargeDeconvo = xDoc.getElementsByTagName('dataProcessing').item(0);
chargeDeconvoluted = chargeDeconvo.getAttribute('chargeDeconvoluted');
if chargeDeconvoluted == "1"
   chargeDeconvoluted = "Yes";
else
   chargeDeconvoluted = "N/A";
endif

%Note, if chargeDeconvoluted is ==1, the data has been previously centroided.

%Spot-Integrated?
spotInt = xDoc.getElementsByTagName('dataProcessing').item(0);
spotIntegrated = spotInt.getAttribute('spotIntegrated');
if spotIntegrated =="1"
   spotIntegrated = "Yes";
else
   spotIntegrated = "N/A";
endif
%Note, if spotIntegrated is ==1, the data has been previously centroided.


%Tandem?
elem = xDoc.getElementsByTagName('dataProcessing').item(0);
```

```
info = struct("Filename", filename, "Filesize", filesize, "Number of Scans", NumScans, "Start Time",
StartTime, "End Time", EndTime, "Centroided?", centroided, "Deisotoped?", deisotoped, "Charge
Deconvoluted?",
chargeDeconvoluted, "Spot Integrated?", spotIntegrated);
endfunction
```

Function 7. Sparse_matrix_load.m

```
function [mz_list,intensity_list,info] = sparse_matrix_load(filenames,verbose)
if (!nargin == 1:2 || !nargout == 2:3)
        print_usage();
endif

%Check if single or multiple files are presented
if (ischar(filenames))
  filenames={filenames};
elseif (iscellstr(filenames))
  if (length(filenames) == 0)
  %Error. Warn user.
  error('sparse_load: zero files supplied. Check your filenames array');
  endif
endif

%Initialize a structure to store the data
data_struct=struct();
data_struct.files=filenames;
data_struct.sample=cell(size(data_struct.files));
```

```
%Load the file and begin munging.

for i=1:length(filenames)

        file_as_string = fileread(filenames{i});
  #check for version incompatibilities  with fileread()
  v=version;
  if ((v="3.6.2"))
    file_as_string = rot90(file_as_string);
  endif
  %Match the scans
  [~,~,~,rt_instances,~,~,text_splits]=regexp(file_as_string, 'cvParam: scan start time,
[\d]+[\.,][\s\d]+[\s\S]');




  %Get the mz values
  [~,~,~,mz_splits,~,~,~] = regexp(text_splits, 'cvParam: m/z array, m/z\s+binary:\s\[\d+\]\s[\d\.\s]+');
  [~,~,~,~,~,~,mz_stripped]= regexp(mz_splits, 'cvParam: m/z array, m/z\s+binary: \[\d+\]');
  %Partitioning the mz values
  mz_list={};
  for i= 2:length(mz_stripped)
          temp =mz_stripped{i};
          if (length(temp)==2)
                  mz_unpacked=temp{2};
                  mz_trim=strtrim(mz_unpacked);
```

```
                mz_split=strsplit(mz_trim, ' ');
                mz_list{(i-1)}=mz_split;% (i-1) since first entry will always be non-pertinent metadata
        else
                mz_list{(i-1)}={};
        endif
endfor


%Get the intensity values
[~,~,~,intensity_splits,~,~,~] = regexp(text_splits, 'cvParam: intensity array, number of detector
counts\s+binary:\s\[\d+\]\s[\d\.\s]+');
[~,~,~,~,~,~,int_stripped] =regexp(intensity_splits, 'cvParam: intensity array, number of detector
counts\s+binary: \[\d+\]');
%Partitioning the intensity values
intensity_list={};
for i= 2:length(int_stripped)
    temp =int_stripped{i};
    if (length(temp)==2)
        int_unpacked=temp{2};
        int_trim=strtrim(int_unpacked);
        int_split=strsplit(int_trim, ' ');
    intensity_list{(i-1)}=int_split; %First entry is metadata
    else
        intensity_list{(i-1)}={};
    endif
endfor


endfor
```

```
  %Generate the retention time and m/z axes.
  %Match the scans first.
   [~,~,~,rt_instances,~,~,text_splits]=regexp(file_as_string, 'cvParam: scan start time, [\d]+[\.,][\s\d,]+[\s]+second');
  %NOTE this regex pattern is different (and more correct) than the one used in the loadmzxml function.
  %Still cant figure out how to extract the actual units (assuming seconds?) from the file itself.
endfunction


%Determine whether to display function progress
#{
if (verbose="True")
  scan=0
  for i=1:
    printf("Processing Scan #: %d \n", scan);
    fflush(stdout);
 sleep(1); (not required since for loop itself takes comp time)
 scan++;
 endfor
endif
#}


  #{
  %Sample Metadata and Information Extraction
  %Empty Scan Information
  empty_scans=[];
  for i = 1:length(mz)
    if length(mz{i})==0
    empty_scans(end+1)=i;
    endif
  endfor


  info.text=sprintf("This sample contains %d empty scans", length(empty_scans));
  info.empty_scans=empty_scans;


  #}
endfunction
```

Function 8. Array_converter.m

```
function [numeric_data_array]=array_converter(string_data_array)



string_data_array=strsplit(string_data_array);
string_array=string_data_array(4:end-1);
numeric_data_array=cellfun(@str2num,string_array);

endfunction
```

Function 9. Loader_function.m

```
#Load the text strings corresponding to the m/z and int values into octave.
#Load the datafile
name='insertfilenamehere.txt';
file=fopen(name);

####2. Iterates and constructs the data array
#Note, file should already be MSConvert output and CWT filter applied.####
#init values
data={};
i=1;
#start loop
while(!feof(file))
string=fgetl(file);
data{i}=string;
i++; #leave out semicolon if you want to monitor progress.
printf("Processing Line#: %d\n",i);
fflush(stdout);
endwhile
endfunction
```