Universitat Politècnica de Catalunya

Doctoral Thesis

Machine Learning Methods for the Analysis of Liquid Chromatography-Mass Spectrometry datasets in Metabolomics

Author:
Francesc Fernández Albert

Supervisors:
Dr Alexandre Perera Lluna
Dr Rafael Llorach Asunción

A thesis submitted in fulfilment of the requirements for the degree of Doctor in Biomedical Engineering

B2SLab
Department d'Enginyeria de Sistemes, Automàtica i Informàtica Industrial

October 2014

UNIVERSITAT POLITÈCNICA
DE CATALUNYA
BARCELONATECH

If you thought that science was certain... well, that is just an error on your part!

Richard P. Feynman

Phfft! Facts! Facts are meaningless. You can use them to prove anything that is even remotely true.

Homer J. Simpson

"¡Ésto peta y repeta!"

Dr Rafael Llorach debugging the MAIT package

"El teorema del jardín? Es quan un es posa a explicar massa coses i acaba en un jardín, llavors... és igual, treu aquesta part.

Dr Alexandre Perera "debugging" some (or any) of my papers

UNIVERSITAT POLITÈCNICA DE CATALUNYA

Abstract

Departament d'Enginyeria de Sistemes, Automàtica i Informàtica Industrial

Doctor of Philosophy

Machine Learning Methods for the Analysis of Liquid Chromatography-Mass Spectrometry datasets in Metabolomics

by Francesc FERNÁNDEZ ALBERT

Liquid Chromatography-Mass Spectrometry (LC/MS) instruments are widely used in Metabolomics. To analyse their output, it is necessary to use computational tools and algorithms to extract meaningful biological information. The main goal of this thesis is to provide with new computational methods and tools to process and analyse LC/MS datasets in a metabolomic context. A total of 4 tools and methods were developed in the context of this thesis.

First, it was developed a new method to correct possible non-linear drift effects in the retention time of the LC/MS data in Metabolomics, and it was coded as an R package called HCor. This method takes advantage of the retention time drift correlation found in typical LC/MS data, in which there are chromatographic regions in which their retention time drift is consistently different than other regions. Our method makes the hypothesis that this correlation structure is monotonous in the retention time and fits a non-linear model to remove the unwanted drift from the dataset. This method was found to perform especially well on datasets suffering from large drift effects when compared to other state-of-the art algorithms.

Second, it was implemented and developed a new method to solve known issues of peak intensity drifts in metabolomics datasets. This method is based on a two-step approach in which are corrected possible intensity drift effects by modelling the drift and then the data is normalised using the median of the resulting dataset. The drift was modelled using a Common Principal Components Analysis decomposition on the Quality Control classes and taking one, two or three Common Principal Components to model the drift space. This method was compared to four other drift correction and normalisation methods. The two-step method was shown to perform a better intensity drift removal than all the other methods. All the tested methods including the two-step method were coded as an R package called intCor and it is publicly available.

Third, a new processing step in the LC/MS data analysis workflow was proposed. In general, when LC/MS instruments are used in a metabolomic context, a metabolite may give a set of peaks as an output. However, the general approach is to consider each peak as a variable in the machine learning algorithms and statistical tests despite the important correlation structure found between those peaks coming from the same source metabolite. It was developed an strategy called peak aggregation techniques, that allow to extract a measure for each metabolite considering the intensity values of the peaks coming from this metabolite across the samples in study. If the peak aggregation techniques are applied on each metabolite, the result is a transformed dataset in which the variables are no longer the peaks but the metabolites. 4 different peak aggregation techniques were defined and, running a repeated random sub-sampling cross-validation stage, it was shown that the predictive power of the data was improved when the peak aggregation techniques were used regardless of the technique used.

Fourth, a computational tool to perform end-to-end analysis called MAIT was developed and coded under the R environment. The MAIT package is highly modular and programmable which ease replacing existing modules for user-created modules and allow the users to perform their personalised LC/MS data analysis workflows. By default, MAIT takes the raw output files from an LC/MS instrument as an input and, by applying a set of functions, gives a metabolite identification table as a result. It also gives a set of figures and tables to allow for a detailed analysis of the metabolomic data. MAIT even accepts external peak data as an input. Therefore, the user can insert peak table obtained by any other available tool and MAIT can still perform all its other capabilities on this dataset like a classification or mining the Human Metabolome Dataset which is included in the package.

Acknowledgements

I know that this section might be the most read section of this document so I will try to thoroughly thank and acknowledge any individual (or set of) related to this work and I will also try not to disappoint anyone expecting some memorable words. It might seem a cliché, but first of all I really would like to express my sincere gratitude to my advisors Dr Rafael Llorach (AKA Rafa) and Dr Alexandre Perera (AKA Alex). Both of them taught me so many things these years that would not fit in these pages, so I will just say that if I am far better scientist than four years ago, it is merely their merit. I especially thank Rafa for his extreme degree of patience to perform a thorough debugging process of the MAIT package by running uncountable data analysis and pointing me to possible inconsistencies and errors in the coding. He even managed to insert some really biological/chemical knowledge into a mind of a physicist. I think that is really an accomplishment, especially because I am that physicist. I would like to especially thank Alex for his overall guidance throughout all my thesis, especially when I was a freshman and I still had to learn the tools of the trade. He was also patient enough that he never said a word when I was popping up continuously to his office to ask him (mostly stupid) questions. He is also taught me the Garden's Theorem, the Tararí Effect and the OWA's Method which I found of paramount importance in science and which I will definitely not forget (do not worry Alex!). I would also like to say that I consider Alex and Rafa not only my mentors but also my friends.

I would like to thank Dr Pere Caminal and Dr Cristina Andrés-Lacueva for letting me stay and work in their groups. Working in these two groups having so different scientific focus truly enriched me as a scientist. I really appreciate that they treated me as a member of their own team since the very beginning. Thanks to all the staff from the nutrimetabolomics group in UB for making me feel at home while I was there. Very special thanks to Dr Mar Garcia Aloy who not only recorded the large datasets I used for the intensity and retention time drift chapters, but also provided me with the necessary data to code the drift correction methods.

Very special thanks to the people from the gsisbio group (now B2SLab), especially to the PhD students with whom I shared my good and bad moments as well as theirs. In particular, thanks a lot to Jan Maynou to teach me many things (probably more than he thinks), especially that the end is not as important as the way to get there and the people you meet in that way; to Dr Helena Brunel who is one of the strongest people I know (even though she will deny it), for teaching me to persevere and for being a better FC Barcelona fan than myself (shame on me!); to Raimon Massanet for teaching me that there are some people that can repair computers and printers using magic (I think it is some kind of computer engineers' spell...) and for enlighten me with the

Pastafarianism; to Andrey Ziyatdinov, who I expect by the time I will be defending this thesis he will be already a doctor (congrats!), for teaching me to stay calm in moments I was nervous, for learning how to improve my coding skills in R, for not getting mad about every time I asked him how to pronounce/write his surname and for not following my advice of installing a cannon on the top of his robot; To Giovana Gavidia with whom I shared the lab for the last months and to Samir Kanaan, keep it up! I hope to be in your PhD defence the coming year! To Sergi Picart, who is inheriting my research line in the group, I have no doubt he will perform better than me, so I will just say to him good luck (especially when you will be in Australia) and keep in touch.

Another group of special thanks for the people in the Bioengineering Institute of Catalonia (IBEC) with whom I share the lunch everyday. Thank you very much to Dr Leonardo Sarlabous for listening me when I am having lunch annoyed by some reason and by teaching me the mysteries that underly in the PIN codes of the SIM cards; to Rudys Magrans who is a very nice guy and whom I will encourage to finish his thesis until the very last day; to Puy Ruiz de Alba for being always happy and friendly to everyone and the best DJ I know; to Andrés Arcentales with whom I have shared a classroom first and later a lab for the last 6 years, for being always there to share a laugh or a joke despite singing to loud and not very well.

Another thanks all the pairs of running shoes I have shredded during the whole thesis that allowed me to get rid of the great deal of stress I usually accumulated and to the coffee machines that provided me with enough caffeine to work every day and to shred those shoes.

I cannot remark enough in words how thankful I am to my family and to my parents. For being always there when I needed support even for the slightest thing, and for providing me with guidance when I felt lost. Moltes gràcies.

And finally the biggest thank you goes to the person who really helped me the most in carrying on not only this thesis, but also my entire life. For listening and cheering me up everyday when I told her about my worries and complaints, for continuously encouraging me to follow my dreams even when that interfered with hers, and for just being herself, to my wife Anna.

Contents

List of Figures

List of Tables

Abbreviations

NMR	**N**uclear **M**agnetic **R**esonance
GC/MS	**G**as **C**hromatography **M**ass **S**ectrometry
LC/MS	**L**iquid **C**hromatography **M**ass **S**ectrometry
MS	**M**ass **S**pectrometry devices
HPLC-MS	**H**igh **P**erformance **LC/MS**
UPLC-MS	**U**ltra **P**erformance **LC/MS**
ESI	**E**lectro**S**pray **I**onisation
TOF	**T**ime **O**f **F**light
NIST	**N**ational **I**nstitute of **S**tandards and **T**echnology
NIH	**N**ational **I**nstitute of **H**ealth
GMD	**G**olm **M**etabolome **D**atabase
PLS	**P**artial **L**east **S**quares
PCA	**P**rincipal **C**omponent **A**nalysis
PC	**P**rincipal **C**omponent
PLSDA	**P**artial **L**east **S**quares and linear **D**iscriminant **A**nalysis
OPLS	**O**rtogonal **P**artial **L**east **S**quares
OPLSDA	**O**rtogonal **P**artial **L**east **S**quares and linear **D**iscriminant **A**nalysis
SVM	**S**upport **V**ector **M**achine
ROI	**R**egions **O**f **I**nterest
CWT	**C**ontinuous **W**avelet **T**ransform
LOESS	**LO**cally w**E**ighted **S**catterplot **S**moothing
PTW	**P**arametric **T**ime **W**arping
QC	**Q**uality **C**ontrols
FDR	**F**alse **D**iscovery **R**ate
HMDB	**H**uman **M**etabolome **D**ata**B**ase

SOM	**S**elf **O**rganising **M**aps
NMF	**N**on-negative **M**atrix **F**actorisation
MAIT	**M**etabolite **A**utomatic **I**dentification **T**oolkit
CC	**C**omponent **C**orrection

Symbols

m/z	ratio between the mass and the charge	Da
rt	retention time	s or min

Chapter 1

Introduction

1.1 The Metabolome

The genome of a specie is defined as the set of genes that comprises all its hereditable material [1]. The proteome for that specie is made of all the proteins that are obtained from the genome of that specie [2]. In a similar way, the metabolome of that specie is defined as the set of metabolites involved in the metabolic reactions of that specie [3]. The corresponding sciences that study the genome, the proteome and the metabolome are called Genomics, Proteomics and Metabolomics respectively. It is known that these three omics show deep and complex links between them. For example some metabolite abundance variations seem to be related to some associated genetic variances [4]. Although being related, there is an important difference between the Human Genome and the Human Metabolome: the Human Genome has been sequenced, which means that the human DNA sequence is known [5], whereas the exact size of the Human Metabolome at the present time is unknown [6].

1.2 Experimental Devices

In Metabolomics it is used a wide range of analytical devices to obtain empirical metabolite measures [6]. Gas Chromatography coupled to Mass Spectrometry (GC/MS) and NMR platforms were the first devices used in the early stages of Metabolomics [7, 8]. The use of Liquid Chromatography coupled to Mass Spectrometry (LC/MS) devices has increased in metabolomic analyses since the development of the High Performance LC/MS (HPLC-MS) and Ultra Performance LC/MS (UPLC-MS) [9].

1.2.1 GC/MS

GC/MS platforms consist in two coupled analytical instruments: a gas chromatograph and a Mass Spectrometer (MS) [10]. The main feature of the gas chromatograph is a large column whose walls are coated with an stationary phase. Through the column it flows a carrier gas called mobile phase. Once a sample is injected into the GC/MS, the mobile phase carries the metabolites in the sample through all the column. These metabolites interact with the stationary phase of the column by means of intermolecular forces. As a consequence, the molecules that in the sample were all mixed, at the end of the column, appear at different times [10]. The time spent for a molecule to get through all the column is defined as retention time (rt) of that molecule. Typical GC/MS single sample analysis lasts for about 20-35 min [11].

At the end of the chromatographic column it is attached the MS. In many MS configurations, the device uses a ionisation source to bombard the molecules to ionise them or to break them down to pieces. Using a mass analyser and a detector, the MS computes the ratio between the mass and the charge (m/z) of the ionised molecules or pieces of molecules. To detect a signal, the mass analysers need these ionised molecules and/or pieces of molecules to have a net electric charge. Among the available mass analysers, the most commonly used are the time-of-flight (TOF) and the single or triple quadrupoles [8, 12]. Other types of available mass analysers are ion traps, orbitrap and fourier transform ion cyclotron resonance [13].

As the main advantages of the GC/MS devices, it is highlighted its high sensitivity and reproducibility of the molecule fragmentation process [11, 14]. Because of this high reproducibility in the fragmentation patterns of the molecules, there are GC/MS mass spectral searchable libraries like the National Institute of Standards and Technology (NIST) / National Institute of Health (NIH) Mass Spectral Library [1] or GOLM

[1] http://www.nist.gov/srd/nist1a.cfm

Metabolome Database [2] that allow for a fast metabolite identification [15]. However, GC/MS platforms can only can detect volatile compounds. Some of the non-volatile molecules can be turned into volatile by applying on them a chemical process known as derivatisation [11, 14]. As a consequence, an GC/MS platform can only profile apolar metabolites which are in the range from volatile to semi-volatile having masses typically under 700 Da [11].

1.2.2 LC/MS

LC/MS platforms (also in its HPLC and UPLC configurations) also consist of two different coupled analytical instruments: a liquid chromatograph and a MS. LC/MS devices have been used widely in Metabolomics and they are probably one of the most widely analytical device used in metabolomic studies at the moment [13, 16]. Figure 1.1 shows a scheme of a LC/MS experimental device. A liquid chromatograph uses a liquid as a mobile phase to transport the sample molecules through the chromatographic column. In Metabolomics, it is specially used the LC/MS setup having an Electrospray Ionisation (ESI) source and a TOF mass analyser. The molecules reaching the electrospray undergo a soft ionisation process called Electrospray Ionisation [8, 17]. In general this ionisation does not break the molecules in the samples but it ionises them, allowing the creation of aggregates with the gain or loss of atoms or molecules. If the resulting aggregate has greater mass than the original molecule it is called an adduct and if it has lower mass, it is a fragment. ESI ionisers generate a great number of ions and they can be set up in either positive or negative polarisation modes. Depending on the characteristics of the molecule, it is easier ionised in the positive or negative polarisation set up. To obtain a complete metabolite profiling of the samples it is required to analyse the samples in both ionisation modes [11]. The reverse phase chromatograph set up is a usual choice when using an LC/MS platform in the Metabolomics context [11]. This configuration allows for suitable analysis of medium and low polarity metabolites. However, there are important metabolites like aminoacids or sugars that are hardly detected under this setup, as they elute in a very small retention time [11, 18]. Due to this fact, it has been developed of a new type of LC/MS device called Hydrophilic Interaction LIquid Chromatography (HILIC) that uses a special chromatographic column to detect this kind of

[2] http://gmd.mpimp-golm.mpg.de/

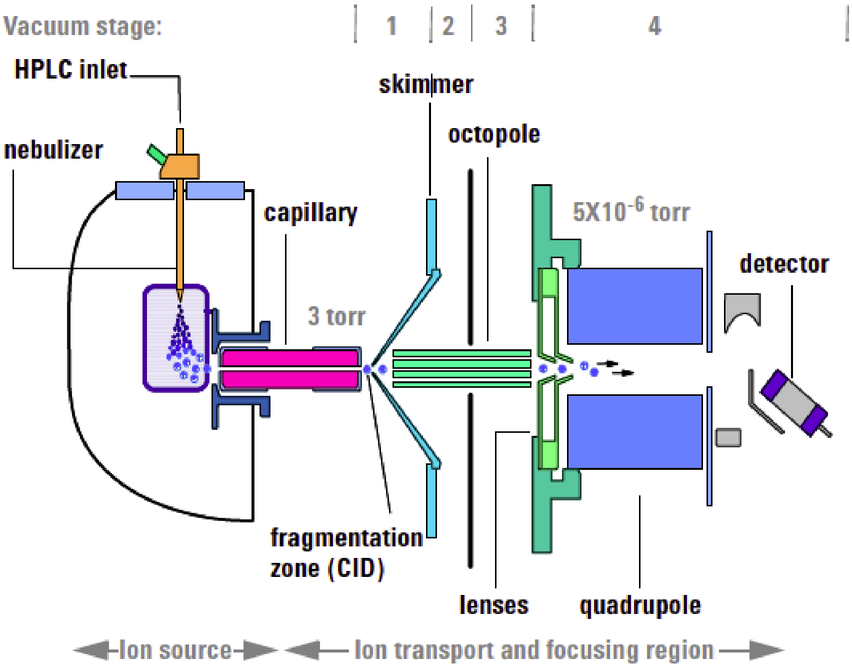


FIGURE 1.1: Diagram of an LC/MS device using a quadrupole as a mass analyser. The model of the picture is a Agilent 6100 Series Quadrupole http://www.kprime.net/pdf/products/6100_Data_Sheet.pdf
.

molecules [11, 18]. It has been suggested that combining the output of the LC/MS with the HILIC might give a more complete profiling of the metabolites in the samples [11]. Typical single sample analysis in LC/MS in Metabolomics lasts for about 10-15 min [11].

In contrast to GC/MS, the samples need not to be derivatised prior to be analysed with a LC/MS device [14, 15]. Other advantages are a wider polarity range and molecular mass compared to GC/MS [15]. On the other hand, LC/MS sample analysis, as all MS-related devices, may suffer from ion suppression effect. This problem is related to the coelution of matrix components with the analytes affecting the device detection capabilities that is reflected in a drop of peaks intensities [11].

Bibliography

[1] Matt Ridley. *Genome: The Autobiography of a Species in 23 Chapters*. NY: Harper Perennial, New York, first edition, 2006. ISBN 0-06-019497-9.

[2] N. Leigh Anderson and Norman G. Anderson. Proteome and proteomics: New technologies, new concepts, and new words. *ELECTROPHORESIS*, 19(11):1853–1861, 1998. ISSN 1522-2683. doi: 10.1002/elps.1150191103.

[3] S G Oliver, M K Winson, D B Kell, and F Baganz. Systematic functional analysis of the yeast genome. *Trends in biotechnology*, 16(9):373–378, 1998.

[4] Eva K. F. Chan, Heather C. Rowe, Bjarne G. Hansen, and Daniel J. Kliebenstein. The complex genetic architecture of the metabolome. *PLoS Genetics*, 6(11): e1001198, 11 2010. doi: 10.1371/journal.pgen.1001198.

[5] Venter J. Craig et al. The sequence of the human genome. *Science*, 291(5507): 1304–1351, 2001. doi: 10.1126/science.1058040. URL http://www.sciencemag.org/content/291/5507/1304.abstract.

[6] David S. Wishart, Timothy Jewison, An Chi Guo, Michael Wilson, Craig Knox, Yifeng Liu, Yannick Djoumbou, Rupasri Mandal, Farid Aziat, Edison Dong, Souhaila Bouatra, Igor Sinelnikov, David Arndt, Jianguo Xia, Philip Liu, Faizath Yallou, Trent Bjorndahl, Rolando Perez-Pineiro, Roman Eisner, Felicity Allen, Vanessa Neveu, Russ Greiner, and Augustin Scalbert. Hmdb 3.0the human metabolome database in 2013. *Nucleic Acids Research*, 2012. doi: 10.1093/nar/gks1065.

[7] E.J. Want, B.F. Cravatt, and G Siuzdak. The future of liquid chromatographymass spectrometry (lc-ms) in metabolic profiling and metabolomic studies for biomarker discovery. *Biomark Med*, 1(1):159–185, 2007.

[8] E.J. Want, B.F. Cravatt, and G Siuzdak. The expanding role of mass spectrometry in metabolite profiling and characterization. *Chembiochem A European Journal Of Chemical Biology*, 711:1941–1951, 2005.

[9] Katherine Hollywood, Daniel R. Brison, and Royston Goodacre. Metabolomics: Current technologies and future trends. *PROTEOMICS*, 6(17):4716–4723, 2006. ISSN 1615-9861. doi: 10.1002/pmic.200600106. URL http://dx.doi.org/10.1002/pmic.200600106.

[10] Kermit K Murray. Glossary of terms for separations coupled to mass spectrometry. *Journal of chromatography A*, pages 209–212, 2010. URL http://www.ncbi.nlm.nih.gov/pubmed/20484981.

[11] G. Theodoridis, H.G. Gika, and I.D. Wilson. Mass spectrometry-based holistic analytical approaches for metabolite profiling in systems biology studies. *Mass Spectrometry Reviews*, 30:884–906, 2011.

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

[13] Lu Xin, Zhao Xinjie, Bai Changmin, Zhao Chunxia, Lu Guo, and Xu Guowang. Lc-ms-based metabonomics analysis. *Journal of Chromatography B*, 866:64–76, 2007.

[14] Zhang Aihua, Sun Hui, Wang Ping, Han Ying, and Wang Hijun. Modern analytical techniques in metabolomics analysis. *Analyst*, 137:293–300, 2012.

[15] Vladimir Shulaev. Metabolomics technology and bioinformatics. *Briefings in Bioinformatics*, 7(2):128–139, 2006. doi: 10.1093/bib/bbl012.

[16] Aurélie Roux, Dominique Lison, Christophe Junot, and Jean-François Heilier. Applications of liquid chromatography coupled to mass spectrometry-based metabolomics in clinical chemistry and toxicology: A review. *Clinical Biochemistry*, 44:119–135, 2011.

[17] John B. Fenn. Electrospray ionization mass spectrometry: How it all began. *Journal of Biomolecular Techniques*, 13(3):101–118, 2002.

[18] Zhou Bin, Xiao Jun Feng, Tuli Leepika, and Ressom Habtom W. Lc-ms-based metabolomics. *Molecular BioSystems*, 8:470–481, 2012.

Chapter 2

State of the Art

Metabolomics has been the latest omic science (after genomics and proteomics) to undergo an important computational development in their data analysis workflows. Metabolomic studies apply experimental instruments and protocols devised in analytical chemistry to biological samples, with special emphasis in analysing biofluids and tissues [1]. Metabolomics aims at detecting and identifying low weight molecules (typically under 1500-1800 Da) called metabolites [2, 3] in biological samples under certain external conditions [4, 5, 6]. Despite the fact that initially there were some differences between the terms Metabolomics and Metabonomics (see the classical Nicholson references [7, 8, 9]) the two terms are nowadays used interchangeably. In this thesis it will only be used the term Metabolomics.

There are two main approaches to perform Metabolomic studies depending on whether the metabolites to be detected are known (Targeted Metabolomics) or they are unknown (Untargeted Metabolomics) [3, 4, 10, 11, 12, 13]. Performing a targeted or an untargeted metabolomic study is a critical issue for sample preparation [10], for the choice of the experimental setup [11] and for the statistical approach used [4, 10]. The basic underlying difference between both types of study is that the main objective in Untargeted Metabolomics is to detect the highest possible amount of metabolites in the samples, whereas Targeted Metabolomics is only focused on detecting certain metabolites of interest.

2.1 Metabolomic Workflow

The main objective in Metabolomics is to study the metabolism using either quantitatively or semi-quantitatively approaches [1, 3, 13]. In metabolomic studies it is followed a well-established pipeline going from experiment design and statistical data analysis to biological interpretation of the results [1, 3, 6, 14]. Figure 2.1 depicts the stages of the typical workflow for metabolomic studies. The main processing steps in a metabolomic analysis include the following stages:

- First steps are focused on an experimental context and they include the experimental design and collecting the biological samples. The samples are gathered following a sample protocol extraction [13, 15]. The data acquisition is performed through analytical instruments such as an LC/MS or a Nuclear Magnetic Resonance (NMR) [4, 11].
- Next stages include computational data processing steps such as peak detection, data filtering and data normalisation (Figure 2.1). One of the objectives of this stage for metabolomic untargeted studies is to obtain which peaks of the system are the most significant in terms of class separability in the data. That means to find which variables separate mostly separate the classes in the dataset. These classes usually refer to different study conditions like patients that have followed different diets or that have undergone some drug treatment. To find these significant masses, the data are analysed through different statistical approaches [16] including multivariate and machine learning methods [3, 17, 18, 19, 20].
- The Statistical Analysis stage shown in Figure 2.1, might include a validation stage, whose objective is to evaluate the statistical predictive power contained in a subset of variables. Different Machine Learning techniques are applied in this stage being Partial Least Squares linear Discriminant Analysis (PLSDA) one of the most used approaches in Metabolomics [17].
- Peak annotation procedure is a processing step to improve the biological interpretation of the metabolomic data. This stage gives biological and chemical insight by labelling the variables (i.e. peaks) of the data with biological and/or chemical information. There are several kinds of peak annotation. Database mining is a type of peak annotation in which a database (or databases) are mined to match the

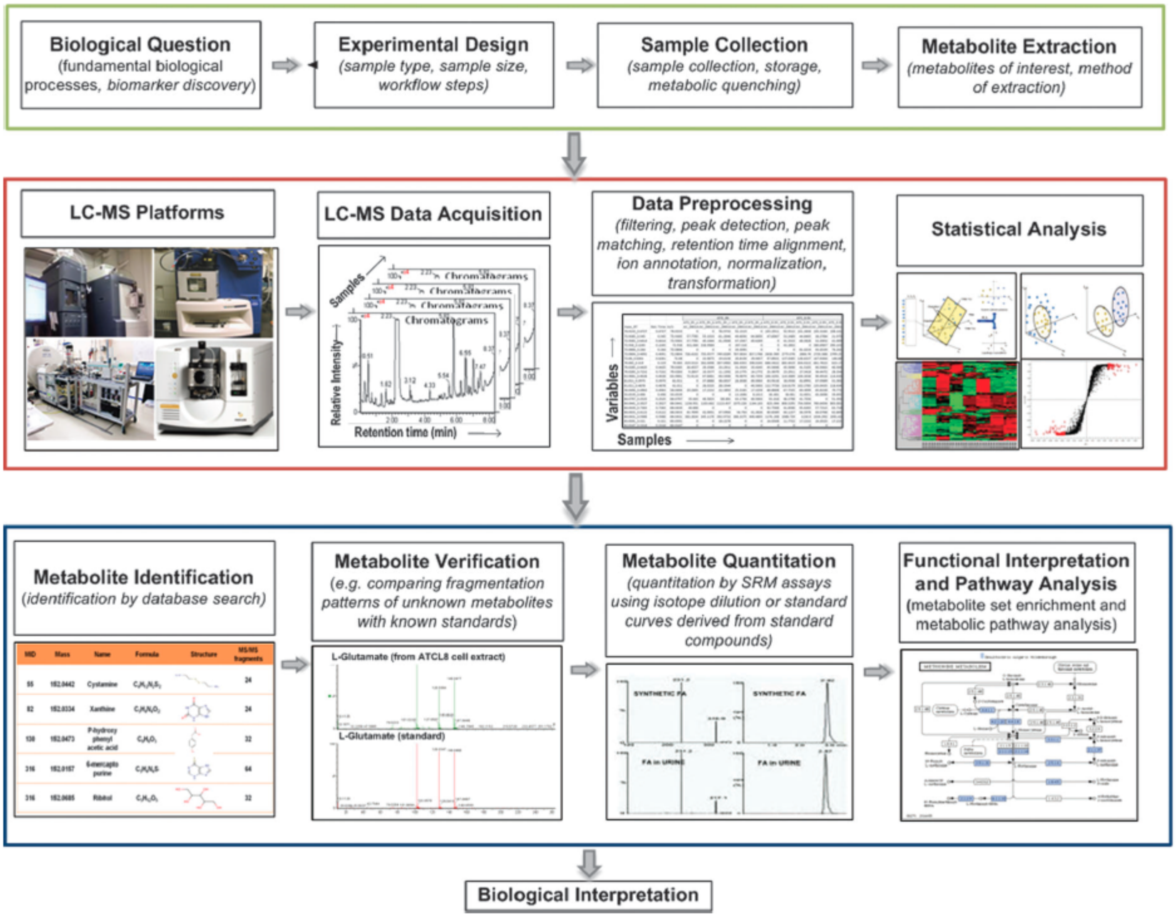

Biological Question
(fundamental biological processes, biomarker discovery)
Experimental Design
(sample type, sample size, workflow steps)
Sample Collection
(sample collection, storage, metabolic quenching)
Metabolite Extraction
(metabolites of interest, method of extraction)
LC-MS Platforms
LC-MS Data Acquisition
Samples
Chromatograms
Relative Intensity
Retention time (min)
Data Preprocessing
(filtering, peak detection, peak matching, retention time alignment, ion annotation, normalization, transformation)
Variables
Samples
Statistical Analysis
Metabolite Identification
(identification by database search)
Metabolite Verification
(e.g. comparing fragmentation patterns of unknown metabolites with known standards)
L-Glutamate (from ATCL8 cell extract)
L-Glutamate (standard)
Metabolite Quantitation
(quantitation by SRM assays using isotope dilution or standard curves derived from standard compounds)
SYNTHETIC FA
SYNTHETIC FA
FA in URINE
FA in URINE
Functional Interpretation and Pathway Analysis
(metabolite set enrichment and metabolic pathway analysis)
Biological Interpretation

FIGURE 2.1: Metabolomic Workflow for Metabolomic Studies from Zhou et al [21].

peaks in the data to actual metabolites in the data. Another type of peak annotation step is the adduct or fragment identification depending on the polarisation used in the LC/MS instrument (see Section 2.2.5).

- Last steps of the workflow might include pathway and metabolic network analysis [1, 3]. Functional analysis is performed by searching for overrepresented and underrepresented labels from known biological data. In most metabolomic studies, it is common to perform additional steps based on metabolite verification and quantification of the candidate metabolites (e.g. using a tandem MS for better metabolite identification) [3].

2.2 LC/MS Data Processing

The first data preprocessing steps after LC/MS data acquisition are signal filtering and peak picking, also known as peak detection (Figure 2.1). The output of LC/MS instruments is a sparse 3D signal whose dimensions are intensity, m/z and the rt for each detected feature (peak mass). Figure 2.2 depicts an LC/MS profiling for a urine sample. The figure shows that the LC/MS signals are intrinsically sparse, as the majority of the 2D plane formed by m/z and rt does not contain any peaks. Furthermore, Figure 2.2 also shows that LC/MS signals are highly anisotropic. If we consider the integral of the whole signal in Figure 2.2 on the plane I-rt, the resultant signal has broader peaks and almost all the rt range has an intensity signal. On the other hand, the integral of the signal on the I-m/z plane gives narrow peaks and gaps in the m/z range without any intensity signal. This anisotropy is also depicted in the TIC shown in Figure 2.3 compared to the masses of Figure 2.4

Peak mass signals can be recorded in either profile or centroid mode. Centroid mode records peak masses as a set of discrete values in which the peak masses have no width (Figure 2.4 shows a centroid mass recording). On the other hand, in profile recording the peak mass signal is continuous. Centroid mode recording produces smaller file sizes as only the peak centroid is saved. There are algorithms and tools provided by the commercial vendors of analytical devices to switch the signal recording from profile mode to centroid mode in a process called centroidisation [22].

2.2.1 Peak Detection

Peak Detection (also called Peak Picking) is a complex mathematical step that usually involve the use of complex peak detection algorithms and filtering methods [22, 24, 25, 26]. The main objective of this step is to detect signal peaks which will ultimately be related to metabolites. The anisotropy of the LC/MS signals is exploited by many peak picking algorithms that detect the peaks taking into account that rt and m/z dimensions are different in terms of peak behaviour and peak resolution (m/z measures are more precise than rt measures) [24]. Two of the most used methods for peak detection are Matched

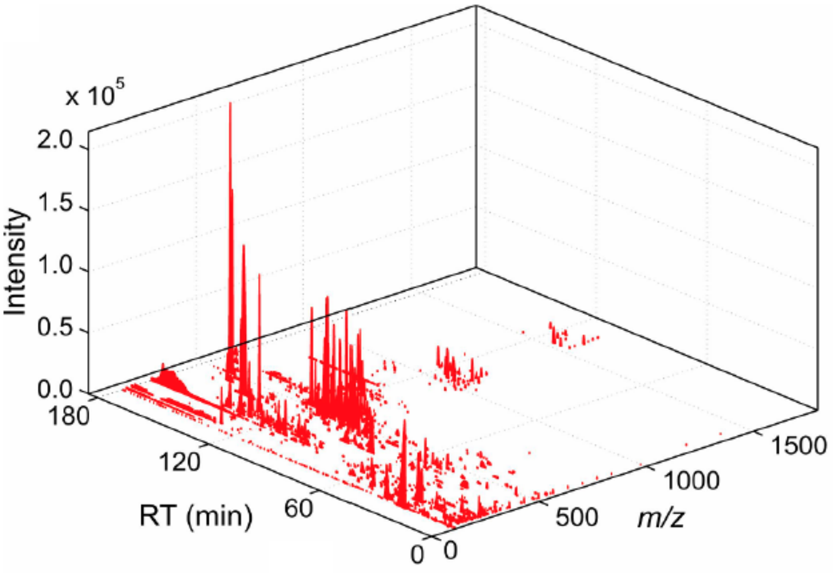


Figure 2.2: LC/MS profile for a urine sample from Guan et al [23]. The height depicts the intensity in arbitrary units, horizontal axis is the m/z for the piece whereas the in-depth axis is the retention time in seconds

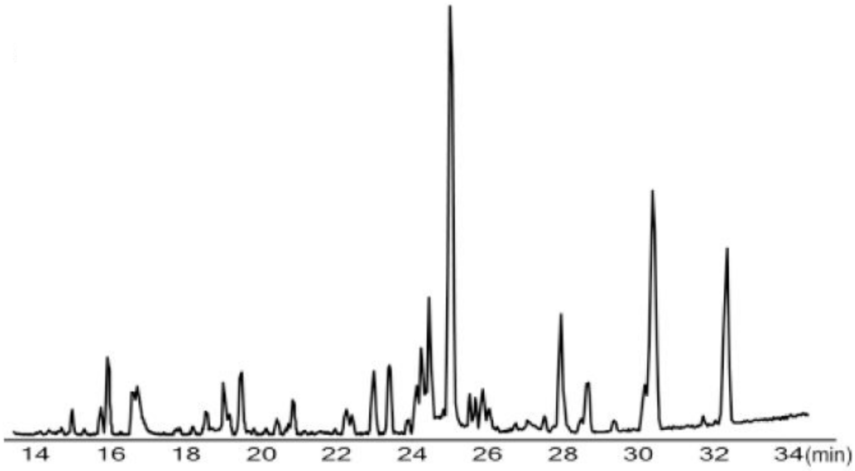


Figure 2.3: Chromatographic profile registered in a LC/MS device from Zhu et al [27]. Horizontal axis is the retention time in minutes whereas the vertical axis is the intensity measured in arbitrary units.

Filter [24, 26] and centWave [22].

2.2.1.1 Matched Filter

Matched Filter method uses a two-fold differentiation approach and it has been one of the most widely used algorithms to filter LC/MS signals [26]. This method is implemented in the R package XCMS [24]. It makes the hypothesis that the chromatographic peaks can

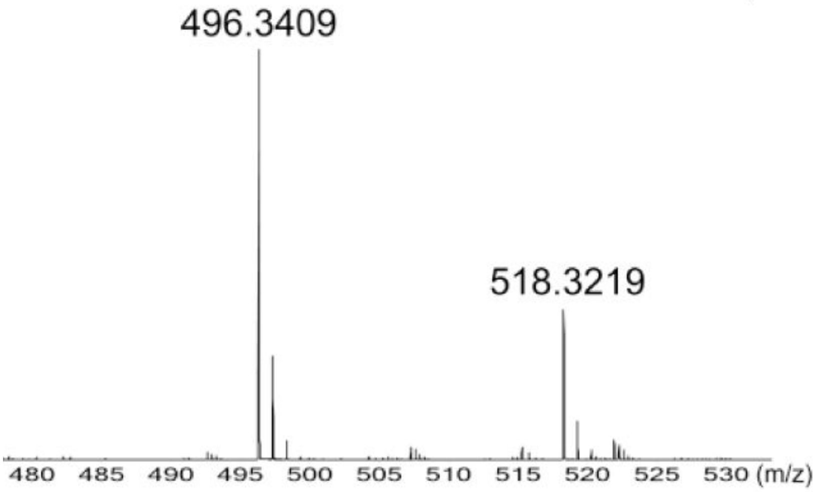


FIGURE 2.4: Sample centroid spectrum profile registered from a urine sample in a LC/MS device from Zhu et al [27]. Horizontal axis is the mass charge ratio in Dalton whereas the vertical axis is the intensity measured in arbitrary units

be approximated by a certain function. It is usually assumed that the peak shapes are similar to a gaussian function. The next step is to perform a slicing of the signal known as binning in the m/z domain, and superimpose the second derivative of the gaussian function on the rt dimension of the signal to obtain a sharper chromatographic shape of the peaks [26]. XCMS software suggests taking a signal-to-noise cutoff value to finally detect the peaks once the filter has been applied [24]. This software tool also proposes to take the mean of the unfiltered data to determine the noise threshold value. The Matched Filter method shows some drawbacks related to having to manually choose the binning value, and to the dependence of the method with the gaussian function parameters [22]. If the binning value is too small, the m/z slices are so thin that the same chromatographic peaks are found in many slices and they are not detected as peaks. On the other hand, if the binning value is so large, small chromatographic peaks will not be detected as they will be added to other chromatographic peaks [22, 24]. The issue with the gaussian parameters is related to the different shapes that chromatographic peaks show in biological samples. Depending on the input parameters, the Matched Filter method can lead to detect more or less peaks than exist in the chromatogram due to the different shape between the input gaussian and the real chromatographic peaks.[22].

2.2.1.2 centWave

Another different approach is the centWave method which was specially design to avoid the binning problems associated to the Matched Filter algorithm. The centWave algorithm is based on detecting the so-called regions of interest (ROIs) in the centroided m/z domain, combined with a Continuous Wavelet Transform (CWT) approach for chromatographic peak resolution [22]. Given a mass accuracy value, masses are classified to ROIs depending on their mass value. Regions with an amount of centroids below a user-defined threshold are removed. centWave method proposes the use of CWT to replace the second derivative of the Gaussian in the filtering as it correctly detects chromatographic peaks of different width [22]. The CWT is applied to the extracted ion chromatogram and the local maxima of the coefficients are used to detect the chromatographic peak of the ROI [22]. Chronologically, the Matched Filter method was developed earlier than the centWave. Nowadays it is recommended to use the centWave method instead of the Matched Filter [28].

In a similar way than in the Matched Filter method, the centWave algorithm is also implemented in the XCMS package as another peak detection method.

2.2.2 Peak alignment

In a metabolomic study many biological samples are collected and analysed. A critical step in the metabolomic workflow is the a peak alignment stage. Computational data processing workflows typically include a transformation of the original raw data into a data matrix containing information of the peak masses and retention time values for all the samples [29]. This mathematical procedure requires the samples to be aligned in the retention time dimension to ensure that the retention time axis is the same for all the samples. The fact that the sample chromatograms may show different peak behaviour can have different reasons: replacement of the chromatographic column, changes in the mobile phase, drift in the instrument etc. [30]. Because of this wide variety of causes, the alignment procedure should consider possible unwanted non-linear effects in the retention time [31]. Figure 2.5 shows how the chromatograms taken from consecutive samples may be unaligned. The objective of the alignment algorithms is to correct these effects

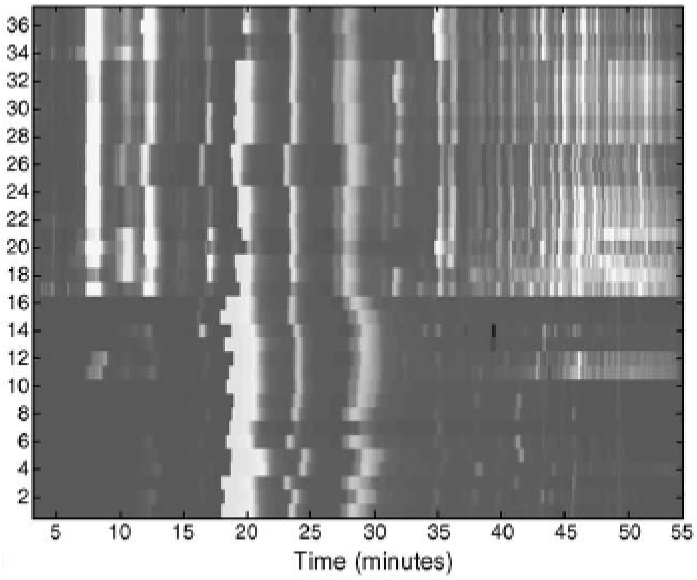


FIGURE 2.5: Group of unaligned LC/MS chromatograms the drift caused by an LC/MS instrument, from van Nederkassel et al [32]. Each row corresponds to a sample and the horizontal axis is the retention time. The lighter the colour, the higher the intensity of the signal.

and to get the same chromatographic peaks from all the samples as much aligned as possible. Ideally, after performing the peak alignment, the peaks of the resulting mean TIC chromatogram would have higher and sharper peaks thus improving the chromatographic behaviour of the samples as the uncertainty of the peaks would be lower.

The need for peak alignment in chromatography has been known for a long time and many alignment algorithms has been proposed [30, 31, 33, 34, 35, 36, 37]. The alignment algorithms can be divided into two main categories: those that require to run a peak mass detection stage prior to perform the alignment [38], and those algorithms that are applied directly on the chromatogram profiles without using the m/z values [30, 37]. Among the most used methods for both types of algorithms there is the Locally Weighted Scatterplot Smoothing (LOESS) that uses peak masses and the Parametric Time Warping (PTW) that need not to detect the peak masses to run the chromatographic alignment.

2.2.2.1 LOESS

The LOESS method uses a warping approach using groups of well-behaved peak mass groups to adjust local polynomials and align the chromatograms [24]. To find the regression function, it is only used those points close to a certain point x where we want to evaluate the function [39]. The weight functions used in the LOESS algorithm are usually either quadratic or cubic polynomials. As a consequence, the LOESS algorithm performs the alignment of the chromatograms by performing a piecewise local adjustment of second- or third-order polynomials where the retention time intervals are found by detecting the groups of well-behaved peak mass groups. The LOESS method is also implemented in the XCMS package. The version implemented in the XCMS package uses high density peak regions considering all the samples to define the well-behaved peaks. When a group of samples show peak masses in a close region, it is likely that that peak is not much unaligned an is chosen as a well-behaved peak.

Despite being widely used in recent times and being implemented in a package like XCMS, LOESS method present the drawbacks of the warping methods. Being a piecewise warping algorithm, the pieces of the chromatograms are stretched or compressed following the optimisation of an objective function. This warping procedure might cause artefacts in the aligned chromatograms or produce models with overfitting depending on the binning parameter.

2.2.3 Amplitude Normalisation

The next step in the LC/MS signal processing is the amplitude or intensity normalisation. This stage corrects the overall changes in the intensity of the samples when different batches were used and/or because some samples were injected in different days and the experimental device shows a drift in the intensity measurements. Figure 2.6 depicts a PCA score plot of the LC/MS metabolomic data described in Wang et al [40]. In this plot it is shown that batch and injection order effects have an strong influence on the structure of the data by being and important source of variance in the data. In particular, the second PC of Figure 2.6 contains a combination of batch and injection order effects.

Batch effects cause the variables to behave differently depending on which sample is being considered. Because the sources of this misleading variance are many, the noise structure of LC/MS data have heteroscedastic behaviour, which means that signals with higher intensity values show higher variance [41]. As a consequence applying machine

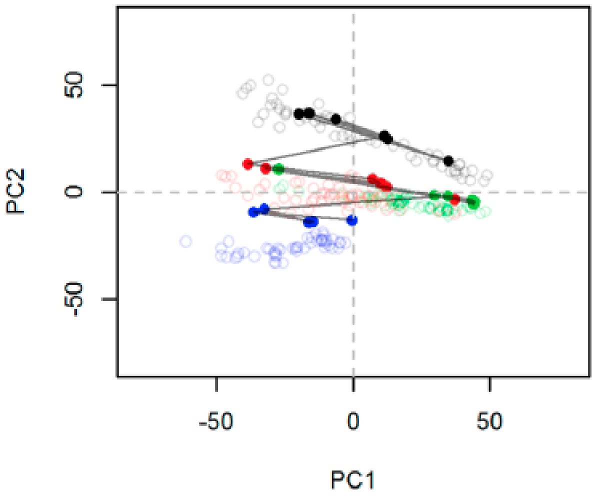


Figure 2.6: PCA Scoreplot of the LC/MS data set from Wang et al [40]. Solid and open circles refer to WC and study samples respectively. Different colours are used to label different batches. Solid lines show the injection order.

learning or multivariate methods such as PCA on the raw data may give wrong information [41] although these statistical methods has been applied in metabolomic studies for long time without normalising the intensity of the signal [18].

In recent years, many amplitude normalisation methods have been developed using different approaches to the same problem [28, 40, 41, 42, 43, 44]. A number of them rely on the use of internal standards [28, 42, 44] to perform the intensity normalisation. The idea is that internal standards only undergo variations caused by the batch effects or time biases. The reason is that the internal standards are always the same samples so they do not contain biological variability (i.e. they will only show technical variability). As a consequence, if the internal standard for a certain sample shows a lower intensity value than a sample of the previous batch, it is expected that all the intensity levels for the samples of that batch (or that have been analysed closely in time to the QC) are going to show overall lower intensity values as well. These internal standards are usually called Quality Controls (QC) and typically can made of a pool of aliquots from all or a subset of the samples being analysed to ensure that they contain similar metabolic information[28]. Other QC strategies include using spikes or water samples as a control of the performance of the analytical instrument [18].

Among the methods used the more relevant are:

- LOESS method is used to correct the between-batch variations [28]. This method uses the QC peak expression to adjust a LOESS curve. As the QCs are injected in each batch, the LOESS adjustment measures the drift of the intensity across the time [28]. This curve is then used to correct the intensity values of the peaks for every sample.

- Veselkov et al. describes and compares the performance of four typical approaches to intensity normalisation with and without a variance stabilisation transformation that compensates the heteroscedastic noise of the LC/MS data [41]. The four tested approaches are LOESS, Median fold change (forces the median of log fold changes of peak intensities to be 0), Total Intensity (all samples are forced to have the same total intensity) and Quantile (forces the peak intensity distribution to be the same in all samples) normalisations. Their conclusions show that the best performance is reached by performing the variance stabilisation transformation and using the median fold change normalisation [41].

- A method called Batch Normaliser based in a linear regression model to normalise the intensity is proposed by Wang et al. [40]. The underlying idea of this model is to fit a linear model using the QC samples to capture the effects of both the batch and the injection order. Once the model is fit, the parameters of the model are used to correct the intensity of the rest of the samples [40]. In this paper, the Batch Normaliser method is compared to other standard methods like the quantile method. The results show that the Batch Normaliser outperforms all the methods tested.

2.2.4 Multivariate and Statistical Data Analysis

Once the features in the samples are detected (i.e. the m/z and rt for each peak), their chromatograms aligned and the intensity normalised, the next data processing stage is to perform the analysis of the data using Statistical or Machine Learning approaches. At this point the data takes form of a matrix with a row for each detected peak and a column for each analysed sample plus two more columns containing the m/z and rt data for each feature. The data of the matrix is the intensity of the peak for that sample. There are usually two different ways of computing this intensity which are the maximum

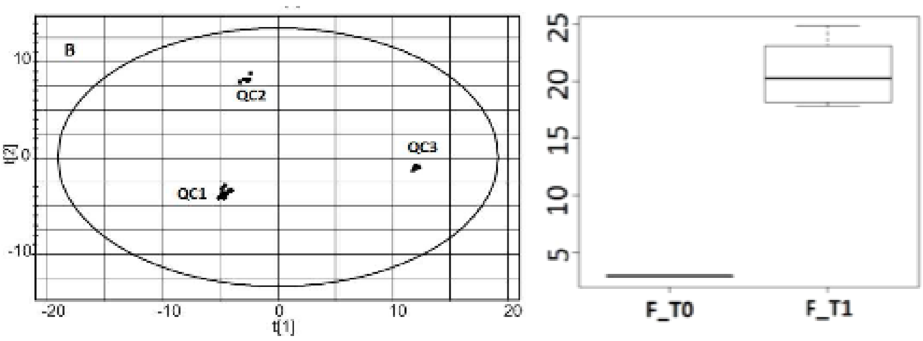


Figure 2.7: Left plot shows an scoreplot of QC samples analysed using an LC/MS instrument. Right plot depicts a box plot of a single statistically significant feature of a dataset containing two classes (F_T0 and F_T1). Both plots are from Tulipani et al. [45]

of the chromatographic peak or the area of the chromatographic peak [24].

Three main types of mathematical approaches might applied on this data matrix:

- A range of statistical tests such as parametric tests (e.g. Student's t-test or ANOVA) or non-parametric tests (e.g. Mann-Whitney or Welch tests) are normally applied on the data in order to obtain the most statistically significant features that separate the classes of the data [45]. As these tests are applied on each feature individually, the number of computed tests is high and performing a multiple test correction like Bonferroni or False Discovery Rate (FDR) is recommended to avoid false positives [46]. Plotting Boxplots (see the one shown in the right plot of Figure 2.7) is also a classical approach used in metabolomic studies to evaluate the differences of statistically significant features between the classes involved in the data. For example the box plot of Figure 2.7 shows that the plotted feature is statistically significant because the metabolite is found for samples of class F_T1 but it is not found for both class F_T0.

- The most common non-supervised technique applied on metabolomic LC/MS data is a PCA score plot of the data for visualisation purposes [47, 48]. Plotting the data in a PCA score plot allows a dimensionality reduction of the data and it revels the main sources of variance in the data. Left plot of Figure 2.7 depicts a 2D PCA score plot with three different types of QC samples. In this Figure it can be seen that the three classes are clearly separable in the PC1/PC2 plane as they appear depicted wide apart in the score plot. Another mathematical approach

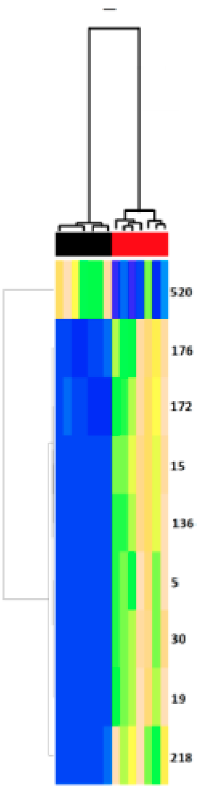


FIGURE 2.8: Heat map showing the unsupervised clustering of samples at the top (colours refer to real sample classification) and of the statistically significant features at the left side of the plot from Tulipani et al [45].

regarding non-supervised techniques are the unsupervised classifiers. In LC/MS metabolomic studies, unsupervised classifiers usually take the form of heat map plots in which are performed unsupervised clusterings of features and samples. Figure 2.8 depicts a heat map of samples having of two classes. The actual sample labelling is shown at the column colour at the top of the plot. The samples shown in the figure correspond to the statistically significant features found after performing a Student's test on each feature. The most typical distances used for these hierarchical clustering are either the euclidean or the correlation distances. As it is clear from Figure 2.8, heatmap figures are useful for analysing the results of metabolomic studies as they allow an easy correspondence of every significant feature to a class.

- Supervised classifiers such as PLSDA and SVM are normally used in an LC/MS metabolomic context to evaluate the class-related information of the features [47, 50, 49]. Repeated random sub-sampling cross-validation stages are applied using SVM and PLSDA to evaluate the statistical predictive power of the LC/MS

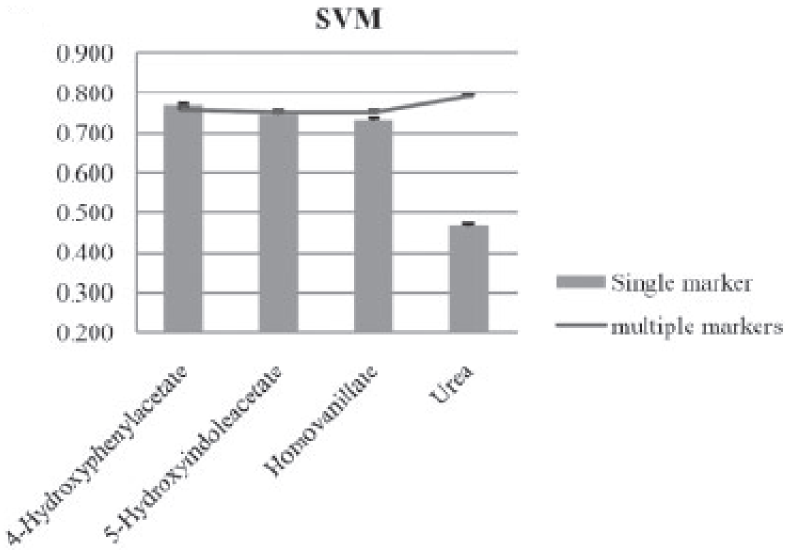


FIGURE 2.9: Boxplot showing the classification ratio in a 50-fold random sub-sampling cross-validation SVM classification stage from Nam et al [49]. The different boxes refer to different biomarkers used as variables to predict the samples.

metabolomic data. Figure 2.9 shows the classification ratio of performing a cross-validation stage in a metabolomic study using four different biomarkers in SVM. Random forests can be used either as a supervised or unsupervised classifier and they are based on constructing sets of classification trees. The use of this technique in the LC/MS metabolomic context has increased in recent years [51].

2.2.5 Peak Annotation

Peak Annotation is a stage of the metabolomic data that has as an objective, to make the biological interpretation of the results easier by gaining some chemical and biological insight. In general, in LC/MS metabolomic data, the ionisation of a single source metabolite might give a set of features. To make easier the metabolite identification, there are some approaches that find the peaks coming from the same metabolite and relate them through possible chemical transformations that the molecules might have undergone in the MS. Depending on the polarisation mode used when analysing the samples, the peaks might form agglomerates with some other atoms or molecules, resulting in a overall charged positive (if the polarisation mode of the ESI was set to positive) or negative (if the polarisation mode of the ESI was set to negative) feature. The positively charged agglomerates are called adducts (for example if a sodium atom was attached a piece of metabolite having neutral electrical charge) whereas the negative are called fragments (for example if a hydrogen atom was removed from a piece of metabolite having neutral electrical charge). A typical approach to find these agglomerated peaks is to define a retention time allowance window and a correlation threshold value [52, 53, 54]. Another agglomeration that the molecule pieces might undergo is the neutral loss or biotransformations, in which a neutral molecule is lost or attached to a piece of the ionised molecule (for example a loss of glucose from a a piece of metabolite having neutral electrical charge) [52].

Database mining can also be considered a type of peak annotation. Databases usually contain a relation between the metabolite and a primary mass i.e. a characteristic piece of the metabolite fragmentation or ionisation. To perform the metabolite identification step, the typical approach is using an allowance mass window to query a metabolite database for example METLIN [55] or Human Metabolome DataBase (HMDB) [21, 27, 55, 56]. The main problem of this strategy is that, in general, the allowance window method is not restrictive enough and there are multiple possible hits for each query mass. Several approaches have been proposed to tackle this issue. A first approach to choose between several metabolite hit candidates, proposes to introduce a gaussian model in order to generate a probabilistic measure of the possible candidates [57]. Another approach is to used the so-called ”In silico fragmentation”. This strategy is based on performing the opposite procedure than the regular metabolite identification. The idea is to simulate the fragmentation pattern of the metabolites in the database following chemical laws and the check which of the simulated peaks appear in the real data [57, 58]. As the final step of this procedure, the candidate metabolites are then ranked according to the number of matched peaks between the real and the simulated peaks.

2.3 Computational Tools

Analysing biological samples using an LC/MS in a metabolomic context produces high throughput data. Therefore, due to the sizes of the data involved in a typical metabolomic analysis, using software tools is a mandatory step in the data processing workflow. Many commercial brands producing their own LC/MS instruments, deliver their own in-house software tools with the analytical instrument. The softwares from Analyst `http://`

www.absciex.com/products/software/analyst-software or the software MarkerLynx from Waters http://www.waters.com/waters/en_US/MarkerLynx-/nav.htm?cid=513801&locale=en_US perform the first signal processing steps. There also exist computational tools from companies focused on the statistical analysis of high-throughput data produced by such devices. These software tools are usually applied as black boxes with limited user intervention. Among these commercial computational tools, one of the most used softwares is SIMCA-P+ from Umetrics (http://www.umetrics.com/simca). Free tools usually developed in R [24], Python [59] or Java [60] to cite some, are also available and they are also widely used. R is a free software programming language specially focused on statistics with which have been developed many tools for analysing metabolomic data [61]. Besides R, in recent times it has been developed a great number of computational tools developed in many other programming languages like Python, Java or web tools. In this thesis, only free tools will be considered. Basically, the available free tools fall within two different classes. There are tools which are specific of certain processing stages of the metabolomic data analysis workflow, like for example the CAMERA package which is focused on peak annotation (i.e. adduct, fragment and neutral loss annotation stages)[52]. Most of R packages fall within this category. Nevertheless more tools are needed to perform a complete complex metabolomic analysis when these kind of tools are used. On the other hand, there also exist free online tools that allow to perform a complete end-to-end metabolomic analysis such as Metaboanalyst [62]. However, this kind of tools are usually implemented as a web service and they are not as flexible as the other set of tools due to its limited user customisation capabilities. Additionally, due to their web-based nature, it is hard to implement an automatic procedure to fully analyse the data with little user intervention. Table 2.1 contains a summary of the main available computational tools with their main stages of application in the metabolomic workflow.

2.3.1 R packages and other tools

R is perhaps the programming language in which the number of metabolomic packages is growing the most. Because of the package-based nature of the R language, newer R packages are normally supported on other older R packages. Therefore, the user can develop complex R packages with ease. Because of this feature of the R language, the library of R packages which are focused on processing metabolomic data is increasing in number and complexity. Normally, the available R packages to perform metabolomic data analysis are focused on the first stages of the metabolomic workflow [24, 52, 53].

Among the available R packages in the metabolomic context, XCMS is one of the most used packages. The main tasks of this package are to perform the peak detection and peak alignment, as well as basic statistical processing in metabolomic data analysis. The package is designed to analyse signals from both GC/MS and LC/MS platforms.

As it is explained in Sections 2.2.1.1 and 2.2.1.2, XCMS allows the use of both Matched Filter and centWave methods when it comes to filtering the raw signals and perform the peak detection stage. Before moving on to the peak alignment stage, XCMS groups the peaks across the samples. As it is said in section 2.2.2.1, Besides the LOESS method (explained in section 2.2.2.1) to align the chromatograms, the XCMS package also contains other alignment methods such as Obiwarp [64] or the piece-wise linear alignment methods.

Besides the peak detection and peak alignment stages, XCMS also allows to perform basic univariate statistical tests such as ANOVA test (when there are more than two classes in the data) or a Student's t-test (when there are two clases on the data) for each feature. The results of these statistical tests are retrieved as p-values in the output table of the package. Moreover, XCMS is able to perform and online query the Metlin database [55]. It was also published a protocol englobing the use of XCMS to perform the peak detection stage and the Metlin Database to identify the statistically significant features [27].

Another used package is CAMERA. This R package is designed to perform the peak annotation of adducts, fragments and neutral losses. The package requires the previous use of the package XCMS to detect the peaks before computing the peak annotation. It includes tables of adducts, fragments and neutral losses but it accepts a user-defined table as an input parameter to allow customisation of the peak annotations.

As it is explained in section 2.2.5, one of the methods to perform the peak annotation is to group the peaks using an allowance retention time window and a peak correlation threshold strategy. The Retention time allowance window is defined according to the shape of the chromatographic peak and a user input parameter. According to this procedure, the peaks falling inside the retention time allowance window are grouped as

possible candidates to come from the same source metabolite. In a second step, the correlation between the peaks of each group and across the samples is computed. If the correlation value is lower than a certain user-defined correlation threshold value, the group is split in groups where the peak correlation in the peak groups is equal or greater than the threshold.
In a similar way than the XCMS package, the output of the CAMERA analysis is a table. This table includes the XCMS output the peak annotation and also contains the peak groups obtained in the CAMERA workflow.

metaXCMS is another R package that proposes to perform an extra processing step besides the XCMS statistical tests [65, 66]. The package is based on finding significant features between different pairwise experiments, giving as an the different combinations of statistical significant features in a Venn Diagram.

mzMine is a toolset implemented in Java and designed to analyse metabolomic signals [67]. The original mzMine launched in 2006 has been updated and improved recently to its second version mzMine 2.0 [60]. It performs the peak detection stage and basic statistical processing while the tool is designed putting emphasis on visualisation tools.

2.3.2 Online Computational Tools

Online Computational tools to perform metabolomic data analysis are a recent development compared to many software packages. Among the web tools available, MetaboAnalyst [62, 68] and XCMS online [69] are two of the most widely used online tools for metabolomic data processing.

The first version of the MetaboAnalyst web server was launched in 2009 [68] but it has been updated recently to its second version [62]. Overall MetaboAnalyst is one of the most complete computational tools available for performing Metabolomic data analysis. It allows for and end-to-end processing workflow that analyse GC/MS, LC/MS and NMR metabolomic data. Among its processing steps, there are the data processing (including peak detection and noise filtering), data normalisation, statistical analysis (including time-series analysis) and pathway/enrichment analysis [62]. A wide array of statistical tools are available in the MetaboAnalyst 2.0 classified in traditional chemometric analysis including multivariate (like PCA, PLSDA) and univariate (ANOVA, t-tests) approaches or more advanced machine learning apporaches such as SVM or Random Forest methods. It also includes several methods to perform unsupervised clustering analysis like K-means or self-organising maps (SOM). Furthermore, MetaboAnalyst 2.0 has enhanced graphical interface compared to its first version making it more user-friendly. Before launching the second version of the MetaboAnalyst software tool, it was published a protocol to perform and end-to-end metabolomic analysis and obtain biological interpretation from raw data [70].

XCMS Online is another recently launched tool to perform end-to-end untargeted analysis of metabolomic data. Its objective is to be an easy-to-use computational tool to analyse LC/MS data in a user-friendly environment [69]. XCMS Online uses the R package XCMS to perform the peak detection and peak alignment stage. The software automatically identifies the statistically significant metabolites using the Metlin database [69].

Table 2.1: Summary of the main available computational tools.

Tool	Type	Description and Functionality
XCMS	R Package	Peak Detection, Statistical Test (Student's T-test/ANOVA) and Metabolite Identification through Metlin Database query
CAMERA	R Package	Peak Annotation of Adducts/Fragments and Neutral Losses. It complements the XCMS package.
metaXCMS	R Package	Statistical Analysis. It complements the XCMS package by adding pairwise comparison of the significant features for pairs of experiments.
mzMine	Java tool	Peak detection and basic statistical processing. Its main feature is the great range of visualisation tools.
metaboAnalyst	R Online Tool	The tool covers all the workflow, including peak detection, statistical analysis, peak annotation and pathway enrichment.
XCMS Online	R Online Tool	Peak detection, statistical analysis and Metabolite Identification through Metlin Database.

Bibliography

[1] Lu Xin, Zhao Xinjie, Bai Changmin, Zhao Chunxia, Lu Guo, and Xu Guowang. Lc-ms-based metabonomics analysis. *Journal of Chromatography B*, 866:64–76, 2007.

[2] Zhang Aihua, Sun Hui, Wang Ping, Han Ying, and Wang Hijun. Modern analytical techniques in metabolomics analysis. *Analyst*, 137:293–300, 2012.

[3] Zhou Bin, Xiao Jun Feng, Tuli Leepika, and Ressom Habtom W. Lc-ms-based metabolomics. *Molecular BioSystems*, 8:470–481, 2012.

[4] G. Theodoridis, H.G. Gika, and I.D. Wilson. Mass spectrometry-based holistic analytical approaches for metabolite profiling in systems biology studies. *Mass Spectrometry Reviews*, 30:884–906, 2011.

[5] E.J. Want, B.F. Cravatt, and G Siuzdak. The future of liquid chromatographymass spectrometry (lc-ms) in metabolic profiling and metabolomic studies for biomarker discovery. *Biomark Med*, 1(1):159–185, 2007.

[6] Aurélie Roux, Dominique Lison, Christophe Junot, and Jean-François Heilier. Applications of liquid chromatography coupled to mass spectrometry-based metabolomics in clinical chemistry and toxicology: A review. *Clinical Biochemistry*, 44:119–135, 2011.

[7] J. K. Nicholson, J. C. Lindon, and E. Holmes. 'Metabonomics': understanding the metabolic responses of living systems to pathophysiological stimuli via multivariate statistical analysis of biological NMR spectroscopic data. *Xenobiotica*, 29(11):1181–1189, November 1999. ISSN 0049-8254.

[8] J. K. Nicholson, J. Connelly, J. C. Lindon, and E. Holmes. Metabonomics: a platform for studying drug toxicity and gene function. *Nat Rev Drug Discov*, 1(2): 153–161, February 2002. ISSN 1474-1776.

[9] J. K. Nicholson and Wilson I.D. Understanding 'Global' Systems Biology: Metabonomics and the Continuum of Metabolism. *Nat Rev Drug Discov*, 2(2):668–676, 2003.

[10] Nawaporn Vinayavekhin and Alan Saghatelian. Untargeted metabolomics. *Current protocols in molecular biology edited by Frederick M Ausubel et al*, Chapter 30 (April):Unit 30.1.1–24, 2010.

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

[12] Vladimir Shulaev. Metabolomics technology and bioinformatics. *Briefings in Bioinformatics*, 7(2):128–139, 2006. doi: 10.1093/bib/bbl012.

[13] D Lee Roberts, Amanda L. Souza, Robert E Gerszten, and Clary B Clish. Targeted metabolomics. *Current protocols in molecular biology edited by Frederick M Ausubel et al*, Chapter 30(April):Unit 30.2, 2012.

[14] Monica Chagoyen and Florencio Pazos. Tools for the functional interpretation of metabolomic experiments. *Briefings in Bioinformatics*, 2012. doi: 10.1093/bib/bbs055.

[15] Oscar Yanes, Ralf Tautenhahn, Gary J. Patti, and G Siuzdak. Expanding coverage of the metabolome for global metabolite profiling. *Anal Chem*, 15(83), 2011.

[16] Maria Vinaixa, Sara Samino, Isabel Saez, Jordi Duran, Joan J Guinovart, and Oscar Yanes. A guideline to univariate statistical analysis for lc/ms-based untargeted metabolomics-derived data. *Metabolites*, 2(4):775–795, 2012. URL http://www.mdpi.com/2218-1989/2/4/775/.

[17] Johan Trygg, Elaine Holmes, and Torbjörn Lundstedt. Chemometrics in metabonomics. *Journal of proteome research*, 6(2):469–79, February 2007. ISSN 1535-3893. doi: 10.1021/pr060594q.

[18] Sara Tulipani, Rafael Llorach, Olga Jáuregui, Patricia López-Uriarte, Mar Garcia-Aloy, Mònica Bullo, Jordi Salas-Salvadó, and Cristina Andrés-Lacueva. Metabolomics Unveils Urinary Changes in Subjects with Metabolic Syndrome following 12-Week Nut Consumption. *Journal of Proteome Research*, 2011. ISSN 15353907. doi: 10.1021/pr200514h.

[19] Katherine Hollywood, Daniel R. Brison, and Royston Goodacre. Metabolomics: Current technologies and future trends. *PROTEOMICS*, 6(17):4716–4723, 2006. ISSN 1615-9861. doi: 10.1002/pmic.200600106. URL http://dx.doi.org/10.1002/pmic.200600106.

[20] Gary J. Patti, Ralf Tautenhahn, Duane Rinehart, Kevin Cho, Leah P. Shriver, Marianne Manchester, Igor Nikolskiy, Caroline H. Johnson, Nathaniel G. Mahieu, and Gary Siuzdak. A view from above: Cloud plots to visualize global metabolomic data. *Analytical Chemistry*, 85(2):798–804, 2013. doi: 10.1021/ac3029745.

[21] Bin Zhou, Jinlian Wang, and Habtom W Ressom. Metabosearch: Tool for mass-based metabolite identification using multiple databases. *PLoS ONE*, 7(6):e40096, 2012. URL http://www.pubmedcentral.nih.gov/articlerender.fcgi?artid=3387018&tool=pmcentrez&rendertype=abstract.

[22] Ralf Tautenhahn, Christoph Böttcher, and Steffen Neumann. Highly sensitive feature detection for high resolution lc/ms. *BMC Bioinformatics*, 9(1):504, 2008.

[23] Wei Guan, Manshui Zhou, Christina Y Hampton, Benedict B Benigno, L Deette Walker, Alexander Gray, John F McDonald, and Facundo M Fernández. Ovarian cancer detection from metabolomic liquid chromatography/mass spectrometry data by support vector machines. *BMC Bioinformatics*, 10(August 2009):259, 2009. URL http://www.pubmedcentral.nih.gov/articlerender.fcgi?artid=2741455&tool=pmcentrez&rendertype=abstract.

[24] Colin A Smith, Elizabeth J Want, Grace O?Maille, Ruben Abagyan, and Gary Siuzdak. Xcms: processing mass spectrometry data for metabolite profiling using nonlinear peak alignment, matching, and identification. *Analytical Chemistry*, 78 (3):779–787, 2006.

[25] Mikko Katajamaa and Matej Ore?i? Processing methods for differential analysis of lc/ms profile data. *BMC Bioinformatics*, 6(1):179, 2005.

[26] Rolf Danielsson, Dan Bylund, and Karin E. Markides. Matched filtering with background supression for improved quality of base peak chromatograms and mass spectra in liquid chromatography-mass spectrometry. *Analytica Chimica Acta*, 454(2): 167–184, November 2002. ISSN 01697439.

[27] Zheng-Jiang Zhu, Andrew W Schultz, Junhua Wang, Caroline H Johnson, Steven M Yannone, Gary J Patti, and Gary Siuzdak. Liquid chromatography quadrupole time-of-flight mass spectrometry characterization of metabolites guided by the metlin database. *Nature Protocols*, 8(3):451–460., 2013.

[28] Warwick B Dunn, David Broadhurst, Paul Begley, Eva Zelena, Sue Francis-McIntyre, Nadine Anderson, Marie Brown, Joshau D Knowles, Antony Halsall, John N Haselden, and et al. Procedures for large-scale metabolic profiling of serum and plasma using gas chromatography and liquid chromatography coupled to mass spectrometry. *Nature Protocols*, 6(7):1060–1083, 2011. URL http://www.ncbi.nlm.nih.gov/pubmed/21720319.

[29] Micha Daszykowski and Beata Walczak. Use and abuse of chemometrics in chromatography. *TrAC Trends in Analytical Chemistry*, 25(11):1081–1096, 2006.

[30] NPV Nielsen, JM Carstensen, and J Smedsgaard. Aligning of single and multiple wavelength chromatographic profiles for chemometric data analysis using correlation optimised warping. *J. Chromatogr. A*, 805(1-2):17–35, 1998.

[31] Katharina Podwojski, Arno Fritsch, Daniel C. Chamrad, Wolfgang Paul, Barbara Sitek, Kai Stühler, Petra Mutzel, Christian Stephan, Helmut E. Meyer, Wolfgang Urfer, Katja Ickstadt, and Jörg Rahnenführer. Retention time alignment algorithms for lc/ms data must consider non-linear shifts. *Bioinformatics*, 25(6):758–764, 2009.

[32] A M Van Nederkassel, M Daszykowski, P H C Eilers, and Y Vander Heyden. A comparison of three algorithms for chromatograms alignment. *Journal of Chromatography A*, 1118(2):199–210, 2006.

[33] H. Chen and Cs Horvágh. High-speed high-performance liquid chromatography of peptides and proteins. *Journal of chromatography A*, 705:3–30, 1995.

[34] Nils Hoffmann, Matthias Keck, Heiko Neuweger, Mathias Wilhelm, Petra Högy, Karsten Niehaus, and Jens Stoye. Combining peak- and chromatogram-based retention time alignment algorithms for multiple chromatographymass spectrometry datasets. *BMC Bioinformatics*, 13:214, 2012.

[35] Zhongqi Zhang. Retention time alignment of lc/ms data by a divide-and-conquer algorithm. *Journal of the American Society for Mass Spectrometry*, 23(4):764–772, 2012. URL http://www.ncbi.nlm.nih.gov/pubmed/22298290.

[36] Linda Kortz, Christin Helmshrodt, and Uta Ceglarek. Fast liquid chromatography combined with mass spectrometry for the analysis of metabolites and proteins in human body fluids. *Anal. Bioanal Chem*, 399:2635–2644, 2011.

[37] Paul H C Eilers. Parametric time warping. *Analytical Chemistry*, 76(2):404–411, 2004.

[38] K Magnus Aberg, Ralf J O Torgrip, Johan Kolmert, Ina Schuppe-Koistinen, and Johan Lindberg. Feature detection and alignment of hyphenated chromatographic-mass spectrometric data. extraction of pure ion chromatograms using kalman tracking. *Journal of Chromatography A*, 1192(1):139–146, 2008.

[39] W S Cleveland. Lowess: A program for smoothing scatterplots by robust locally weighted regression. *American Statistician*, 35(1):54, 1981. URL http://www.jstor.org/stable/2683591.

[40] San-Yuan Wang, Ching-Hua Kuo, and Yufeng J. Tseng. Batch normalizer: A fast total abundance regression calibration method to simultaneously adjust batch and injection order effects in liquid chromatography/time-of-flight mass spectrometry-based metabolomics data and comparison with current calibration methods. *Analytical Chemistry*, 85(2):1037–1046, 2013.

[41] Kirill A Veselkov, Lisa K Vingara, Perrine Masson, Steven L Robinette, Elizabeth Want, Jia V Li, Richard H Barton, Claire Boursier-Neyret, Bernard Walther, Timothy M Ebbels, and et al. Optimized preprocessing of ultra-performance liquid chromatography/mass spectrometry urinary metabolic profiles for improved information recovery. *ANALYTICAL CHEMISTRY*, 83(15):5864–5872, 2011.

[42] Frans M Van Der Kloet, Ivana Bobeldijk, Elwin R Verheij, and Renger H Jellema. Analytical error reduction using single point calibration for accurate and precise metabolomic phenotyping. *Journal of Proteome Research*, 8(11):5132–5141, 2009.

[43] Harmen H M Draisma, Theo H Reijmers, Frans Van Der Kloet, Ivana Bobeldijk-Pastorova, Elly Spies-Faber, Jack T W E Vogels, Jacqueline J Meulman, Dorret I Boomsma, Jan Van Der Greef, and Thomas Hankemeier. Equating, or correction for between-block effects with application to body fluid lc-ms and nmr metabolomics data sets. *Analytical Chemistry*, 82(3):1039–1046, 2010.

[44] Marko Sysi-Aho, Mikko Katajamaa, Laxman Yetukuri, and Matej Oresic. Normalization method for metabolomics data using optimal selection of multiple internal standards. *BMC BIOINFORMATICS*, 8, 2007.

[45] Sara Tulipani, Rafael Llorach, Mireia Urpi-Sarda, and Cristina Andres-Lacueva. Comparative analysis of sample preparation methods to handle the complexity of the blood fluid metabolome: When less is more. *Analytical Chemistry*, 85(1):341–348, 2013. doi: 10.1021/ac302919t. URL http://pubs.acs.org/doi/abs/10.1021/ac302919t.

[46] David I Broadhurst and Douglas B Kell. Statistical strategies for avoiding false discoveries in metabolomics and related experiments. *Metabolomics*, 2(4):171–196, 2006. URL http://www.springerlink.com/index/10.1007/s11306-006-0037-z.

[47] Masahiro Sugimoto, Masato Kawakami, Martin Robert, Tomoyoshi Soga, and Masaru Tomita. Bioinformatics tools for mass spectroscopy-based metabolomic data processing and analysis. *Current Bioinformatics*, 7(1):96–108, 2012. URL http://www.ingentaconnect.com/content/ben/cbio/2012/00000007/00000001/art00010.

[48] Chi Chen, Frank J Gonzalez, and Jeffrey R Idle. Lc-ms-based metabolomics in drug metabolism. *Drug Metabolism Reviews*, 39(2-3):581–597, 2007. URL http://www.pubmedcentral.nih.gov/articlerender.fcgi?artid=2140249&tool=pmcentrez&rendertype=abstract.

[49] Hojung Nam, Bong Chul Chung, Younghoon Kim, Kiyoung Lee, and Doheon Lee. Combining tissue transcriptomics and urine metabolomics for breast cancer biomarker identification. *Bioinformatics*, 25(23):3151–3157, 2009. URL http://www.ncbi.nlm.nih.gov/pubmed/19783829.

[50] Johan A Westerhuis, Huub C J Hoefsloot, Suzanne Smit, Daniel J Vis, Age K Smilde, Ewoud J J Velzen, John P M Duijnhoven, and Ferdi A Dorsten. Assessment of plsda cross validation. *Metabolomics*, 4(1):81–89, 2008. URL http://www.springerlink.com/index/10.1007/s11306-007-0099-6.

[51] Yue Liu, Zhanying Hong, Guangguo Tan, Xin Dong, Genjin Yang, Liang Zhao, Xiaofei Chen, Zhenyu Zhu, Ziyang Lou, Baohua Qian, Guoqing Zhang, and Yifeng Chai. Nmr and lc/ms-based global metabolomics to identify serum biomarkers differentiating hepatocellular carcinoma from liver cirrhosis. *International Journal*

of Cancer, 135(3):658–668, 2014. ISSN 1097-0215. doi: 10.1002/ijc.28706. URL http://dx.doi.org/10.1002/ijc.28706.

[52] C. Kuhl, R. Tautenhahn, C. Boettcher, T. R. Larson, and S. Neumann. Camera: an integrated strategy for compound spectra extraction and annotation of liquid chromatography/mass spectrometry data sets. *Analytical Chemistry*, 84:283–289, 2012. URL http://pubs.acs.org/doi/abs/10.1021/ac202450g.

[53] Arnald Alonso, Antonio Julià, Antoni Beltran, Maria Vinaixa, Marta Díaz, Lourdes Ibañez, Xavier Correig, and Sara Marsal. Astream: an r package for annotating lc/ms metabolomic data. *Bioinformatics*, 27(9):1339–1340, 2011. URL http://www.ncbi.nlm.nih.gov/pubmed/21414990.

[54] M. Brown, D.C. Wedge, R Goodacre, D.B. Kell, P.N. Baker, L.C. Kenny, M.A. Mamas, L. Neyses, and W. Dunn. Automated workflows for accurate mass-based putative metabolite identification in lc/ms-derived metabolomic datasets. *Bioinformatics*, 27(8):1108–1112, 2011.

[55] Colin A Smith, Grace O?Maille, Elizabeth J Want, Chuan Qin, Sunia A Trauger, Theodore R Brandon, Darlene E Custodio, Ruben Abagyan, and Gary Siuzdak. Metlin: a metabolite mass spectral database. *Therapeutic Drug Monitoring*, 27(6): 747–751, 2005.

[56] Ralf Tautenhahn, Kevin Cho, Winnie Uritboonthai, Zhengjiang Zhu, Gary J Patti, and Gary Siuzdak. An accelerated workflow for untargeted metabolomics using the metlin database. *Nature Biotechnology*, 30(9):826–828., 2013.

[57] John Draper, David P Enot, David Parker, Manfred Beckmann, Stuart Snowdon, Wanchang Lin, and Hassan Zubair. Metabolite signal identification in accurate mass metabolomics data with mzeddb, an interactive m/z annotation tool utilising predicted ionisation behaviour ”rules”. *BMC Bioinformatics*, 10(1):227, 2009. URL http://www.pubmedcentral.nih.gov/articlerender.fcgi?artid=2721842&tool=pmcentrez&rendertype=abstract.

[58] S. Wolf, S. Schmidt, M Müller-Hannemann, and S. Neumann. In silico fragmentation for computed assisted identification of metabolite mass spectra. *BMC Bioinformatics*, 11:148+, 2010.

[59] Sean O'Callaghan, David P. De Souza, Andrew Isaac, Qiao Wang, Luke Hodkinson, Moshe Olshansky, Tim Erwin, Bill Appelbe, Dedreia L. Tull, Ute Roessner, Antony Bacic, Malcolm J. McConville, and Vladimir A. Likic. Pyms: a python toolkit for processing of gas chromatographymass spectrometry (gc-ms) data. application and comparative study of selected tools. *BMC Bioinformatics*, 13:115, 2012.

[60] Tomás Pluskal, Sandra Castillo, Alejandro Villar-Briones, and Matej Oresic. Mzmine 2: modular framework for processing, visualizing, and analyzing mass spectrometry-based molecular profile data. *BMC Bioinformatics*, 11(1):395, 2010. URL http://www.pubmedcentral.nih.gov/articlerender.fcgi?artid=2918584&tool=pmcentrez&rendertype=abstract.

[61] R Development Core Team. *R: A Language and Environment for Statistical Computing*. R Foundation for Statistical Computing, Vienna, Austria, 2008. URL http://www.R-project.org. ISBN 3-900051-07-0.

[62] J Xia, R Mandal, I V Sinelnikov, D Broadhurst, and D S Wishart. Metaboanalyst 2.0–a comprehensive server for metabolomic data analysis. *Nucleic Acids Research*, 40(Web Server issue):1–7, 2012.

[63] Gabor Csardi and Tamas Nepusz. The igraph software package for complex network research. *InterJournal*, Complex Systems:1695, 2006. URL http://igraph.sf.net.

[64] John T Prince and Edward M Marcotte. Chromatographic alignment of esi-lc-ms proteomics data sets by ordered bijective interpolated warping. *Analytical Chemistry*, 78(17):6140–6152, 2006. URL http://www.ncbi.nlm.nih.gov/pubmed/16944896.

[65] Ralf Tautenhahn, Gary J Patti, Ewa Kalisiak, Takashi Miyamoto, Manuela Schmidt, Fang Yin Lo, Joshua McBee, Nitin S Baliga, and Gary Siuzdak. metaxcms: Second-order analysis of untargeted metabolomics data. *Analytical Chemistry*, 83(3):696–700, 2010. URL http://dx.doi.org/10.1021/ac102980g.

[66] Gary J Patti, Ralf Tautenhahn, and Gary Siuzdak. Meta-analysis of untargeted metabolomic data from multiple profiling experiments. *Nature Protocols*, 7(3):508–16, 2012. URL http://www.ncbi.nlm.nih.gov/pubmed/22343432.

[67] Mikko Katajamaa, Jarkko Miettinen, and Matej Oresic. Mzmine: toolbox for processing and visualization of mass spectrometry based molecular profile data. *Bioinformatics*, 22(5):634–6, 2006. URL http://www.ncbi.nlm.nih.gov/pubmed/16403790.

[68] Jianguo Xia, Nick Psychogios, Nelson Young, and David S Wishart. Metaboanalyst: a web server for metabolomic data analysis and interpretation. *Nucleic Acids Research*, 37(Web Server issue):W652–W660, 2009.

[69] Ralf Tautenhahn, Gary J Patti, Duane Rinehart, and Gary Siuzdak. Xcms online: A web-based platform to process untargeted metabolomic data. *Analytical Chemistry*, 2012.

[70] Jianguo Xia and David S Wishart. Web-based inference of biological patterns, functions and pathways from metabolomic data using metaboanalyst. *Nature Protocols*, 6(6):743–760, 2011. URL http://www.ncbi.nlm.nih.gov/pubmed/21637195.

Chapter 3

Goals

3.1 Main Objective

The main objective of the thesis is to improve the methodology of data analysis in metabolomics from end to end covering from the signal processing to statistical analysis and network analysis using machine learning methods. The data used in the thesis will be based on real human samples (especially urine) analysed through LC/MS devices with ESI.

3.2 Goals of the Project

- Improve the statistical prediction capabilities of the data using machine learning methods.
- Implement an easy-to-use R library to perform a flexible, robust and automatic analysis of LC/MS data.
- Develop a new algorithm to perform the peak alignment taking advantage of the experimental conditions particularities (7.5 min recording reverse-phase UPLC-MS).
- Implement the developed peak alignment algorithm as an independent R library.
- Solve the LC/MS normalisation issues by finding an appropriate normalisation algorithm or algorithms.

- Implement the required normalisation algorithms in an independent normalisation R package.
- The three developed R packages should be compatible. They have to be implemented to be applied sequentially as a workflow.

3.3 Expected Contributions

The most important expected contribution is a methodology based on a set of tools implemented in R that allow the user to analyse LC/MS metabolomic data. The tools design should allow flexibility in its use, meaning that proficient users might design or perform modification with ease. Additionally, the tools should be highly compatible between them as they should be used as black-boxes for most unexperienced users.

All the improvements performed in the metabolomic analysis workflow will be implemented as R packages (either additional packages or inside one of the three R packages mentioned in Section 3.2). Using these packages that are expected to contain improvements in the metabolomic workflow will provide the user with the necessary tools to perform an analysis of LC/MS metabolomic data.

Chapter 4

Peak aggregation as an innovative strategy for improving the predictive power of LC-MS metabolomic profiles

Note: This chapter is an exact copy of the paper:

Peak Aggregation as an Innovative Strategy for Improving the Predictive Power of LC-MS Metabolomic Profiles
Francesc Fernández-Albert, Rafael Llorach, Cristina Andres-Lacueva, Alexandre Perera-Lluna

Analytical Chemistry 2014 86 (5), 2320-2325

4.1 Abstract

The Liquid Chromatography-Mass Spectrometry (LC-MS)-based metabolomic datasets consist of different features including (de)protonated ions, fragments, adducts and isotopes that may show high correlation values related to a high level of collinearity. There have been described several sources of these high correlation patterns regarding metabolomic datasets. Among these sources, it should be highlighted the high level of correlation computed between features coming from the same metabolite. It is well known that soft ionisation methods (such as electrospray) produce several mass features from a particular compound (i.e. metabolite spectrum). Typically, the statistical methods used in metabolomics consider spectral peaks as variables. However it has been reported that a high collinearity between variables might be the responsible for high uncertainty values in the predictors of a regression. In this context, this technical note proposes a new strategy based on the application of the so-called peak aggregation methods (NMF Reduction, PCA Decomposition, Maximum Peak and Spectrum Mean) to take advantage of the variable collinearity and solve the issue of high variable collinearity. A set of real samples obtained after human nutritional intervention with placebo or polyphenol-rich beverages was used to test this methodology. The results showed that applying any peak aggregation method (especially NMF and PCA) improves the statistical prediction power of class pertinence independently of the nature of the classifier (linear PLS-DA or non-linear SVM). Overall, the introduction of this new approach resulted in a reduction of the dimensionality of the data and, in addition, in a significant increase in the overall predictive power of the data.

4.2 Introduction

The Liquid Chromatography-Mass Spectrometry (LC-MS)-based metabolomic analyses produce data sets with a high level of complexity. High feature (i.e. peak) collinearity values are an important characteristic of this complexity. The origin and meaning of this collinearity (e.g. correlations) has already been revised [1, 2]. Among the different kind of correlations found in the LC-MS metabolomics data sets it should be highlighted the statistical relevance of the correlations between features coming from the same metabolite. In this context, it is assumed that an LC-MS metabolomics dataset consists in a mixture where, with the (de)protonated ion, it is possible to find different features corresponding to the formation of adducts [3], isotopes and fragments ions coming from the ionization source [4] that show high levels of correlation. In particular, Moco et al [5] showed that after analysis of the correlations in an LC-MS dataset containing more than 3.000 mass signals the highest positive correlations were found for mass signals belonging to the same metabolite.

Typically, after the LC-MS acquisition stage[6, 7, 8], the data takes the form of a matrix having mass features (treated as variables) as columns and samples as rows. This matrix is used as the basic dataset in the algorithms which process LC-MS data. In this context, multivariable methods such as Principal Components Analysis (PCA), Partial Least Squares (PLS) and Partial Least Squares Discriminant Analysis (PLS-DA) are widely used to extract the significant features from such data sets [9]. Some more specific procedures, for example the application of an orthogonal filter before the PLS (OPLS) or PLS-DA (OPLS-DA) to separate the between-class from the within-class variance [9, 10, 11] are also commonly used. Moreover, univariate statistical tests such as t-tests or ANOVA[12] are also applied obtaining a p-value for each mass feature. It has been reported that when collinearity among variables is found in performing regression or discriminant analysis, this may lead to biased regressor estimators [13]. In the case of an exact linear relationship between the variables, it is not even possible to find a unique predictor[13]. In consequence, collinearity among variables should be controlled in order to arrive at better-fitting regressions.

The main aim of this paper is to explore the potential of shifting from single mass features towards a mass peak spectrum oriented analysis. For each mass spectrum, the feature measure is obtained through peak aggregation techniques. The effect of considering this new variable measures instead of single features is evaluated by the selection of significant features using a standard ANOVA test and by obtaining the classification ratio in a classification stage by using two different classifiers: Support Vector Machines (SVM) and PLS-DA [14].

4.3 Materials and Methods

4.3.1 Spectrum Definition

In the context of data obtained in full scan mode, the concept of spectrum is defined as the set of features/peaks including the (de)protonated ion, isotopes, adducts and fragments originated in the ionisation source that eluted at the same retention time. Therefore, all peaks produced by the same metabolite will show large correlation properties.

4.3.2 Peak Aggregation Techniques

Let Y be the peak data set matrix (dimension of $Y = n \times m$) where the elements of this matrix are the intensity of the peaks in the samples, n is the number of samples in the peak data set, m is the number of peaks in the peak data set. Let p be the total number of spectra present in the peak data set with $p \leq m$ (assuming that p matches the actual number of metabolites in the data). In the process of changing to a metabolite-based scheme, each peak is assigned to a metabolite as defined in previous section. Several criteria are available to perform this assignment. In this paper, we use the correlation-based approach proposed by Kuhl et al [15]. This method relates each spectrum to a submatrix of the peak data set known as a spectral submatrix. In general, as each spectrum may have any number of peaks, the dimensions of these spectral submatrices are different. The complete set of peaks Y from now on is considered to consist of a set of p spectral sub matrices Y^k $(n \times o^k)$ each one having o^k peaks:

$$Y = \{Y^k, k = 1, 2, ...p\} \tag{4.1}$$

Applying either multivariate techniques or statistical tests on LC-MS samples using spectra as variables is not straightforward due to the different dimensions of the spectral submatrices. Different peak aggregation techniques are studied in this technical note to reduce the dimensionality of all spectral submatrices. Each peak aggregation method is based on a multivariate process that is applied independently over each spectrum, resulting in a 1D-spectral submatrix per spectrum.

For each method and for a spectrum k, the effect of applying the method over the spectral data Y^k is mathematically expressed as shown at equation (4.2).

$$Y^k = S^k \cdot (L^k)^T + E^k \tag{4.2}$$

Matrices S^k $(n \times 1)$ and L^k $(o^k \times 1)$ are method-dependent and correspond to the scores and loadings matrices for the k-th spectrum respectively. Matrix E^k $(n \times o^k)$ is the error matrix of the model. The loadings matrix can be thought of as the spectral representation obtained by the applied peak aggregation technique, whereas the scores matrix is the expression of L^k across the samples. Graphically, the interpretation of both loadings and scores matrices for each peak aggregation method used and for a certain spectrum k is depicted in Figure 4.2.

All the resulting p-spectral submatrices S^k can be combined to build a new data set called a spectral data set S $(n \times p)$. Statistical tests or multivariate techniques can then be applied to the spectral data set to extract its significant features. We report results for the following peak aggregation methods: *Maximum Peak*, *Spectral Mean*, *PCA Decomposition* and *Non-negative Matrix Factorisation Reduction* (*NMF Reduction*).

4.3.2.1 No Peak Aggregation Method

When peaks are used as variables, no peak aggregation is performed and the spectral data set S_{None} equals the peak data set Y:

$$S_{None} = Y \tag{4.3}$$

4.3.2.2 Maximum Peak

The *Maximum Peak* method consists in taking the peak having maximum mean values across samples within the spectrum. Mathematically, it can be expressed as is shown in (4.4)

$$S^k_{max} = (Y^k_{lq} \mid q = max_j(\sum_{i=1}^{n} \frac{Y^k_{ij}}{n}),\ l = 1, 2, ..., n) \tag{4.4}$$

where S^k_{max} is an $(n \times 1)$ dimensional matrix. In some metabolomic processing algorithms, the *Maximum Peak* approach is used after building the spectra, to select the peak to be sent as a query to the database [16]. In this kind of algorithm, the spectral maximum peak is chosen as the representative of the spectra and sent to the database to identify the whole metabolite.

4.3.2.3 Spectral Mean

The *Spectral Mean* is a peak aggregation method that applies a mean to the peaks of the spectrum, expressed as (4.5).

$$S^k_{mean} = (\sum_{j=1}^{o^k} \frac{Y^k_{ij}}{o^k} \mid i = 1, 2, ..., n) \tag{4.5}$$

This method considers all peaks in a spectrum with the same weight disregarding of their statistical properties.

4.3.2.4 Principal Component Analysis Decomposition

A natural evolution from the *Spectral Mean* method is the *PCA Decomposition* method. In this peak aggregation method, a PCA is performed on every spectrum k as shown in equation 4.6. This method builds an aggregated factor through a maximum variance criteria through a PCA decomposition. A data centered PCA model is constructed for each Y^k matrix and the set of scores corresponding to the first principal component is employed as aggregated index. In equation (4.6), T^k $(o^k \times 1)$ is the first principal component and P^k $(n \times 1)$ the first score vector whereas E^k_{PCA} $(n \times o^k)$ is the error matrix.

$$Y^k = P^k \cdot (T^k)^T + E^k_{PCA} \tag{4.6}$$

The spectral data set when *PCA Decomposition* method is used is shown at (4.7).

$$S^k_{PCA} = P^k \tag{4.7}$$

Provided that the loadings change for each of the peaks of the spectrum, this peak aggregation method can take into account complex peak collinearity patterns between peaks of the same spectrum. However, interpretability of the output of the PCA method is not possible as, in that case, negative values are allowed in both the loadings and the scores.

4.3.2.5 Non-Negative Matrix Factorisation Reduction

An alternative decomposition is to impose non-negativity on the computation of the peak loadings corresponding to each spectrum. This can be achieved through a Non Negative Matrix Factorisation of Y^k into the product of two matrices H^k $(n \times 1)$ and W^k $(o^k \times 1)$ plus an error matrix E^k_{NMF} $(n \times o^k)$ shown at (4.8) [17]. NMF is a versatile technique which has been used in some other pattern discovery fields[18] and which may be obtained by several different mathematical criteria [19]. In the NMF method, all the matrix components of the H-matrix and the W-matrix are non-negative [17]. The matrix components were obtained by performing an optimisation procedure, which consisted of a minimisation of the Kullback-Leibler divergence [20] between the spectral data set Y^k and the product $H^k \cdot (W^k)^T$.

$$Y^k = H^k \cdot (W^k)^T + E^k_{NMF} \tag{4.8}$$

When NMF is applied to a spectrum, the component capturing maximum variance of the Y^k matrix is chosen to be the 1D-spectral matrix as shown in (4.9).

$$S^k_{NMF} = H^k \tag{4.9}$$

This method has the advantage of better interpretability. As the initial data values are intensities, all the components of the peak data set X are positive. The resulting spectral data set under the NMF method, S_{NMF}, is a matrix all of whose components are positive. The components of the spectral data set $S_{ij;NMF}$ can be understood as the expression of a certain spectrum j in the sample i. As the W-matrix represents a certain averaged spectrum, the components $S_{ij;NMF}$ can be understood as a type of spectral intensity. The R source code of the methods will be available through the Bioconductor repository.

4.3.3 Experimental Data

4.3.3.1 Experimental design

A randomized, crossover, placebo-controlled, double-blind intervention study was performed with 30 volunteers, who consumed 187 mL of a control placebo (class A) or a functional beverage (FB) (class B) in an acute study, and twice a day during 15 days for a chronic consumption study (15 days placebo samples were labelled as class C and 15 days FB samples as class D). Twenty-four-hour urine samples were collected the day before the acute intervention study and on the last day of the chronic study. For the acute study, the urine samples were collected in the first 4 hours. The FB was a grape extract preparation obtained from grape skins with alcohol mixtures at different temperature conditions. The study protocol was approved by the Ethics Committee of the Hospital Universitario La Paz, Madrid.

4.3.3.2 Urine analysis

The samples were analysed by liquid chromatography coupled with a hybrid quadrupole time-of-flight (LC-q-TOF, AB/MDS Sciex) in positive mode using the protocol proposed by Tulipani et al[10]. LC was performed in HPLC Agilent using a RP 18 Luna column (50 X 2.0 mm, 5μm), with a sample injection volume of 15 μL. A linear gradient elution was performed consisting of [A] Milli-Q water 0.1% HCOOH (v/v) and [B] acetonitrile 0.1% HCOOH (v/v). The gradient elution (v/v) of [B] was: (time, min; B, %): (0, 1), (4, 20), (6, 95), (7.5, 95), (8, 1), (12, 1). Q-TOF spray parameters were set as previously described[21, 22] and full data acquisition was performed scanning from 70 to 700 m/z. The TOF was calibrated with reserpine (1 pmol/μL). LC-MS data were acquired in random order to avoid possible bias. Finally, the files were translated to the netCDF format.

4.3.4 Statistical Validation

In order to evaluate the quality of the significant features found by all the five methods, a cross-validation classification step was performed over the spectral data set S, using only the significant features (section A.2) and 5 training samples for each class in a repeated random sub-sampling cross-validation stage. For each of the 5 methods (the standard method and the four peak aggregated methods) a total of 300 repetitions was performed. In each repetition, the training set of 5 samples/class was chosen randomly between all the samples. Two different classifiers were used: PLS-DA which is a linear classifier, and SVM which is a non-linear one [14] (further details in section A.2). In order to selectively check the effect of the peak aggregation strategy, only spectra having more than one peak are considered valid in this analysis.

4.4 Results

4.4.1 Effect of the Peak Aggregation

Figure 4.1 shows a typical pattern for correlation values between peaks in LC-MS data sets (section A.1). It can be seen from this Figure that there exist peak blocks having high correlation values. The peaks of these highly correlated peak blocks showing similar retention times are likely to come from the same metabolite and are considered to constitute a spectrum.

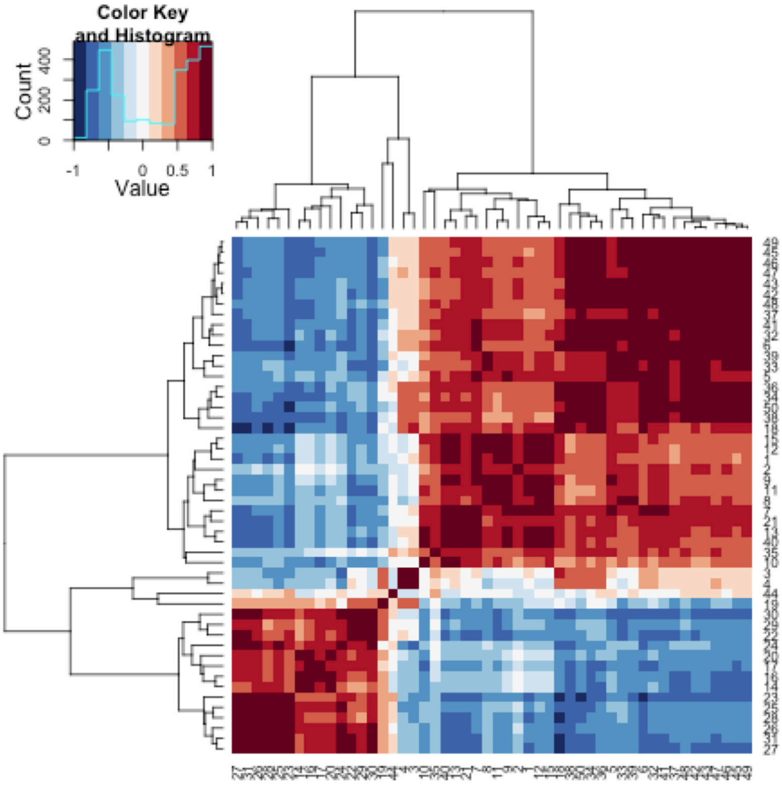


Figure 4.1: Heatmap showing the correlation values profile for a subset of 50 peaks in the metabolomic LC-MS sample set. Both dendrograms were performed using correlation distance. The Figure depicts a typical pattern for correlation values between peaks in LC-MS data sets (section A.1) showing that there exist peak blocks having high correlation values. It is possible to conclude that those peaks showing high correlation and similar retention time (because they were produced after chromatography) came from the same metabolite.

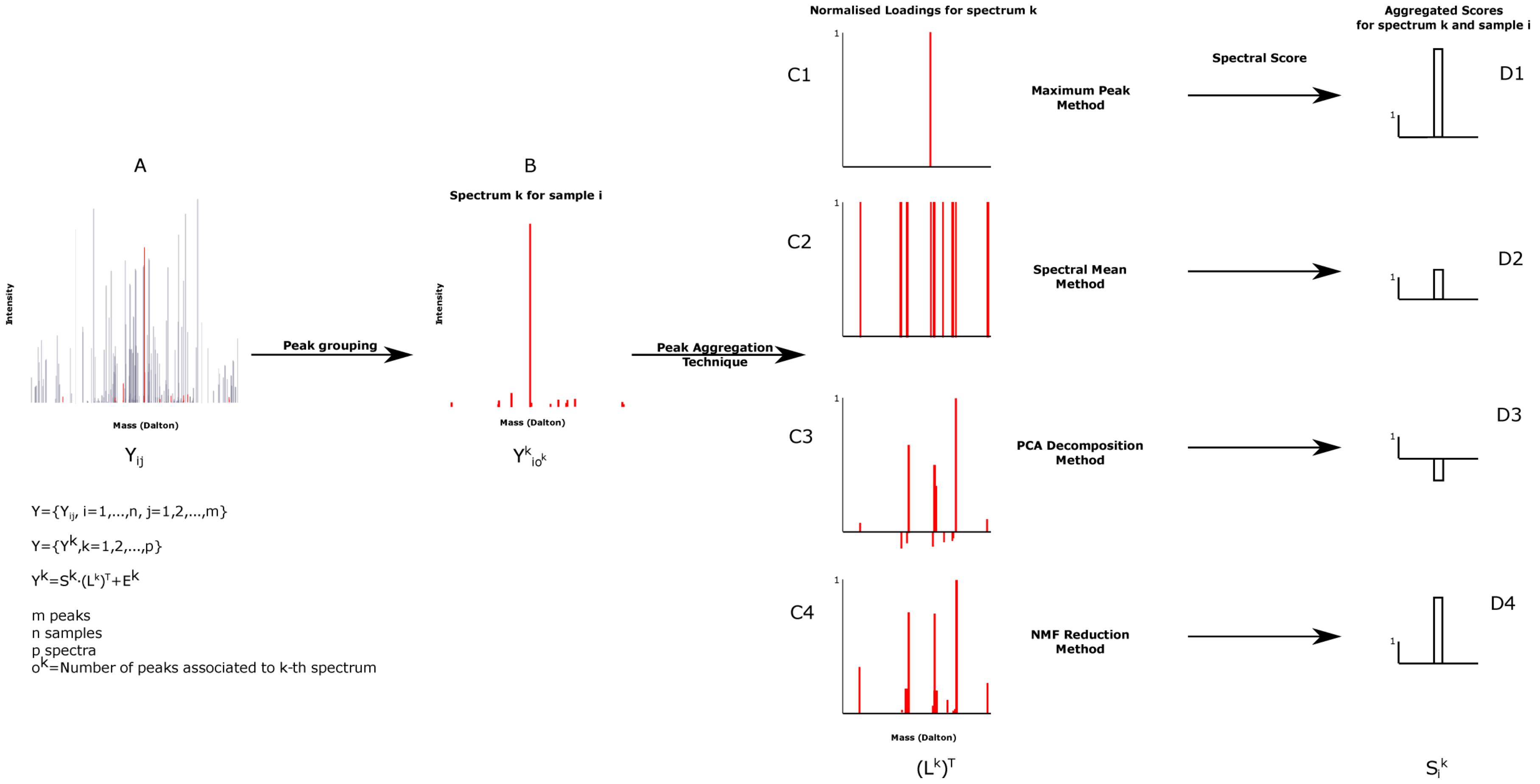


FIGURE 4.2: Sequence of plots depicting an example of the differences in using peak aggregation methods on a single spectrum. Plot A shows in red a spectrum for one of the samples among other peaks. Plot B depicts just this spectrum. Plots C1 to C4 show the spectrum under the chosen peak aggregation technique (loadings matrix $(L^k)^T$ at Equation 4.2) whereas Plots D1 to D4 show how this spectrum is expressed across samples (scores matrix S^k at Equation 4.2)

In order to illustrate in more detail the effect of using one or another peak aggregation method on the spectra, Figure 4.2 shows how a sample spectrum is processed depending on which method is applied. Using different peak aggregation techniques implies having a different spectral pattern (matrix L^k at Equation 4.2) and a different expression of this spectrum across samples (cf. matrix S^k at Equation 4.2). In Figure 4.2, the picture B corresponds to the original spectrum for only one sample, whereas the C plots of the Figure show the result of applying peak aggregation methods to a single spectrum. The D plots correspond to the aggregated value for the sample spectrum and the analysed sample.

From Figure 4.2, it can be seen that the *Maximum Peak* method (plots C1 and D1) is a restrictive one, as only the maximum peak of each spectrum is considered. In this case, the expression of the spectrum over samples is the expression of the maximum peak over samples. The *Spectral Mean* method (plots C2 and D2) enforces all the peaks of the spectrum to have equal weight. The main difference between using *PCA Decomposition* (plots C3 and D3) and *NMF Reduction* (plots C4 and D4) as opposed to using the *Maximum Peak* or the *Spectral Mean* is that the weight of each peak in the final spectrum value depends not only on the peaks, but also on the expression of those peaks over the samples. The C3 and D3 plots of Figure 4.2 show that the *PCA Decomposition* allows for negative values for both the spectrum pattern and for the expression of the spectrum over samples

Finally, plots C4 and D4 show the effect of applying the *NMF Reduction* on the sample spectrum. All the values in both the spectrum pattern $(L^k)^T$ and the expression of the spectrum over samples S^k are positive. This gives better interpretability of the spectral pattern which can now be understood as weighted intensity measures over samples of the spectrum. In consequence, the C4 plot would be the spectrum where the weights have taken into account the different peak patterns across samples and the D4 plot would be the expression of this spectrum on the sample i.

4.4.2 Class Prediction Results

After applying a peak aggregation method, the variables are now the metabolite spectra L^k rather than the peaks themselves. In consequence, the peak correlation patterns due to the peaks coming from the same metabolite are no longer present in the data and hidden correlation patterns then emerge. To test if the spectral data set S improves the predictive power of the data, the metabolomic LC-MS sample files were processed following the steps described in section A.2 after which the classification procedure explained in section "Statistical Validation" was applied.

The results of performing the 300 classification steps and their associated Fisher's Least Significant Difference (LSD) parameters for each method and both classifiers are shown at table 4.1 (Figures S-4 and S-5). Irrespective of the classifier used, clear tendencies emerge when peak aggregation methods are used. The results, using both classifiers, show that applying peak aggregation methods where spectra are used as variables improves the predictive performance compared to the standard methods, where peaks are used. This improvement is at least equal to 14% and as much as 18% in the mean value of the classification ratio depending on the method used (Table 4.1). *PCA Decomposition* and *NMF Reduction* showed the best performance and were indistinguishable in both classifiers according to the Fisher's LSD results. The main difference between using the PLS-DA and the SVM classifiers was that, when PLS-DA was used, there was no difference between applying the *Spectral Mean* or the *Maximum Peak* as peak aggregation methods, whereas when SVM is used, the *Spectral Mean* showed better performance than the *Maximum Peak* method.

To give a deeper insight to the improvement of the overall classification ratio, Figure 4.3 shows the classification ratios for each of the four classes, using either *NMF Reduction* as a peak aggregation technique or not using any method. The results show improvements in the classification ratios for all the four classes when *NMF Reduction* is used. This suggests that using *NMF Reduction* as a peak aggregation method improves the overall statistical power of the data, regardless of the similarity between classes.

Table 4.1: Table showing the output parameters of the Fisher's LSD test for the PLS-DA and SVM classifiers after 300 classification steps. The method labelled "None" corresponds to the standard method in which no peak aggregation is performed. Least Significant Difference was found to be 0.013 for PLS-DA and 0.012 for SVM. Mean differences lower than this value cannot be told apart. The labels LCL and HCL in the columns correspond to Lower Confidence Limit and Higher Confidence Limit respectively. Methods having one asterisk in the superscript mean that Fisher's LSD test found them to be equal regardless of the classifier used. Methods having two asterisk in the superscript mean that Fisher's LSD test only found them to be equal in the PLS-DA classifier.

	PLS-DA				**SVM**			
Method	**Mean**	**Std error**	**LCL**	**HCL**	**Mean**	**Std error**	**LCL**	**HCL**
Mean**	0.803	0.003	0.797	0.809	0.804	0.003	0.798	0.809
NMF*	0.817	0.003	0.812	0.824	0.820	0.003	0.814	0.825
None	0.636	0.004	0.629	0.643	0.652	0.003	0.646	0.659
PCA*	0.821	0.003	0.814	0.827	0.817	0.003	0.812	0.822
Maximum**	0.795	0.003	0.789	0.801	0.790	0.003	0.784	0.797

4.5 Conclusions

This paper has studied the effect of performing peak aggregated measures in the statistical analysis of metabolomic LC-MS urine samples. Using peak aggregation techniques implies a change from a single mass features to a metabolite spectrum oriented analysis. Different peak aggregation techniques, which imply different spectral data sets, have been compared to each other and also to the not-aggregated standard analysis. The results showed that using peak aggregation methods in the statistical analysis improves the statistical power of the LC-MS data independently of whether the classifier is linear (PLS-DA) or non-linear (SVM). Considering the classification ratio as the quality metrics for both classifiers, it was shown that using *NMF Reduction* or a *PCA Decomposition* methods over each spectrum are the methods which most improve the detection of significant features.

Bibliography

[1] Diogo Camacho, Alberto Fuente, and Pedro Mendes. The origin of correlations in metabolomics data. *Metabolomics*, 1(1):53–63, 2005. URL http://www.springerlink.com/index/10.1007/s11306-005-1107-3.

[2] Ralf Steuer. Review: on the analysis and interpretation of correlations in metabolomic data. *Briefings in bioinformatics*, 7(2):151–8, June 2006. ISSN 1467-5463. doi: 10.1093/bib/bbl009. URL http://bib.oxfordjournals.org/cgi/content/long/7/2/151.

[3] Erwan Werner, Jean-François Heilier, Céline Ducruix, Eric Ezan, Christophe Junot, and Jean-Claude Tabet. Mass spectrometry for the identification of the discriminating signals from metabolomics: current status and future trends. *Journal Of Chromatography B Analytical Technologies In The Biomedical And Life Sciences*, 871(2):143–163, 2008.

[4] Robert S Plumb, Kelly A Johnson, Paul Rainville, Brian W Smith, Ian D Wilson, Jose M Castro-Perez, and Jeremy K Nicholson. Uplc/ms(e); a new approach for generating molecular fragment information for biomarker structure elucidation. *Rapid communications in mass spectrometry RCM*, 20(13):1989–1994, 2006. URL http://doi.wiley.com/10.1002/rcm.2602.

[5] Sofia Moco, Jenny Forshed, Ric C H Vos, Raoul J Bino, and Jacques Vervoort. Intra- and inter-metabolite correlation spectroscopy of tomato metabolomics data obtained by liquid chromatography-mass spectrometry and nuclear magnetic resonance. *Metabolomics*, 4(3):202–215, 2008. URL http://www.springerlink.com/index/10.1007/s11306-008-0112-8.

[6] Mikko Katajamaa and Matej Orešič. Processing methods for differential analysis of LC/MS profile data. *BMC Bioinformatics*, 6(1):179, 2005.

[7] C A Smith, E J Want, G O'Maille, R Abagyan, and G Siuzdak. XCMS: processing mass spectrometry data for metabolite profiling using nonlinear peak alignment, matching, and identification. *Analytical Chemistry*, 78(3):779–787, 2006. ISSN 00032700. doi: 10.1021/ac051437y. URL http://pubs3.acs.org/acs/journals/doilookup?in_doi=10.1021/ac051437y.

[8] Rolf Danielsson, Dan Bylund, and Karin E. Markides. Matched filtering with background supression for improved quality of base peak chromatograms and mass spectra in liquid chromatography-mass spectrometry. *Analytica Chimica Acta*, 454(2):167–184, November 2002. ISSN 01697439.

[9] Johan Trygg, Elaine Holmes, and Torbjörn Lundstedt. Chemometrics in metabonomics. *Journal of proteome research*, 6(2):469–79, February 2007. ISSN 1535-3893. doi: 10.1021/pr060594q.

[10] Sara Tulipani, Rafael Llorach, Olga Jáuregui, Patricia López-Uriarte, Mar Garcia-Aloy, Mònica Bullo, Jordi Salas-Salvadó, and Cristina Andrés-Lacueva. Metabolomics Unveils Urinary Changes in Subjects with Metabolic Syndrome following 12-Week Nut Consumption. *Journal of Proteome Research*, 2011. ISSN 15353907. doi: 10.1021/pr200514h.

[11] Hiroyuki Yamamoto, Hideki Yamaji, Yuichiro Abe, Kazuo Harada, Danang Waluyo, Eiichiro Fukusaki, Akihiko Kondo, Hiromu Ohno, and Hideki Fukuda. Dimensionality reduction for metabolome data using PCA, PLS, OPLS, and RFDA with differential penalties to latent variables. *Chemometrics and Intelligent Laboratory Systems*, 98(2):136–142, October 2009. ISSN 01697439. doi: 10.1016/j.chemolab.2009.05.006.

[12] Lin Hui-Ming, J. Edmunds Selley, A. Helsby Nuala, Lynette R. Ferguson, and Daryl D. Rowan. Nontargeted Urinary Metabolite Profiling of a Mouse Model of Crohn's Disease. *Journal of Proteome Research*, 8:2045–2057, January 2009.

[13] Tormod Ns and Bjrn-Helge Mevik. Understanding the collinearity problem in regression and discriminant analysis. *Journal of Chemometrics*, 15(4):413–426, 2001. ISSN 08869383. doi: 10.1002/cem.676.

[14] T. Hastie, R. Tibshirani, and J. H. Friedman. *The Elements of Statistical Learning*. Springer, corrected edition, July 2003. ISBN 0387952845. URL http://www.worldcat.org/isbn/0387952845.

[15] C. Kuhl, R. Tautenhahn, C. Boettcher, T. R. Larson, and S. Neumann. Camera: an integrated strategy for compound spectra extraction and annotation of liquid chromatography/mass spectrometry data sets. *Analytical Chemistry*, 84:283–289, 2012. URL http://pubs.acs.org/doi/abs/10.1021/ac202450g.

[16] Y M Tikunov, S Laptenok, R D Hall, A Bovy, and R C H Vos. MSClust: a tool for unsupervised mass spectra extraction of chromatography-mass spectrometry ionwise aligned data. *Metabolomics*, pages 1–5, 2011. ISSN 15733882. doi: 10.1007/s11306-011-0368-2.

[17] D D Lee and H S Seung. Learning the parts of objects by non-negative matrix factorization. *Nature*, 401(6755):788–91, 1999. ISSN 00280836. doi: 10.1038/44565.

[18] Jean-Philippe Brunet, Pablo Tamayo, Todd R Golub, and Jill P Mesirov. Metagenes and molecular pattern discovery using matrix factorization. *Proceedings of the National Academy of Sciences of the United States of America*, 101(12):4164–4169, 2004.

[19] M Berry, M Browne, a Langville, V Pauca, and R Plemmons. Algorithms and applications for approximate nonnegative matrix factorization. *Computational Statistics & Data Analysis*, 52(1):155–173, September 2007. ISSN 01679473. doi: 10.1016/j.csda.2006.11.006.

[20] S Kullback and R A Leibler. On information and sufficiency. *The Annals of Mathematical Statistics*, 22(1):79–86, 1951. ISSN 00034851. doi: 10.1214/aoms/1177729694.

[21] R Llorach, M Urpi-Sarda, O Jauregui, M Monagas, and C Andres-Lacueva. An lc-ms-based metabolomics approach for exploring urinary metabolome modifications after cocoa consumption. *J Proteome Res*, 8:5060–5068, 2009.

[22] Rafael Llorach, Ignacio Garrido, Maria Monagas, Mireia Urpi-Sarda, Sara Tulipani, Begona Bartolome, and Cristina Andres-Lacueva. Metabolomics study of human urinary metabolome modifications after intake of almond (prunus dulcis (mill .) d . a . webb) skin polyphenols research articles. *Journal of Proteome Research*, 9 (11):5859–5867, 2010.

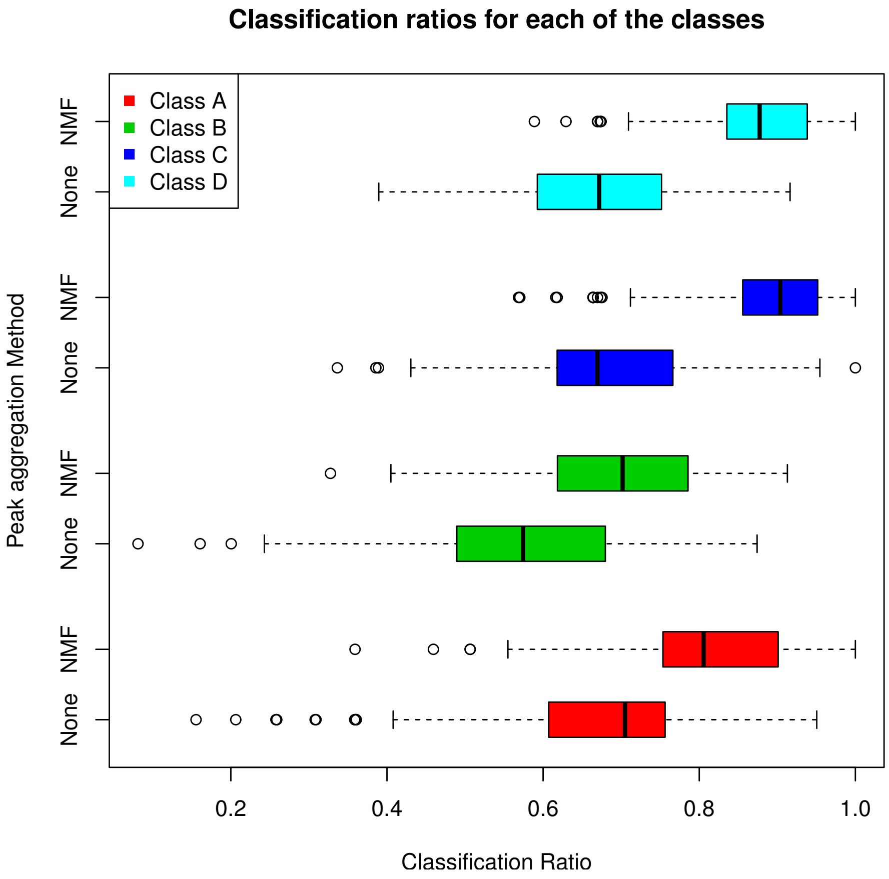


Figure 4.3: Comparative boxplot showing the different classification ratios for the four classes of the samples using either NMF as a peak aggregation technique or no using any peak aggregation technique. The classifier used was SVM.

Chapter 5

An R package to analyse LC/MS metabolomic data: MAIT (Metabolite Automatic Identification Toolkit)

Note: This chapter is an exact copy of the paper:

An R package to analyse LC/MS metabolomic data: MAIT (Metabolite Automatic Identification Toolkit)
Francesc Fernández-Albert, Rafael Llorach, Cristina Andres-Lacueva, Alexandre Perera-Lluna
Bioinformatics 2014 30 (13): 1937-1939. doi: 10.1093/bioinformatics/btu136

5.1 Abstract

Current tools for Liquid Chromatography and Mass Spectrometry (LC/MS) for metabolomic data cover a limited number of processing steps, whereas online tools are hard to use in a programmable fashion. This paper introduces the Metabolite Automatic Identification Toolkit (MAIT) package, which makes it possible for users to perform metabolomic end-to-end LC/MS data analysis. MAIT is focused on improving the peak annotation

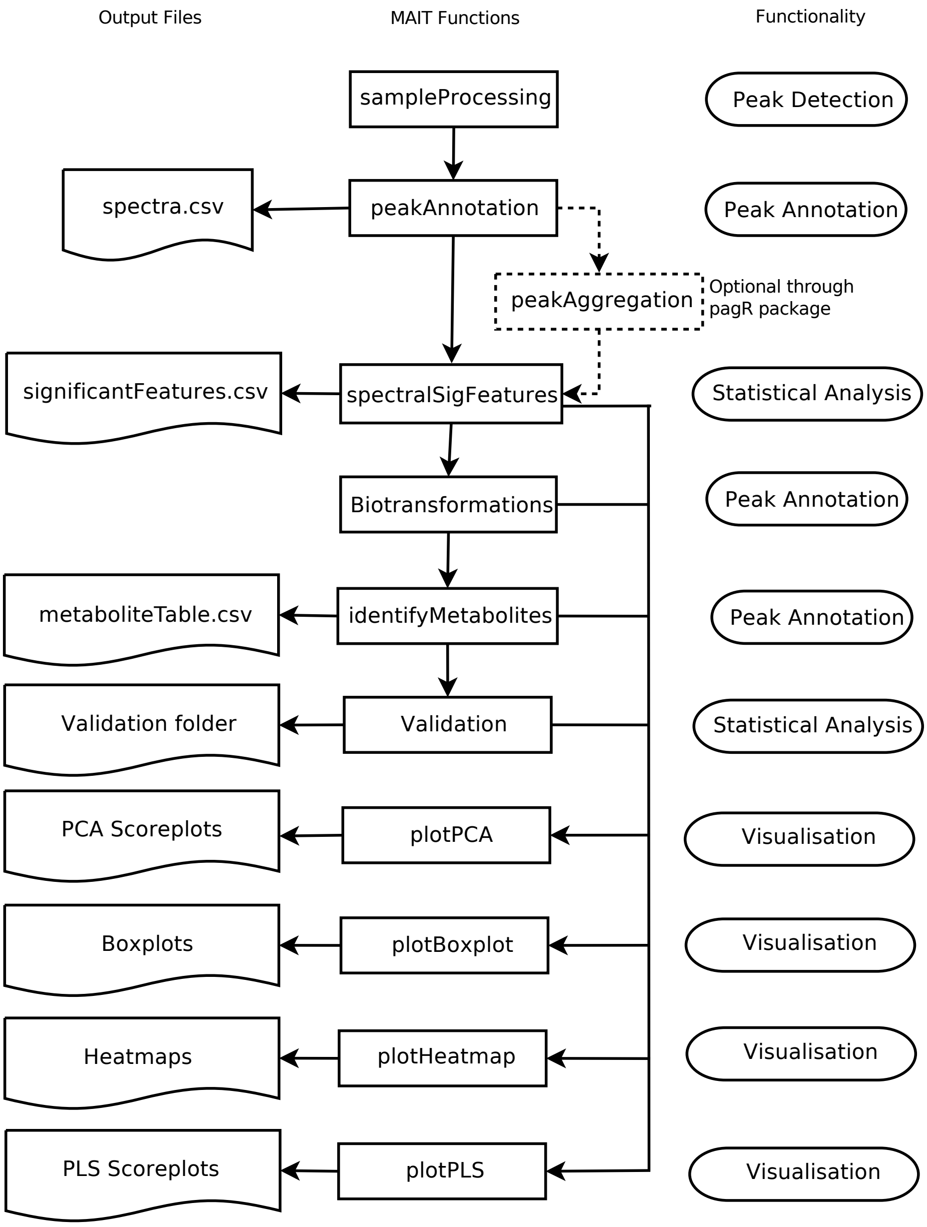


FIGURE 5.1: Correspondence between MAIT functions (centre column), generated output files (left column) and their functionality (right column).

stage and provides essential tools to validate statistical analysis results. MAIT generates output files with the statistical results, peak annotation and metabolite identification.

5.2 Availability:

http://b2slab.upc.edu/software-and-downloads/metabolite-automatic-identification-toolkit/

5.3 Introduction

Liquid Chromatography and Mass Spectrometry (LC/MS) is an analytical technique used widely in metabolomics to detect molecules in biological samples [1]. A wide array of software tools are available for LC/MS profiling data analysis, including commercial, programmatic and online tools. A commercial example is Analyst® whereas some open source packages permit programmatic processing, such as the R package XCMS [2] to detect peaks or CAMERA [3] and AStream [4] for peak annotations. There have been efforts on just peak annotation using JAVA [5]. MZmine and mzMatch are modularised tools coded in JAVA that are focused on LC/MS data preprocessing and visualisation [6, 7, 8]. Online tools permit sample processing through a web GUI, such as XCMSOnline (`http://xcmsonline.scripps.edu`) or MetaboAnalyst [9]. Refer to table 1 of the supplementary material for a comparative between the capabilities for some of the main available tools. In this context, we introduce a new R package called Metabolite Automatic Identification Toolkit (MAIT) for automatic LC/MS analysis. The goal of the MAIT package is to provide an array of tools that make programmable metabolomic end-to-end statistical analysis possible (see section 3 of the supplementary material for details about the MAIT modularity). MAIT includes functions to improve peak annotation through the processes called biotransformations and to assess the predictive power of statistically significant metabolites that quantify class separability.

5.4 Methods

MAIT includes the stages: peak detection, peak annotation, statistical analysis and table and plots creation (see Figure 5.1). The peak detection stage detects the peaks in the LC/MS sample files. The peak annotation stage improves the identification of the metabolites in the metabolomic samples by increasing the chemical and biological information in the data set. A statistical analysis reveals the significant sample features and measures their predictive power. MAIT uses the R package XCMS to detect and align peaks. For the peak annotation step, MAIT uses 3 steps:

- First, MAIT uses the CAMERA package to perform the first annotation step [3]. In this stage, a peak correlation distance and a retention time window to find which peaks came from the same source metabolite based. The peaks within each peak group are annotated following a reference adduct/fragment table and a mass allowance window.

- Biotransformations could be related to specific in-source mass losses. Therefore, in the second annotation step, they are detected using a mass allowance window inside the peak groups [10]. For this search, MAIT already includes a biotransformations table (here Human biotransformations). User-defined biotransformation tables can be set as input, following the procedure defined in Supplementary text (Section 6.6).
- Finally, a predefined metabolite database is mined for significant masses. This identifies metabolites with the help of the Human Metabolome Database [11], 2009/07 version.

The objective of analysing the metabolomic profiling data is to obtain the statistically significant features (SSF) that contain the highest amount of class-related information. To gather these features, MAIT can apply statistical tests such as ANOVA or Student's t-test to every feature, selecting the significant set of features given a threshold P-value. A validation test is included to quantify SSF class separability by a repeated random sub-sampling cross-validation using three methods: partial least squares and discriminant analysis (PLSDA), support vector machines (SVM) and K-nearest neighbours (KNN) [12]. MAIT computes overall and class-related classification ratios to evaluate the SSF class-related information.

5.5 Results

The example data files are a subset of the data used in the reference [13], which are distributed freely through the faahKO package [14]. MAIT was used to read and analyse these samples using the functions depicted in Figure 5.1 (see the tutorial in the supplementary info). The significant features for each class are found using statistical tests and analysed through the different plots that MAIT produces. Using the following function call, 2640 peaks were detected:

```
R> MAIT <- sampleProcessing(dataDir = "Dataxcms", project
  = "MAIT_Demo", snThres = 2, rtStep = 0.03)
```

At this point, the first annotation stage is launched:

```
 R> MAIT <- peakAnnotation(MAIT.object = MAIT)
```

Next, we gather the significant features from the peaks detected. After the Welch's tests, 106 of these features were found to be significant through the spectralSigFeatures function. Statistical plots such as heat maps, boxplots and principal component analysis (PCA) score plots can be generated (Supplementary Figures 3 and 4). Significant features are annotated after checking for certain neutral losses (biotransformations).

```
R> MAIT <- spectralSigFeatures(MAIT, pvalue = 0.05)
R> MAIT <- Biotransformations(MAIT, peakPrecision = 0.005)
```

By using only the SSF, a validation stage is launched, obtaining a classification ratio of 100% with 3 training samples for all classifiers. These results suggest that the significant variables separate both classes completely.

```
R> MAIT <- Validation(MAIT, Iterations = 20, trainSamples= 3)
```

Finally, the database is mined to identify the significant features.

```
R> MAIT <- identifyMetabolites(MAIT, peakTolerance = 0.005)
```

5.6 Conclusions

MAIT provides a set of tools and functions to perform an automatic end-to-end analysis of LC/MS metabolomic data, putting special emphasis on peak annotation and metabolite identification. In addition, MAIT validation functions make it possible to estimate predictive power for significant variables.

Bibliography

[1] Georgios a Theodoridis, Helen G Gika, Elizabeth J Want, and Ian D Wilson. Liquid chromatography-mass spectrometry based global metabolite profiling: a review. *Analytica chimica acta*, 711:7–16, January 2012. ISSN 1873-4324. doi: 10.1016/j.aca.2011.09.042.

[2] C A Smith, E J Want, G O'Maille, R Abagyan, and G Siuzdak. XCMS: processing mass spectrometry data for metabolite profiling using nonlinear peak alignment, matching, and identification. *Analytical Chemistry*, 78(3):779–787, 2006. ISSN 00032700. doi: 10.1021/ac051437y. URL http://pubs3.acs.org/acs/journals/doilookup?in_doi=10.1021/ac051437y.

[3] Carsten Kuhl, Ralf Tautenhahn, and Steffen Neumann. LC-MS Peak Annotation and Identification with CAMERA. *Camera*, pages 1–14, 2011.

[4] Arnald Alonso, Antonio Julià, Antoni Beltran, Maria Vinaixa, Marta Díaz, Lourdes Ibañez, Xavier Correig, and Sara Marsal. Astream: an r package for annotating lc/ms metabolomic data. *Bioinformatics*, 27(9):1339–1340, 2011. URL http://www.ncbi.nlm.nih.gov/pubmed/21414990.

[5] Marie Brown, David C Wedge, Royston Goodacre, Douglas B Kell, Philip N Baker, Louise C Kenny, Mamas A Mamas, Ludwig Neyses, and Warwick B Dunn. Automated workflows for accurate mass-based putative metabolite identification in lc/ms-derived metabolomic datasets. *Bioinformatics*, 2011. URL http://dx.doi.org/10.1093/bioinformatics/btr079.

[6] Mikko Katajamaa, Jarkko Miettinen, and Matej Oresic. MZmine: toolbox for processing and visualization of mass spectrometry based molecular profile data. *Bioinformatics (Oxford, England)*, 22(5):634–636, 2006. ISSN 1367-4803. doi: 10.1093/bioinformatics/btk039.

[7] Tomás Pluskal, Sandra Castillo, Alejandro Villar-Briones, and Matej Oresic. MZmine 2: modular framework for processing, visualizing, and analyzing mass spectrometry-based molecular profile data. *BMC bioinformatics*, 11:395, 2010.

[8] Richard A Scheltema, Andris Jankevics, Ritsert C Jansen, Morris A Swertz, and Rainer Breitling. PeakML/mzMatch: a file format, Java library, R library, and tool-chain for mass spectrometry data analysis. *Analytical chemistry*, 83(7):2786–2793, 2011.

[9] Jianguo Xia, Nick Psychogios, Nelson Young, and David S. Wishart. Metaboanalyst: a web server for metabolomic data analysis and interpretation. *Nucleic Acids Research*, 37(suppl 2):W652–W660, 2009. doi: 10.1093/nar/gkp356. URL http://nar.oxfordjournals.org/content/37/suppl_2/W652.abstract.

[10] Rainer Breitling, Shawn Ritchie, Dayan Goodenowe, Mhairi L. Stewart, and Michael P. Barrett. Ab initio prediction of metabolic networks using Fourier transform mass spectrometry data, 2006. ISSN 1573-3882.

[11] David S Wishart, Craig Knox, An Chi Guo, Roman Eisner, Nelson Young, Bijaya Gautam, David D Hau, Nick Psychogios, Edison Dong, Souhaila Bouatra, and et al. Hmdb: a knowledgebase for the human metabolome. *Nucleic Acids Research*, 37 (Database issue):D603–D610, 2009. URL http://www.pubmedcentral.nih.gov/articlerender.fcgi?artid=2686599&tool=pmcentrez&rendertype=abstract.

[12] T. Hastie, R. Tibshirani, and J. H. Friedman. *The Elements of Statistical Learning*. Springer, corrected edition, July 2003. ISBN 0387952845. URL http://www.worldcat.org/isbn/0387952845.

[13] Alan Saghatelian, Sunia A Trauger, Elizabeth J Want, Edward G Hawkins, Gary Siuzdak, and Benjamin F Cravatt. Assignment of endogenous substrates to enzymes by global metabolite profiling. *Biochemistry*, 43(45):14332–14339, 2004. URL http://www.ncbi.nlm.nih.gov/pubmed/15533037.

[14] Colin A. Smith. *faahKO: Saghatelian et al. (2004) FAAH knockout LC/MS data*, 2012. URL http://dx.doi.org/10.1021/bi0480335. R package version 1.2.13.

Chapter 6

Intensity drift removal in LC/MS metabolomics by Common Variance Compensation

Note: This chapter is an exact copy of the paper:

Intensity drift removal in LC/MS metabolomics by common variance compensation)

Francesc Fernandez-Albert, Rafael Llorach, Mar Garcia-Aloy, Andrey Ziyatdinov, Cristina Andres-Lacueva, Alexandre Perera

Bioinformatics 2014 30 in press. doi: 10.1093/bioinformatics/btu423

6.1 Abstract

Liquid Chromatography coupled to mass Spectrometry (LC/MS) has become widely used in Metabolomics. Several artefacts have been identified during the acquisition step in large LC/MS metabolomics experiments, including ion suppression, carryover or changes in the sensitivity and intensity. Several sources have been pointed out as responsible for these effects. In this context, the drift effects of the peak intensity is one of the most frequent and may even constitute the main source of variance in the data, resulting in misleading statistical results when the samples are analysed. In this paper, we propose the introduction of a methodology based on a common variance analysis prior to the data normalisation to address this issue. This methodology was tested and

compared with four other methods by calculating the Dunn and Silhouette indices of the Quality Control classes. The results showed that our proposed methodology performed better than any of the other four methods. As far as we know, this is the first time that this kind of approach has been applied in the metabolomics context.

6.2 Availability:

The source code of the methods is available as the R package **intCor** at: http://b2slab.upc.edu/software-and-downloads/intensity-drift-correction/

6.3 Introduction

Metabolomics aims to asses the metabolic changes in a global way to infer biological functions and provide the detailed biochemical responses of cellular systems [1]. Liquid Chromatography/Mass Spectrometry (LC/MS) devices are among the most-used experimental setups in metabolomics. LC/MS analyses of biological samples such as urine or plasma give high-throughput data having a three index scheme: retention time, mass/charge ratio and intensity values [2, 3]. In metabolomic data, as in other types of high-dimensional data such as gas sensor arrays or microarray data, the intensity values of the variables might be biased or might suffer from variations due to external factors. Among these factors is a contribution from the drift of the experimental devices, due to various causes such as column ageing in the case of LC/MS, temperature variations or contamination effects [4, 5]. The presence of peak intensity drift in the data is an important issue, as its effects can be important enough to mask the real statistical behaviour of the data and may indeed be the largest source of variance in the data [4, 6].

In most LC/MS protocols, quality control (QC) samples are regularly injected to ensure good analytical device performance [7]. In LC/MS metabolomics studies the quality controls have been carried out using pools of biological samples, spikes with standards or Milli-Q water samples [8]. These quality control samples consist either of a pooling of all the samples in the study or of a spike-in of some known metabolites (several classes having different types of QC samples might be injected). In the data preprocessing stage, one may distinguish two different steps: data normalisation and data equalisation. We understand the data normalisation step as the mathematical process which makes the *variables* in the data set comparable, whereas the data equalisation step, makes the *samples* from the data set comparable. In the literature, many normalisation and equalisation methods, based on several different approaches and scopes, may be found. Regarding equalisation methods, a methodology using certain internal known metabolites as quality standards to normalise the whole data set has been reported [9]. Another approach is to use the injected samples for internal control (i.e. QCs) to fit a smoothed model for the intensity levels of certain features, and then to correct all the biological samples accordingly [7]. The R package sva includes the ComBat function which compensates the batch effects on microarray data using an empirical Bayes approach [10, 11]. This method has been applied to normalise gene expression and methylation data [12, 13]. Equalisation methods based on a sample-wise correction for LC/MS metabolomic data have also been tested and compared by Veselkov et al. [6]. Their results suggest that a variance stabilisation transformation of the data, followed by a median fold change normalisation, gives the best performance as compared to three other methods. Their method performs a normalisation and an equalisation step to give a robust output when having urine samples with different concentration values. Among the equalisation methods, the one proposed by Artursson et al., based on component correction (CC), was developed in the sensor array field [14]. This method is based on the assumption that, in multivariate data, the drift direction is the first Principal Component (PC) of a PCA decomposition for a class consisting of measurements of the same samples. Such samples are known as technical replicates (i.e. there is no biological or chemical variation in addition to the variability of the technical replication of the measure). Once the drift direction is computed, the drift is removed from the data by subtracting the data projection on the drift direction from the original data. However, if some between-class variability is aligned with the drift direction, it will also be subtracted and some non-drift variability will be removed. A natural extension of the CC method is the one proposed by Ziyatdinov et al. which is based in a Common Principal Component Analysis (CPCA) decomposition [15]. This method proposes modelling the drift contribution in the data as the direction capturing maximum variance that simultaneously diagonalises the covariance matrices of a set of classes. All the variability of the samples in that particular direction is considered to be drift-induced variability, and the projection of the data on that direction is subtracted from the data as in the CC method.

In this paper, to find the drift model, we state the hypothesis that the intensity drift of the chromatograms is the common variance direction of all the QC classes that captures the maximum variance. In this context, we propose a preprocessing method based on a two-step approach by first equalising the data through a CPCA, and then normalising the data using a median fold change step.

6.4 Materials and Methods

6.4.1 Description of the Data

The samples were analysed by liquid chromatography coupled with a hybrid quadrupole time-of-flight (LC-q-TOF, Hybrid quadrupole TOF QSTAR Elite, AB/MDS Sciex) in positive mode using the protocol proposed by Tulipani et al. (Tulipani et al. 2011). LC was performed in HPLC Agilent (Agilent 1200 Series Rapid Resolution HPLC system) using a RP 18 Luna column (50 X 2.0mm, 5m), with a sample injection volume of 15 μL. A linear gradient elution was performed consisting of [A] Milli-Q water 0.1% HCOOH (v/v) and [B] acetonitrile 0.1% HCOOH (v/v).The gradient elution (v/v) of [B] was: (time, min; B, %): (0, 1), (4, 20), (6, 95), (7.5, 95),(8, 1), (12, 1). Q-TOF spray parameters were set as previously described [8] and full data acquisition was performed scanning from 70 to 700 m/z. The TOF was calibrated with reserpine (1 pmol/μL). LC-MS data were acquired in random order to avoid possible bias and the batches equilibrated. Throughout all the analysis, data process quality control (QC) samples were analysed in order to monitor the stability and functionality of the system. The sample collecting span was of 18 days and there was a replacement of the chromatographic column in the process on day 14. There were 994 study samples and 182 QC samples. Three classes of QC samples were used for each batch:

- Water: Milli-Q water samples (n=96 samples).
- Spikes: Standard mixture solution (n=48 samples) consisting of 12 metabolites at the final concentration of 5ppm for all of them except for indole-3-acetic-2,2-d2 acid whose final concentration was of 10 ppm.

- Reference: Urine sample belonging to the one volunteer. (n=38 samples).

6.4.2 Preprocessing

All the methods were applied to the chromatograms without any prior feature detection. The R package XCMS was used to read the chromatograms of the mzXML files containing the sample data [16]. The chromatograms were aligned using an in-house developed R package (UB/UPC). The chromatographic data of all the files read were merged, creating an $n \times m$ chromatogram matrix X. This step required the binning of the retention time in m bins that were given by the XCMS package. Therefore, the chromatogram matrix had samples as rows and retention time as columns (in our case, $n = 1176$ samples and $m = 441$ retention time points). Thus, the i-th row of this matrix corresponds to the chromatogram of the i-th sample. From here on, the variable j refers to the retention time bins in the chromatogram matrix. A class-wise outlier detection and removal procedure was applied to the QC classes. This procedure was based on computing the Score Distance (SD) and an Orthogonal Distance (OD) in a PCA model using the pcaPP R package [17, 18]. QC samples having SD and OD distances greater than the suggested critical values were considered to be outliers and were discarded from the data set [19]. The critical values used in the package were (i) a quantile of the chi-squared distribution for the SD and (ii) a Wilson-Hilferty approximation for the scaled chi-squared distribution for the OD [19, 17]. Using this approach, 9 outlier samples were detected (4 samples in class *reference*, 3 in class *water* and 2 in class *spikes*). As it is known that raw LC/MS metabolomic data suffer from multiplicative noise, we took the logarithm of the data to compensate for such error sources and to convert them into additive noise sources [6]. Once the o outliers were removed, then we could define the quantity $p = n - o$ to be the new sample range. The resulting matrix $Y(p \times m)$ was used as an input parameter for all the normalisation methods tested. This matrix contains the data for both the QC classes and the study class and it can be divided into matrices corresponding to each class (i.e. $Y_{QC}(p_{QC} \times m)$ for the corresponding data set for all QC classes, $Y_r(p_r \times m)$ for the corresponding data set for the reference class, etc.).

6.4.3 Methods

The five methods compared in this paper (CPCA, CC, Median fold change, ComBat and our CPCA+Median Fold Change) have different input parameters. The methods based on a CPCA decomposition or the CC method involve a class (or classes) selection step to use them for the drift modelling. These methods also need as input the number of components of the drift decomposition which are supposed to be captured. The ComBat method needs the batch relation for each class, whereas the Median fold change method does not need any specific input parameter in addition to the data to be normalised.

6.4.3.1 Component Correction

The hypothesis underlying this method is that the drift direction is found in the first PC of a reference class. The methodology used to normalise this data is described in Artursson et. al. [14]. As the feature pattern of the QC samples was more complex than that of the other two QC classes, the reference class was selected to generate the PCA model. Because of this higher complexity, this class is better able to capture the drift in the data than would a class with a simpler feature pattern. Mathematically, the CC method can be expressed as in equation (6.1).

$$Y_r = S \cdot L^T + E \tag{6.1}$$

The methodology proposed by Artursson et al removes one PC, but the method can be generalised to remove as many PCs as can be found in the data. If $Ncomps$ is the number of components to be removed, then S ($p_r \times Ncomps$) is the scores matrix, L ($m \times Ncomps$) is the loadings matrix and E ($p_r \times m$) is the error matrix.

As only one PC is required to perform the normalisation, no further increase in the dimensions of the matrices S and L is necessary. The drift direction is the first PC, which in this case corresponds to the matrix L. The next step is to project the data set Y onto this direction to obtain the drift component in the data. This projection is mathematically expressed as the equation (6.2).

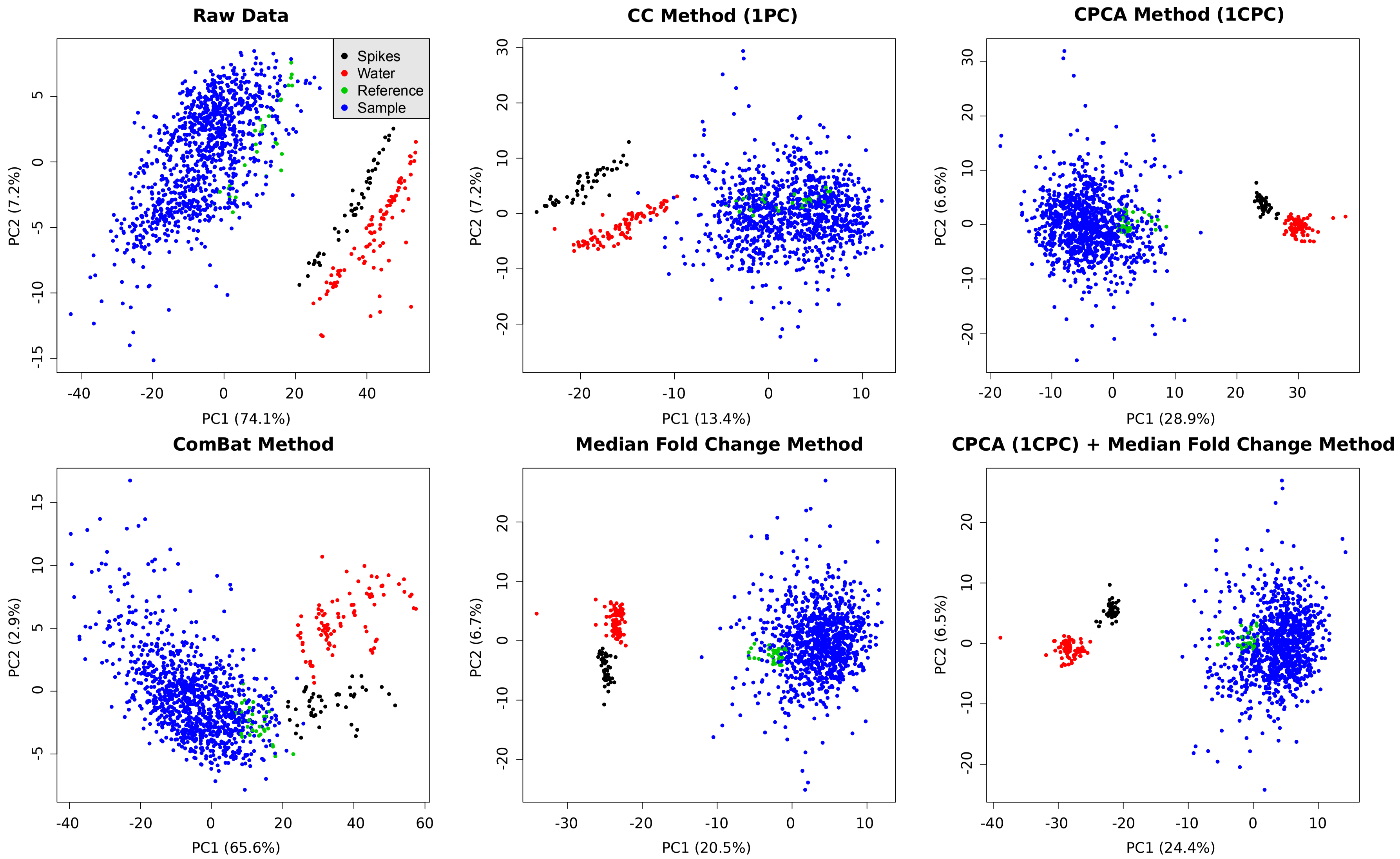


FIGURE 6.1: Set of PCA Scoreplots showing the raw data and the effect on data for each method. The class labelled as sample is the study class. The numbers in brackets on the axes of the plots refer to the estimated variance for that particular direction in the data.

$$Y_d^{cc} = (Y \cdot L) \cdot L^T \tag{6.2}$$

where the superscript cc refers to Component Correction and the subscript d to drift. Once the drift component Y_d^{cc} $(p_r \times m)$ of the data is computed, the last step in removing the drift is to subtract it from the data. Equation (6.3)) shows this last step and the resulting matrix Z^{cc} $(p_r \times m)$ is the corrected matrix using the CC method.

$$Z^{cc} = Y - Y_d^{cc} \tag{6.3}$$

6.4.3.2 Median Fold Change

The Median Fold Change method is not focussed on finding the drift direction. Its objective is to rescale the data to make the median fold changes of the variables close to zero. The methodology followed in applying this method is the one of Veselkov et.al., based on a sample-wise approach [6]. The first step of this method, shown in equation (6.4), is to compute the median for each variable, thus obtaining a vector $\hat{y_i}(1 \times m)$. This vector is used to rescale the original data set Y into a new one, $\hat{Y}(p \times m)$ (see equation (6.4)).

$$\hat{Y_{ij}} = \frac{Y_{ij}}{\hat{y_i}} \text{ where } \hat{y_i} = median_i(Y_{ij}) \tag{6.4}$$

To obtain the normalised data set $Z^M(p \times m)$, the data set Y is divided by the sample median of the matrix $\hat{Y}$ (defined as $\hat{w_j}(p \times 1)$) as shown in equation (6.5)).

$$Z_{ij}^M = \frac{Y_{ij}}{\hat{w_j}} \text{ where } \hat{w_j} = median_j(\hat{Y_{ij}}) \tag{6.5}$$

where the superscript M refers to Median Fold Change method.

6.4.3.3 ComBat

The ComBat method is a function of the R package sva. This function aims to correct the batch effects, which are known to be a source of bias, in gene expression experiments; its extension to LC/MS metabolomic datasets is both natural and straightforward. Firstly, it is assumed that batch effects have multiplicative and additive contributions to the data, and that these effects can be variable-dependent (gene or peak respectively). We state a model following this hypothesis (see equation 6.6)

$$Y_{ijb} = \alpha_j + X\beta_j + \gamma_{jb} + \delta_{jb}\epsilon_{ijb} \tag{6.6}$$

where Y_{ijb} is the intensity value for sample i, variable j and batch b. α_j is the intensity value for variable j, X is the design matrix, β_j contains the regression coefficients of the model, γ_{jb} is a matrix containing the additive batch effects for variable j and batch b, δ_{jb} is a matrix containing the multiplicative batch effects for variable j and batch b and ϵ_{ijb} is the residual matrix of the model. Using either a parametric or a non-parametric empirical prior estimation, the distributions for γ_{jb} and δ^2_{jb} are estimated. The conditional posterior probabilities (γ^*_{jb} and δ^{2*}_{jb}) can then be found and the data is corrected for batch effects as shown in equation (6.7). In the following, all the variables having a hat (ˆ) on them refer to their values estimated from the data.

$$Z^{CB}_{ijb} = \frac{\hat{\sigma_j}}{\hat{\delta}^*_{ij}}(Z_{ijb} - \hat{\gamma_{jb}}^*) + \hat{\alpha_j} + X\hat{\beta_j} \tag{6.7}$$

where the superscript CB refers to the ComBat function and Z_{ijb} is the standardised data and $\hat{\sigma_j}$ is the estimated standard deviation.

6.4.3.4 CPCA

CPCA is a generalisation of the PCA decomposition for different classes first introduced by Flury et. al. [20]. Say we have k classes and Σ_k $(p_k \times p_k)$ are the set of their covariance matrices, then CPCA aims at finding a space such as the one defined by the

V(in general, $p_k \times p_k$) matrix shown in equation (6.8). In the space spanned by V, the covariance matrices for all the classes involved Σ_k are diagonal.

$$\Lambda_k = V^T \cdot \Sigma_k \cdot V \tag{6.8}$$

where Λ_k $(p_k \times p_k)$ is the diagonalised covariance matrix for class k. Each one of the dimensions of this space is called a Common Principal Component (CPC). The hypothesis underlying the CPCA method for drift correction is that the drift direction is contained in the CPC capturing the largest variance. The CPC will be computed by using the Y_{QC} data set (i.e. there are three expressions like equation (6.8), using the different covariance matrices for the QC classes: Σ_r, Σ_{water}, Σ_{spikes}). In a similar way as in a PCA decomposition, given the desired number of CPCs and following a stepwise algorithm, it is possible to compute the number of CPCs one by one [21]. Setting the desired number of CPCs as $Ncomps$, the dimensionality of the V matrix is $(p_k \times Ncomps)$. We have tested the values $Ncomps = 1, 2, 3$ separately for this method.

Once the CPCs are found, the data set is projected onto this space as shown in (6.9)

$$Y_d^{CPCA} = (Y \cdot V) \cdot V^T \tag{6.9}$$

Y_d^{CPCA} $(n \times p)$ contains the drift component in the data. To eliminate the drift from the data, the last step is to subtract this drift from the data (equation 6.10)

$$Z^{CPCA} = Y - Y_d^{CPCA} \tag{6.10}$$

where Z^{CPCA} $(n \times p)$ is the corrected data set using the CPCA method.

6.4.3.5 CPCA + Median Fold Change

The method we propose consists of a two-step approach. Firstly, the data is equalised by removing the drift using CPCA and, in the second step, the data is normalised by

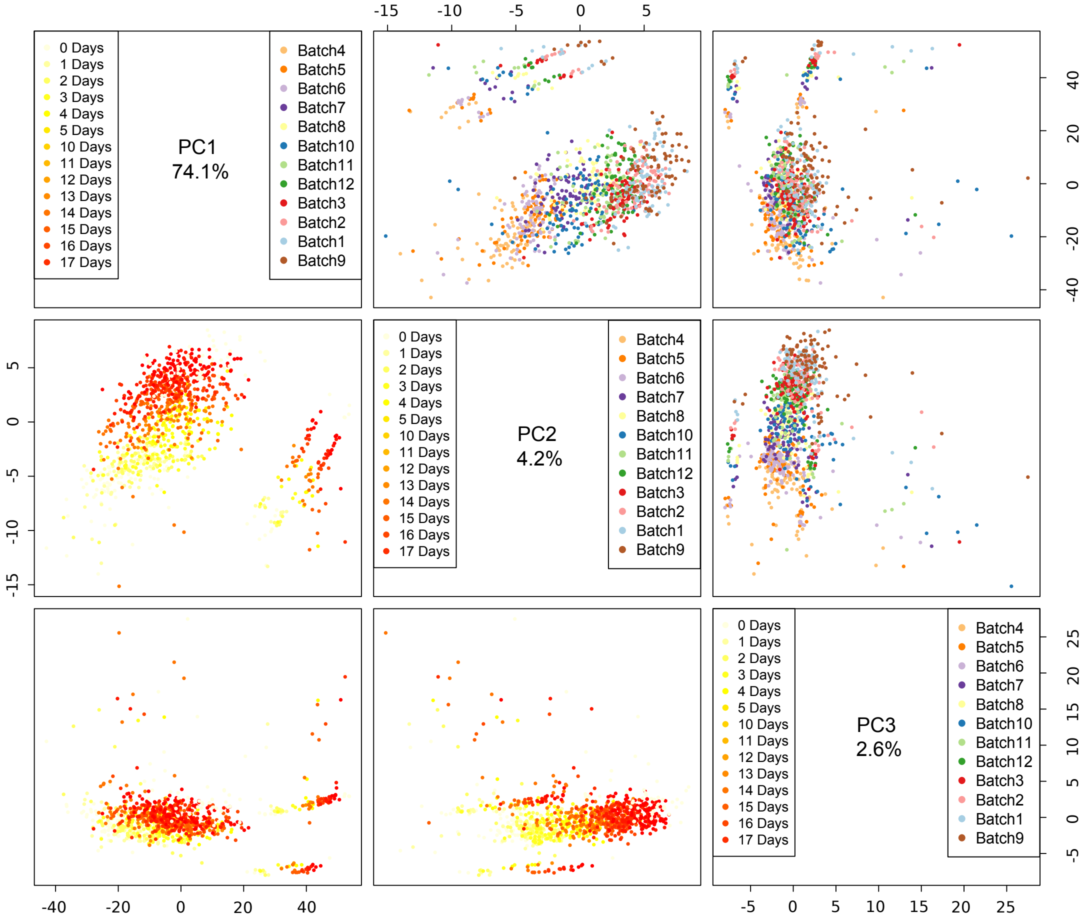


FIGURE 6.2: PCA Scoreplot of all the classes in the data. The three plots in the lower left show the effect of the time elapsed since the first sample was injected, whereas the plots in the top right refer to the batch of the sample. The numbers in the diagonal plots correspond to the variance captured by each PC. The order in the legend of the batches corresponds to the real injection order of the samples.

applying the Median Fold Change method. As the CPCA method was applied three times with different number of extracted CPCs ($Ncomps$ in previous subsection), the proposed method will be computed for the same number of components ($Ncomps = 1, 2, 3$).

6.4.4 Validation

From the class definition in section 6.4.1, it follows that a PCA score plot of all the classes should have the classes clearly separated in different clusters. We propose a quality measure for peak intensity drift correction methods based on the standard clustering internal measures Dunn and Silhouette for the QC classes in the principal plane (the plane explaining maximum variability of the data) score plot of all the classes (including the study class). The clustering technique used was k-means. The R package clValid was used to compute the quality indices [22, 23]. In general, the greater the Dunn and Silhouette indices, the better the clustering, meaning that the QC classes are more easily separable in the principal plane and that the intra-class variance is lower.

6.5 Results and Discussion

The top left plot in Figure 6.1 depicts a PCA score plot for the raw data using all the classes. The Figure shows that one of the main sources of variance is the interclass variability. The drift effect on the variance becomes clear when the same PCA score plots are coloured according to the injection time or the batch of the sample. This effect is shown in Figure 6.2. From this Figure, we conclude that there is a clear drift component (having different sources) that is causing an important drift of the QC classes and which, in all likelihood, affects the samples in the study class as well.

Table 6.1 contains the Dunn and Silhouette values for all the methods used, whereas Figure 6.1 depicts the PCA Scoreplots for the same methods. The CPCA+Median fold change method shows the highest clustering values (highest Dunn index when two components are removed and highest Silhouette index when one component is removed) and it has a slight advantage over the CPCA and the Median Fold Change methods.

From the mathematical formulation of the Dunn index, it is important to note that its measures might give low values when there are a small number of samples some distance from the cluster centre, even if all the other samples and classes are tightly clustered [24]. This might be the case here, since applying the CC method removing 2 PCs had a higher Dunn index than the CPCA+Median Fold Change removing 1 CPC, when the PCA score plots showed a lower drift clustering for the latter (compare Figure D.1 vs the bottom right plot of Figure 6.1).

The Silhouette index (Table 6.1) for the different CPCA methods applied suggests that the drift seems to be contained in just the first CPC, as the quality measures go down as more CPCs are removed from the data. Figure 6.3 depicts the origin of the variance removed by each of the three CPCs and the mean chromatogram of the QC samples. Whereas the first CPC shows a smooth variation along the retention time, the second

Table 6.1: Dunn and Silhouette values for all the tested methods. The CPCA and CC methods were tested removing one, two and three components (the number in brackets refers to the components subtracted from the data). The last three entries of the table correspond to sequentially using the CPCA with one CPC, and then the Median fold change method. The highest clustering indices are shown in bold.

Method / Index	Dunn	Silhouette
None	0.029	0.560
CPCA (1CPC)	0.159	0.749
CPCA (2CPC)	0.129	0.725
CPCA (3CPC)	0.051	0.680
ComBat	0.074	0.553
CC (1PC)	0.182	0.573
CC (2PC)	0.249	0.600
CC (3PC)	0.209	0.566
Median Fold Change	0.171	0.719
CPCA (1CPC)+Median	0.208	**0.794**
CPCA (2CPC)+Median	**0.344**	0.690
CPCA (3CPC)+Median	0.101	0.631

and third CPCs show abrupt changes in their values, especially close to major chromatographic peaks. In some of these peaks, the values of the second and third CPCs have a zigzag behaviour, suddenly going from negative to positive values or vice versa. This behaviour suggests that the variance captured by these two CPCs corresponds to not having the chromatograms perfectly aligned, meaning that the peaks in the chromatograms are not fully coincident across the samples. This fact further reinforces the hypothesis that the drift is contained in just the first CPC. Assuming this hypothesis, it can be noted from Figure 6.3 that the drift of the data is not higher close to the chromatographic peaks, apparently quite the opposite, meaning that the drift affects the baselines of the chromatograms more than their peaks.

The CC method corrects some of the drift in the data although a large drift component is still to be found in the data (Figure 6.1). The larger Dunn index value for the CC method as compared to the raw data value is evidence for the drift correction (Table 6.1). However, this improvement is not validated by the Silhouette index which remains practically unchanged as compared to the raw Silhouette value regardless of the number of components removed.

The ComBat method seems not to be the most suitable method for correcting LC/MS metabolomic data despite being used widely and successfully in the field of gene expression and methylation data. Although it corrects some batch effects in the study samples,

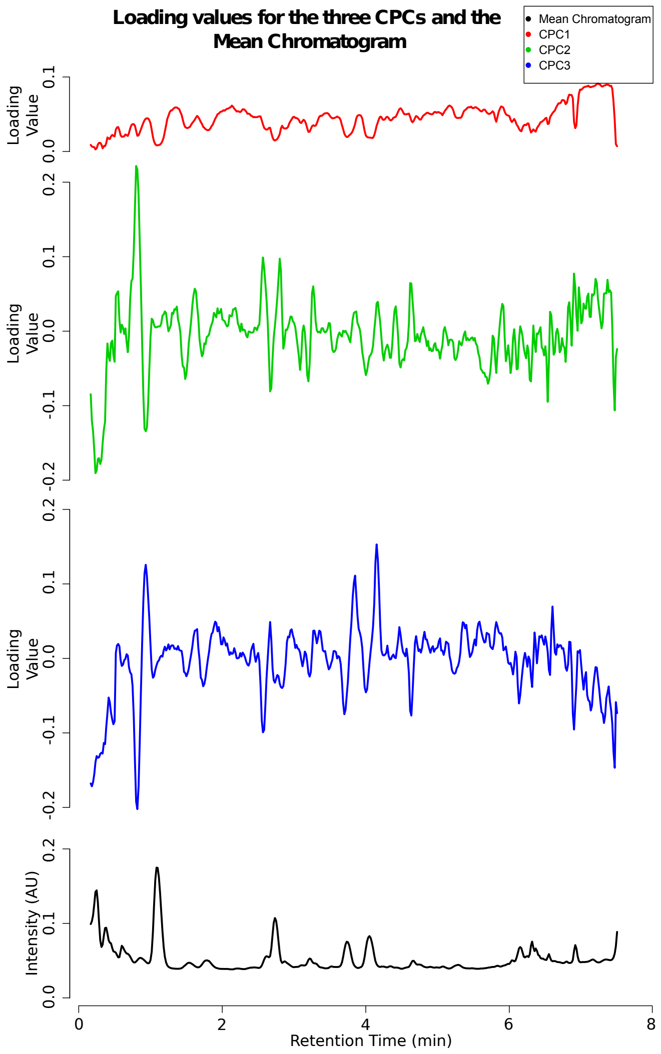


FIGURE 6.3: CPCs loading values and mean chromatogram for the QC samples. The chromatogram is plotted in arbitrary units. The loadings of the CPCs correspond to the columns of the V matrix shown in equation (6.8)

the batch effects are still important in the QC classes after the correction (see circles A-G for some *spikes* and *water* samples compared to the H circle containing some study samples in Figure D.2). Furthermore, the resultant PCA score plot (lower left plot of Figure 6.1) for the ComBat correction suggests that some between-variance component was removed in the correction process as the different classes are closer than for the raw data.

The Median Fold Change method considerably improves both the Dunn and the Silhouette indices. A visual inspection of the resulting PCA score plot for the Median Fold Change method confirms this improvement (Figure 6.1). Nevertheless, the PCA score plot also shows that the *spikes* and *water* classes have similar shapes and these long shapes turn out to be caused by residual uncorrected drift effects (Figure D.3). This fact suggests that, as the Median Fold Change method normalises the data without specifically trying to remove the drift, there may still be a source of variance in the data caused by the drift of the experimental device. On the other hand, because the methods based in the CPCA approach (CPCA and CPCA+Median Fold Change methods) are developed to model the drift direction, their resultant datasets show less residual drift in their corresponding principal plane score plot (Figures D.4 and D.5 for the CPCA and the CPCA+Median Fold Change methods respectively).

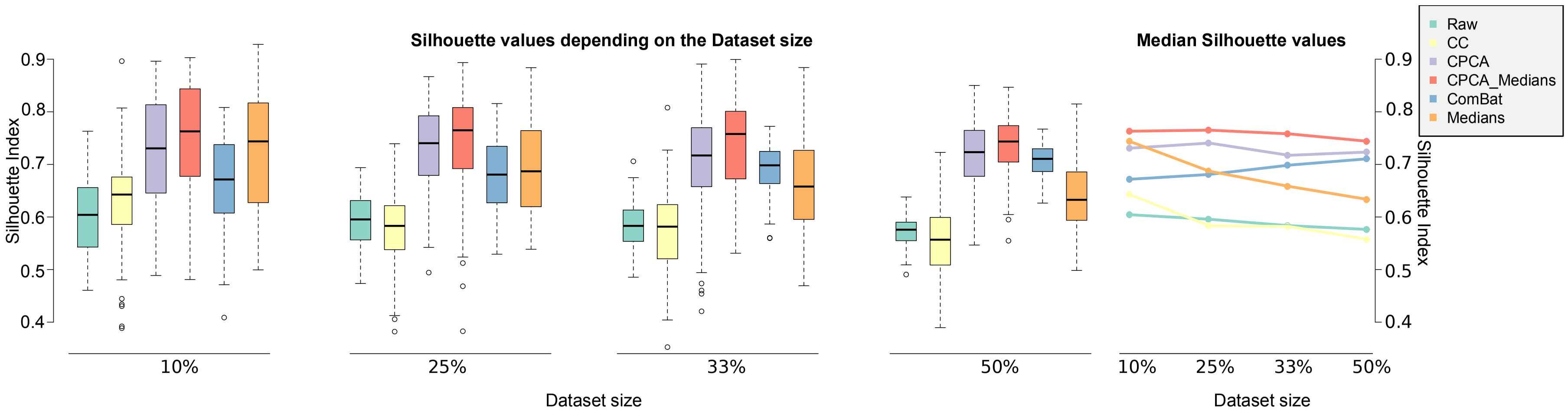


FIGURE 6.4: PCA Scoreplot of all the classes in the data. The three plots in the lower left show the effect of the time elapsed since the first sample was injected, whereas the plots in the top right refer to the batch of the sample. The numbers in the diagonal plots correspond to the variance captured by each PC.

To evaluate the performance of the tested methods for different dataset sizes, we have run a random subsampling stage taking the 10%, 25%, 33% and 50% (taking 100 iterations per subsample). For each subset, we applied the proposed methodology in order to test the compensation effect given dataset size and to find an estimate on the variability of this effect. We computed the Silhouette index for each drift correction method as described in Section D.1. Figure 6.4 depicts the Silhouette values for all the methods and dataset sizes. The Figure shows that the CPCA + Medians method has the highest mean Silhouette values for all the tested values. To measure how the performance of the methods is modified as function of the dataset size, we have fitted a linear model using size as a cofactor and computed an ANOVA test. Table D.1 contains the values of the slopes, the standard errors and the p-values of the ANOVA tests for all the methods. Results show that the Median Fold Change and the CC methods suffer from performance for large datasets. On the other hand, the ComBat method improves its correction as data availability increases. This last result is probably due to a better estimation of the batch components when having larger sample sizes. The Table D.1 also shows that the performance of the CPCA and CPCA + Medians methods is insensitive to the considered dataset size. This suggests that the mathematical approach taken, where the drift component is extracted from a multi-class QC variance analysis is more resilient to the sample size variations.

Overall, in the context of LC/MS drift correction, the proposed two-step methodology shows better clustering properties of the QC samples for large metabolomic studies than the median fold change method. The method also shows a robust behaviour under small sample size conditions. Furthermore, unlike the median fold change method, the two-step method is able to capture intensity drifts that covariate with the retention time.

6.6 Conclusions

Applying the combined method CPCA and the median fold change, results in a data set that contains less drift effects than the data set corrected solely by the median fold change, and the QC class separability of the data set is higher than if just the CPC method is applied.

Results show that, among all the methods tested to normalise the LC/MS metabolomic data, the best approach is to use a two-step method in which the first step is to remove the drift by finding the drift direction in a multivariate space using a CPCA approach. The second step, based on performing a median fold change to account for differences in concentration results, improved between-class separability and hence resulted in a better-normalised data set overall. As far we know, this is the first time that this kind of approach has been applied in the metabolomics context. Applications such as the one proposed open the possibility of carrying out large epidemiological LC/MS metabolomics experiments with high guarantee of the control the quality of the acquisition data step.

Bibliography

[1] Oliver Fiehn, Bruce Kristal, Ben van Ommen, Lloyd W Sumner, Susanna-Assunta Sansone, Chris Taylor, Nigel Hardy, and Rima Kaddurah-Daouk. Establishing reporting standards for metabolomic and metabonomic studies: a call for participation. *Omics : a journal of integrative biology*, 10(2):158–163, 2006. ISSN 1536-2310. doi: 10.1089/omi.2006.10.158.

[2] Zhang Aihua, Sun Hui, Wang Ping, Han Ying, and Wang Hijun. Modern analytical techniques in metabolomics analysis. *Analyst*, 137:293–300, 2012.

[3] Lu Xin, Zhao Xinjie, Bai Changmin, Zhao Chunxia, Lu Guo, and Xu Guowang. Lc-ms-based metabonomics analysis. *Journal of Chromatography B*, 866:64–76, 2007.

[4] San-Yuan Wang, Ching-Hua Kuo, and Yufeng J. Tseng. Batch normalizer: A fast total abundance regression calibration method to simultaneously adjust batch and injection order effects in liquid chromatography/time-of-flight mass spectrometry-based metabolomics data and comparison with current calibration methods. *Analytical Chemistry*, 85(2):1037–1046, 2013.

[5] Lyle Burton, Gordana Ivosev, Stephen Tate, Gary Impey, Julie Wingate, and Ron Bonner. Instrumental and experimental effects in LC-MS-based metabolomics. *Journal of chromatography. B, Analytical technologies in the biomedical and life sciences*, 871(2):227–235, 2008. ISSN 15700232. doi: 10.1016/j.jchromb.2008.04.044.

[6] Kirill A Veselkov, Lisa K Vingara, Perrine Masson, Steven L Robinette, Elizabeth Want, Jia V Li, Richard H Barton, Claire Boursier-Neyret, Bernard Walther, Timothy M Ebbels, and et al. Optimized preprocessing of ultra-performance liquid chromatography/mass spectrometry urinary metabolic profiles for improved information recovery. *Analytical Chemistry*, 83(15):5864–5872, 2011.

[7] Warwick B Dunn, David Broadhurst, Paul Begley, Eva Zelena, Sue Francis-McIntyre, Nadine Anderson, Marie Brown, Joshau D Knowles, Antony Halsall, John N Haselden, and et al. Procedures for large-scale metabolic profiling of serum and plasma using gas chromatography and liquid chromatography coupled to mass spectrometry. *Nature Protocols*, 6(7):1060–1083, 2011. URL http://www.ncbi.nlm.nih.gov/pubmed/21720319.

[8] Rafael Llorach, Mireia Urpi-Sarda, Olga Jauregui, Maria Monagas, and Cristina Andres-Lacueva. An LC-MS-based metabolomics approach for exploring urinary metabolome modifications after cocoa consumption. *Journal of proteome research*, 8(11):5060–5068, 2009. ISSN 1535-3907. doi: 10.1021/pr900470a.

[9] Marko Sysi-Aho, Mikko Katajamaa, Laxman Yetukuri, and Matej Oresic. Normalization method for metabolomics data using optimal selection of multiple internal standards. *BMC Bioinformatics*, 8, 2007.

[10] W. Evan Johnson, Cheng Li, and Ariel Rabinovic. Adjusting batch effects in microarray expression data using empirical Bayes methods. *Biostatistics (Oxford, England)*, 8(1):118–27, 2007. URL http://www.ncbi.nlm.nih.gov/pubmed/16632515.

[11] Jeffrey T. Leek, W. Evan Johnson, Hilary S. Parker, Andrew E. Jaffe, and John D. Storey. The sva package for removing batch effects and other unwanted variation in high-throughput experiments. *Bioinformatics*, 28(6):882–883, 2012.

[12] Kai Chen, Dajiang Wu, Xiaodong Zhu, Haijian Ni, Xianzhao Wei, Ningfang Mao, Yang Xie, Yunfei Niu, and Ming Li. Gene expression profile analysis of human intervertebral disc degeneration. *Genetics and Molecular Biology*, 36:448 – 454, 00 2013. ISSN 1415-4757. URL http://www.scielo.br/scielo.php?script=sci_arttext&pid=S1415-47572013000300021&nrm=iso.

[13] Harry G. Leitch, Kirsten R. McEwen, Aleksandra Turp, Vesela Encheva, Tom Carroll, Nils Grabole, William Mansfield, Buhe Nashun, Jaysen G. Knezovich, Austin Smith, M. Azim Surani, and Petra Hajkova. Naive pluripotency is associated with global DNA hypomethylation. *Nat Struct Mol Biol*, 20(3):311–316, March 2013. ISSN 1545-9985. doi: 10.1038/nsmb.2510. URL http://dx.doi.org/10.1038/nsmb.2510.

[14] Tom Artursson, Tomas Eklöv, Ingemar Lundström, Per Mårtensson, Michael Sjöström, and Martin Holmberg. Drift correction for gas sensors using multivariate methods. *Journal of Chemometrics*, 14(5-6):711–723, 2000. doi: 10.1002/1099-128X(200009/12)14:5/6\%3C711::AID-CEM607\%3E3.0.CO;2-4. URL http://www3.interscience.wiley.com/cgi-bin/abstract/73502597/ABSTRACT.

[15] A Ziyatdinov, S Marco, A Chaudry, K Persaud, P Caminal, and A Perera. Drift compensation of gas sensor array data by common principal component analysis. *Sensors and Actuators B: Chemical*, 146(2):460–465, 2010. URL http://linkinghub.elsevier.com/retrieve/pii/S0925400509008995.

[16] C A Smith, E J Want, G O'Maille, R Abagyan, and G Siuzdak. XCMS: processing mass spectrometry data for metabolite profiling using nonlinear peak alignment, matching, and identification. *Analytical Chemistry*, 78(3):779–787, 2006. ISSN 00032700. doi: 10.1021/ac051437y. URL http://pubs3.acs.org/acs/journals/doilookup?in_doi=10.1021/ac051437y.

[17] Mia Hubert, PJ Rousseeuw, and KV Branden. ROBPCA: a new approach to robust principal component analysis. *Technometrics*, pages 1–34, 2005. ISSN 0040-1706.

[18] Peter Filzmoser, Heinrich Fritz, and Klaudius Kalcher. *pcaPP: Robust PCA by Projection Pursuit*, 2013. URL http://CRAN.R-project.org/package=pcaPP. R package version 1.9-49.

[19] P Filzmoser and H Fritz, editors. *Exploring high-dimensional data with robust principal components.*, volume 1, 2007.

[20] Bernhard N. Flury. Common Principal Components in K Groups. *Journal of the American Statistical Association*, 79(388):892–898, 1984. ISSN 01621459.

URL http://www.jstor.org/stable/2288721$\delimiter"026E30F$nhttp://www.jstor.org/stable/pdfplus/2288721.pdf.

[21] Nickolay T. Trendafilov. Stepwise estimation of common principal components. *Computational Statistics & Data Analysis*, 54(12):3446 – 3457, 2010. ISSN 0167-9473. doi: http://dx.doi.org/10.1016/j.csda.2010.03.010. URL http://www.sciencedirect.com/science/article/pii/S016794731000112X.

[22] Guy Brock, Vasyl Pihur, Susmita Datta, and Somnath Datta. clvalid: An r package for cluster validation. *Journal of Statistical Software*, 25(4):1–22, 3 2008. ISSN 1548-7660. URL http://www.jstatsoft.org/v25/i04.

[23] Guy Brock, Vasyl Pihur, Susmita Datta, and Somnath Datta. *clValid: Validation of Clustering Results*, 2011. URL http://CRAN.R-project.org/package=clValid. R package version 0.6-4.

[24] J. C. Dunn. A Fuzzy Relative of the ISODATA Process and Its Use in Detecting Compact Well-Separated Clusters, 1973. ISSN 0022-0280.

Chapter 7

Correcting time drift effects in Liquid Chromatography using a new non-linear model-based methodology

7.1 Abstract

The statistical control of variance is highly desirable when analysing Liquid Chromatography data. Among the most unwanted sources of variability is instrumental drift which greatly interferes with the data processing by inducing broader chromatographic peaks leading to noisier measures. Most of the variability induced by the chromatographic columns is compensated for by aligning the chromatographic signals. Many of the currently available alignment methods are based on piecewise warping of the data. These powerful curve aligners however, carry a risk of overcorrection and of generating artefacts for datasets which show large drifts. The aim of this paper is to introduce a new methodology to correct the chromatographic drift in LC-MS metabolomic data. With this in mind, our methodology includes applying a parametric model that corrects non-linear drift components which show systematic behaviour over the samples. Using two different datasets, this new methodology (named H-Cor) is compared to three other alignment methods: Locally Weighted Scatterplot Smoothing (LOESS), piecewise linear regression and Parametric Time Warping (PTW). Kurtosis and correlation metrics for the five most intense chromatographic peaks are used as quality measures. Our results show that the H-Cor methodology has a better performance than all others when the drifts show large non-linearities over the retention time. Additionally, the LOESS and piecewise linear warping algorithms generate artefacts in the data when there are large retention time drifts due to overcorrection. This behaviour suggests that a model-based methodology like H-Cor generalises better in controlling the chromatographic drifts generated in Liquid Chromatography.

Availability: The method is available as an R package at http://b2slab.upc.edu/software-and-downloads/retention-time-drift-correction/

7.2 Introduction

The application of LC-MS in large metabolomic studies carries certain risks — among others, the creation of artefacts related to shifts in the intensity and retention time. Retention time drift is, along with possible contamination by background chemicals and the stability of the column, one of the main challenges in LC-MS-based large metabolomic studies. In fact, maintaining excellent retention time repeatability is essential. A lack of consistent repeatability compromises the peak extraction procedure, yielding false peak detection of ar tefacts in the peak covariability structure [1]. The standard LC-MS data workflow includes a peak alignment stage whose aim is to correct the time retention shifts [2]. An additional reason for performing this alignment stage is that, in Liquid Chromatography, the signal is often converted to a data matrix as a precursor to analysis [3]. To perform such a mathematical transformation, it is necessary to ensure a common time scale over all the samples (i.e. the chromatographic region of a sample corresponds to the same chromatographic region of all the other samples). If no alignment is applied, given two or more samples, one might find a chromatographic peak of the same chemical in different chromatographic regions.

Many algorithms have been proposed to perform chromatographic alignment. These algorithms have been extensively reviewed and some comparisons between them have previously been performed [4, 5]. Alignment algorithms may be classified in two different categories, depending on whether or not the mass peaks are detected before the chromatographic alignment [6, 7, 8, 9, 10]. Among the methods that do not need to perform the mass peak detection stage, are piecewise warping algorithms such like Correlation Optimised Warping (COW) [6, 7] or Parametric Time Warping (PTW) [8]. COW uses an aggregated correlation as a quality measure in the optimisation, whereas PTW adopts a parametric approach using the sum of squared residuals as quantitative metric. Among the techniques that do require prior detection of mass peaks, one of the most widely used is Locally Weighted Scatterplot Smoothing (LOESS) [11]. In this algorithm, only data close to a regression function evaluation value, x, are used [7]. In the LOESS approach, it is common to use either quadratic or cubic polynomials as weight functions. As a result, in the piecewise warping algorithms, pieces of the chromatogram are stretched or compressed following the optimisation of an objective function. This warping approach might cause artefacts in the aligned chromatograms or produce models with over-fitting depending on the binning parameter [12].

When performing chromatographic alignment, to avoid overcorrection of the signal due to overfitting, we propose a new methodology which does not use a piecewise approach. This methodology does not require prior detection of the peak masses.

7.3 Materials and Methods

The proposed methodology is based on the assumption that, for Liquid Chromatography, the chromatogram recording time is short enough (7-8 minutes), that highly complex time drift shapes are not observed. Under these circumstances, our methodology includes the fitting of a non-linear model to perform the drift correction. To test its performance, this new methodology was compared to three other alignment methods: LOESS, PTW and Piecewise Linear for two different datasets having different drift patterns. Dataset 1 has a less severe retention time drift than Dataset 2. LOESS and piecewise linear were applied using the R package XCMS [7] whereas PTW was applied using the R package ptw [13] (see Section S1 for package utilization details).

7.3.1 Experimental Section

The samples were analysed by a liquid chromatograph coupled with a hybrid quadrupole time-of-flight device (LC-q-TOF, AB/MDS Sciex) in positive mode using the protocol proposed by Tulipani et al. [14]. LC was performed in HPLC Agilent using a RP 18 Luna column (50 X 2.0mm, 5m), with a sample injection volume of 15 μL. A linear gradient elution was performed consisting of [A] Milli-Q water 0.1% HCOOH (v/v) and [B] acetonitrile 0.1% HCOOH (v/v).The gradient elution (v/v) of [B] was: (time, min; B, %): (0, 1), (4, 20), (6, 95), (7.5, 95),(8, 1), (12, 1). Q-TOF spray parameters were set as previously described [15] and full data acquisition was performed, scanning from 70 to 700 m/z. The TOF was calibrated with reserpine (1 pmol/μL). LC-MS data were acquired in random order to avoid possible bias and the batches were equilibrated. Throughout all the analysis, data process quality control (QC) samples were analysed in order to monitor the stability and functionality of the system. Two different data sets of 300 samples were acquired using the same experimental protocol with different chromatographic columns.

7.3.2 H-Cor Model

Instead of dividing the chromatograms in pieces and appling a warping algorithm on these pieces, we propose a new methodology based on a sample-wise parametric model. As stated in the last section, the sample drift is modelled as shown in (7.1). This model is fitted for each sample separately and following the procedure detailed in Section S2, we obtain a dataset where the retention time drift component has been removed.

$$\Delta rt' = \frac{a}{rt + b}; \quad b \geq 0.5 \tag{7.1}$$

In expression (7.1), the underlying assumption is that the highest signal drift is located in the lowest retention times, while this drift vanishes for high retention times i.e. $rt \to \infty \Rightarrow \Delta rt' {\to} 0$

7.3.3 Validation

In order to evaluate the performance of the algorithms, two quality measures were defined: peak correlation and peak kurtosis (see Section S3 for details). Both quality measures were used in the five highest chromatographic regions to evaluate the effects of the alignments over the whole retention time. Peaks having a smaller width and larger height are expected to have larger kurtosis, leading to a better peak definition in terms of its retention time interval. On the other hand, a higher correlation value would lead to better peak alignment.

7.4 Results and Discussion

Figure 7.1 depicts the chromatographic profiles gathered for the samples of Dataset 1 whereas Figure E.1 shows a the same plot for Dataset 2. A large temporal drift pattern is observed among the later samples shown in Figure 7.1. Qualitatively speaking, for each sample, this pattern seems to behave similarly across the retention time. This drift is larger for lower retention times, whereas the signal at high retention times seems unaffected by such drift behaviour.

The bottom-left plot of Figure 7.2 shows the average chromatogram for the Dataset 1. In this chromatogram, peaks labelled from A to E are the highest peaks over all the samples of the dataset. Peaks A, B and D correspond either to known endogenous metabolites or to a spike in metabolites (see Table E.1). In consequence, these three peaks will be found in all the samples of both datasets. Using the procedure detailed in Section S4, it is possible to obtain the peak drift measures and to characterise the correlation between their shifts. The plots located at the upper-right in Figure 7.2 summarise the results of the peak drift measures for Peaks A, B and D for the first dataset (see Figure E.2 to see the drifts of the five peaks). The peak drift measures of Peaks A, B and D for Dataset 2 are depicted in Figure E.3. We can assess the statistical significance of the correlation between the relative shifts for the different peaks by fitting a linear model and checking its p-values (see Table E.2) — we find a clear correlation. This peak drift correlation is not statistically significant for high retention times (Peak E). Similarly, positive correlation does not exist for high retention times (Figure 7.1). This pattern suggests an underlying non-linear component which extends over the retention time. The

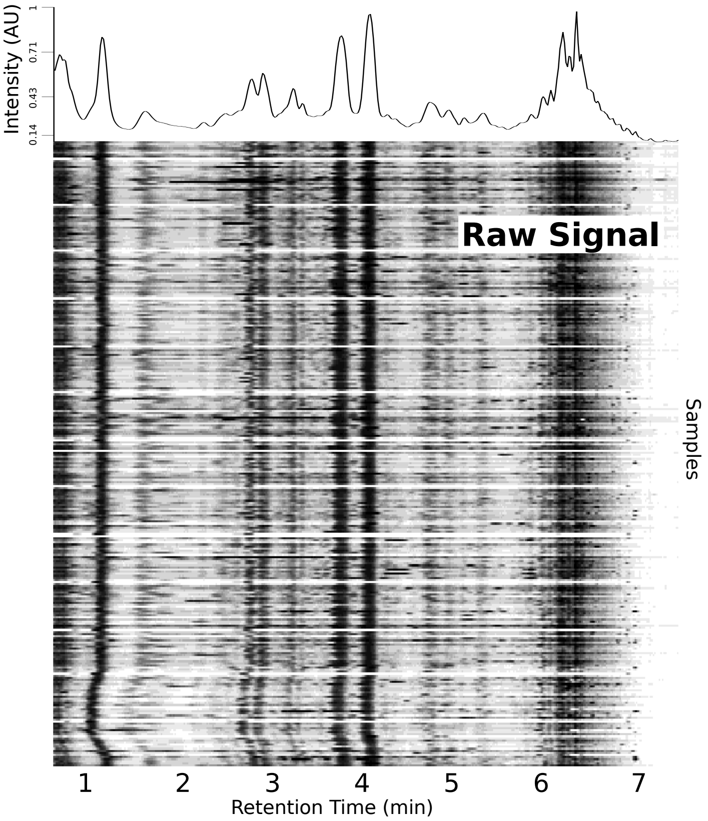


Figure 7.1: Raw chromatograms for the first dataset. Each row refers to a sample, whereas the columns are the retention time in minutes. Darker grey means higher signal intensity. The chromatogram at the top of the image corresponds to the average chromatogram of all the samples of the dataset.

evidence lead us to hypothesise that the observed time drift in Liquid Chromatography is not random, but decreases over retention time in some non-linear manner. This drift component gradually decreases with retention time and is not observed at large retention times. The proposed model, called the H-Cor model, assumes a model for these peak drift pattern features to correct the overall data drift and to align the chromatograms.

7.4.1 Dataset 1

Table 7.1 shows the values of the peak correlation and peak kurtosis respectively for Dataset 1. In this table, the LOESS method results were computed using two different span values (0.7 and 0.8, see Section S4 for details). From Table 7.1 it can be noted that, despite of the alignment method used, Peak D and especially Peak E have lower correlation values than the other peaks. This is evidence that the final region of the chromatograms contains the most noisy data, making this end region (including Peaks

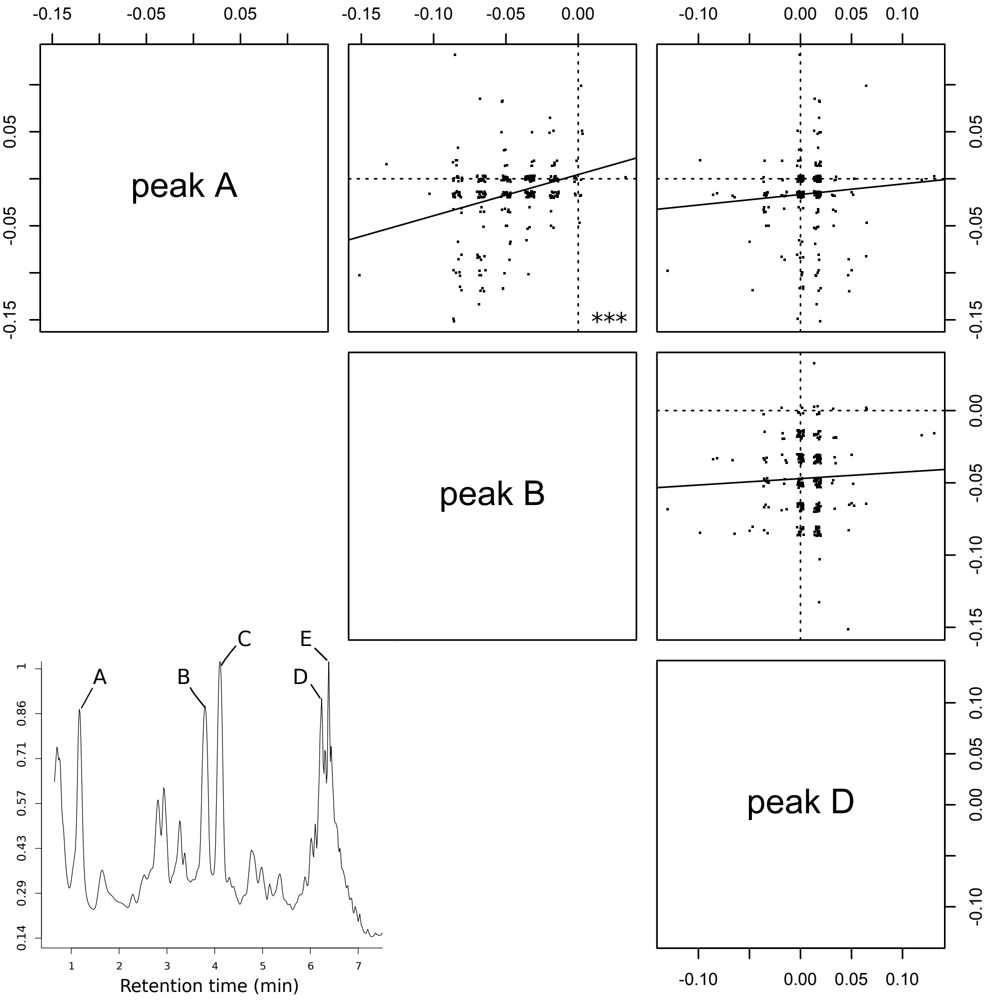


Figure 7.2: The top-right plots show the peak drifts of three chromatographic peaks using the raw data of the first dataset. The bottom-left image corresponds to the chromatogram of a single sample. The labelled chromatographic peaks point to high-intensity regions found in all samples. The asterisk symbol shows the statistical significance of the slope in the linear models following an ANOVA test. Figure E.2 shows the linear models for all five peaks labelled in the chromatogram and Table E.2 contains the p-values of the slopes.

D and E) less reliable in terms of drift correction. In consequence, it contains narrower peaks that are more difficult to align.

The high dependence of the LOESS method on the span parameter appears evident when comparing the entries for span=0.7 and span=0.8 in Table 7.1. Differences up to 0.186 in correlation Peak C) and up to 0.333 in kurtosis (Peak E) are observed. We deduce from these data that the optimal span parameter for this dataset is 0.7 over all higher correlation and kurtosis values.

H-Cor, LOESS and piecewise Linear methods seem to behave similarly for Peaks B and C, with comparable correlation and kurtosis values. However, the H-Cor method outperforms all others for Peak A, showing the highest correlation, 0.901, and kurtosis, 0.083, values. As expected, the H-Corr Method performs greater signal correction for lower retention time zones while this correction effect is lowered for high retention times.

Consequently, considering both the kurtosis and the correlation measures, the H-Cor method outperforms all other methods, showing the best alignment in Peak A with a gain of 0.087 in correlation and 0.316 in kurtosis as compared to the unaligned signal. These two gain values are also the maximum gain values when considering all the methods and peaks. LOESS with span=0.7 and Linear methods show similar correlation patterns whereas LOESS has better kurtosis values. Figure 7.3 depicts a comparison between the H-Cor and LOESS (span=0.7) alignments for the Dataset 1. In this Figure, the LOESS correction is not large enough to align the drift of the last samples of the dataset. The H-Cor method, on the other hand, showed better performance in aligning the samples. Figure E.4 shows the LOESS correction when the parameter span is 0.8. In this case, LOESS is able to solve the drift of the last samples. However, the method is overcorrecting as can be concluded from the saw-like pattern seen in the peaks (especially in Peak A). A similar behaviour is shown by the Linear method in Figure E.5, whereas the corrections performed by the PTW method seem not big enough (see Figure E.6). Note that the profile of Figure E.6 for the PTW method is different from the profile generated by the other methods. The reason is the use of a function to subtract the signal background in the PTW process. The effect of applying the H-Cor method to the peak drifts is depicted in Figure E.2. For the Peaks A, B and C, we see that the drift correlation has been lowered (Table E.2 contains the p-values of the linear regression shown in Figure E.2), whereas the method performs hardly any correction on Peaks D and E.

7.4.2 Dataset 2

Table 7.2 shows the correlation and kurtosis results for the second dataset. As in the first dataset, the quality measures for Peaks D and E are lower than the rest of the peaks regardless of the alignment method used, because these two peaks are noisier than the others. The Loess and H-Cor methods show similar performance for Peaks B and C,

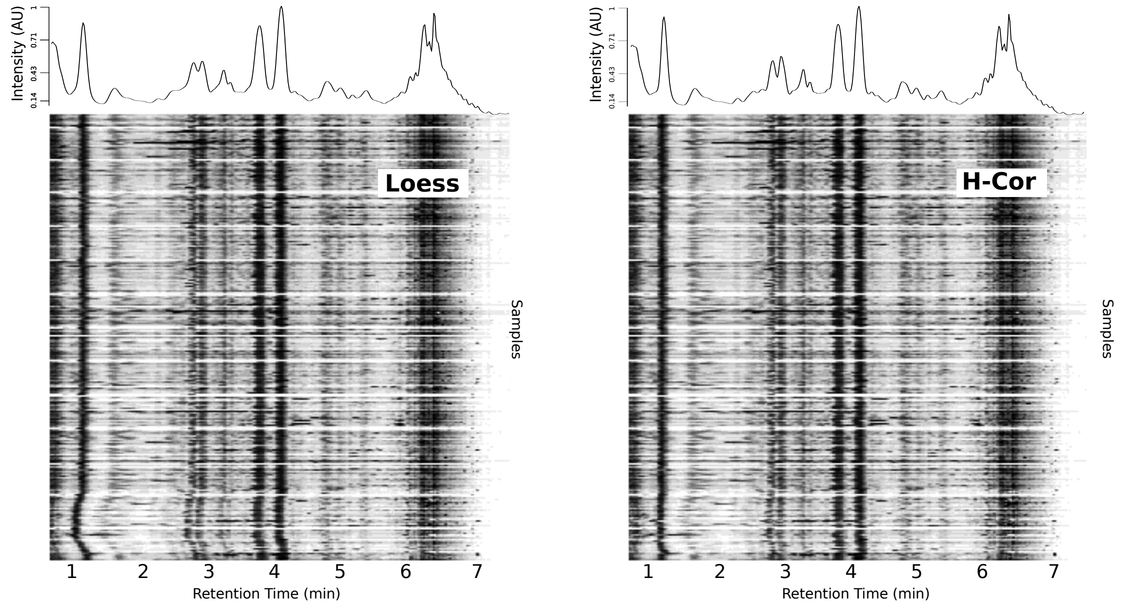


Figure 7.3: Aligned chromatograms for the first dataset using the H-Cor and LOESS methods. Each row refers to a sample, whereas the columns are the retention time in minutes. Darker grey means higher signal intensity. The chromatogram at the top of the image corresponds to the average chromatogram of all the samples of the dataset.

whereas the performance of the Linear method is slightly lower. A similar conclusion is drawn from Peak A where LOESS has the highest correlation value (0.937 for span=0.8), while H-Cor is the method with highest kurtosis (-0.079). However, in correlation terms, the LOESS method with span=0.8 performs badly for Peaks D and E, whereas the H-Cor kurtosis value for the Peak E is lower than any of the LOESS kurtosis values. Overall, in the case of Dataset 2, it seems that LOESS and H-Cor perform equally well and their results are somewhat better than those of the Linear Method. The PTW method showed the worst values when the four methods were compared, having the minimum correlation values for all the peaks and minimum kurtosis in four out of five peaks. Figure E.7 show the aligned chromatograms for LOESS (span=0.8) and H-Cor for the second dataset. Comparing both pictures, it appears that their performance is similar.

Table 7.1: Peak correlation values for all the alignment algorithms used for Dataset 1. The LOESS method uses two different span values.

	Correlation values					**Kurtosis values**				
Method	Peak A	Peak B	Peak C	Peak D	Peak E	Peak A	Peak B	Peak C	Peak D	Peak E
No Correction	0.814	0.849	0.907	0.714	0.446	-0.233	-1.624	-1.528	-1.468	-1.282
Linear	0.808	0.861	0.884	0.719	0.505	-0.308	-1.605	-1.528	-1.558	-1.489
PTW	0.769	0.795	0.820	0.603	0.182	-0.555	-1.673	-1.536	-1.389	-1.871
H-Cor	0.901	0.852	0.890	0.709	0.351	0.083	-1.615	-1.527	-1.448	-1.501
LOESS span=0.7	0.814	0.849	0.907	0.713	0.446	-0.239	-1.623	-1.529	-1.469	-1.215
LOESS span=0.8	0.798	0.853	0.721	0.681	0.316	-0.271	-1.612	-1.553	-1.638	-1.548

Table 7.2: Peak correlation values for all the alignment algorithms used for Dataset 2. The LOESS method uses two different span values.

	Correlation values					**Kurtosis values**				
Method	Peak A	Peak B	Peak C	Peak D	Peak E	Peak A	Peak B	Peak C	Peak D	Peak E
No Correction	0.865	0.867	0.909	0.573	0.642	-0.133	-1.578	-1.559	-1.350	-1.513
Linear	0.926	0.818	0.876	0.502	0.536	-0.117	-1.608	-1.569	-1.289	-1.460
PTW	0.782	0.678	0.663	0.482	0.140	-0.266	-1.621	-1.598	-1.233	-1.814
H-Cor	0.889	0.862	0.884	0.619	0.608	-0.079	-1.564	-1.561	-1.350	-1.609
LOESS span=0.7	0.865	0.867	0.909	0.574	0.641	-0.133	-1.581	-1.560	-1.293	-1.536
LOESS span=0.8	0.937	0.874	0.900	0.508	0.543	-0.093	-1.611	-1.557	-1.356	-1.485

Hence, for both datasets, the H-Cor methodology corrects the chromatography temporal drifts, performing best when the data shows high non-linear drifts over the retention time. Even when the dataset drifts are low, the H-Cor method does not seem to overcorrect the data.

7.5 Conclusions

In this paper, the H-Cor methodology has been introduced to correct signal retention time drifts in Liquid Chromatography metabolomics data. This alignment procedure has been compared to other widely used methods such as LOESS, PTW and piecewise linear fitting. In order to test the alignment of the chromatograms, two different datasets were used. One of the datasets showed large non-linear drift components over the retention time, whereas the other dataset showed small drift effects. To quantify the performance of the algorithms, we computed kurtosis and correlation metrics for the five highest chromatographic peaks for each method and dataset. The results of the alignment showed that H-Cor outperformed all other methods when there was a large non-linear drift component over the retention time. It displayed a similar performance to LOESS, without producing artefacts due to overcorrection, when there was no systematic drift in the samples. The LOESS and Linear methods showed overcorrection phenomena which led to a worse performance compared to H-Cor method in the first (large drift) dataset. When the methods were applied to the second (small drift) dataset, the performance of LOESS and H-Cor was similar and the Linear method was slightly worse, whereas PTW had the worst metrics. All these facts suggest that applying the H-Cor model and reducing the peak drift correlation constitutes a good strategy for reducing the overall drift of the data and for performing the chromatographic alignment stage in Liquid Chromatography.

Bibliography

[1] Helen G. Gika, Georgios A. Theodoridis, Robert S. Plumb, and Ian D. Wilson. Current practice of liquid chromatography–mass spectrometry in metabolomics and metabonomics. *Journal of Pharmaceutical and Biomedical Analysis*, 87(0):12 – 25,

2014. ISSN 0731-7085. doi: http://dx.doi.org/10.1016/j.jpba.2013.06.032. Review Papers on Pharmaceutical and Biomedical Analysis 2013.

[2] Beata Walczak and Wen Wu. Fuzzy warping of chromatograms. *Chemometrics and Intelligent Laboratory Systems*, 77(1-2):173–180, 2005.

[3] Micha? Daszykowski and Beata Walczak. Use and abuse of chemometrics in chromatography. *TrAC Trends in Analytical Chemistry*, 25(11):1081–1096, 2006.

[4] A M Van Nederkassel, M Daszykowski, P H C Eilers, and Y Vander Heyden. A comparison of three algorithms for chromatograms alignment. *Journal of Chromatography A*, 1118(2):199–210, 2006.

[5] Eva Lange, Ralf Tautenhahn, Steffen Neumann, and Clemens Gröpl. Critical assessment of alignment procedures for lc-ms proteomics and metabolomics measurements. *BMC Bioinformatics*, 9(1):375, 2008.

[6] Niels-Peter Vest Nielsen, Jens Michael Carstensen, and J¿rn Smedsgaard. Aligning of single and multiple wavelength chromatographic profiles for chemometric data analysis using correlation optimised warping. *Journal of Chromatography A*, 805 (1-2):17–35, 1998.

[7] C A Smith, E J Want, G O'Maille, R Abagyan, and G Siuzdak. XCMS: processing mass spectrometry data for metabolite profiling using nonlinear peak alignment, matching, and identification. *Analytical Chemistry*, 78(3):779–787, 2006. ISSN 00032700. doi: 10.1021/ac051437y.

[8] Paul H C Eilers. Parametric time warping. *Analytical Chemistry*, 76(2):404–411, 2004.

[9] Zhongqi Zhang. Retention time alignment of lc/ms data by a divide-and-conquer algorithm. *Journal of the American Society for Mass Spectrometry*, 23(4):764–772, 2012.

[10] Xiaoli Wei, Xue Shi, Seongho Kim, Li Zhang, Jeffrey S. Patrick, Joe Binkley, Craig McClain, and Xiang Zhang. Data preprocessing method for liquid chromatography-mass spectrometry based metabolomics. *Analytical Chemistry*, 84(18):7963–7971, 2012. doi: 10.1021/ac3016856.

[11] W S Cleveland. Lowess: A program for smoothing scatterplots by robust locally weighted regression. *American Statistician*, 35(1):54, 1981.

[12] Katharina Podwojski, Arno Fritsch, Daniel C. Chamrad, Wolfgang Paul, Barbara Sitek, Kai Stühler, Petra Mutzel, Christian Stephan, Helmut E. Meyer, Wolfgang Urfer, Katja Ickstadt, and Jörg Rahnenführer. Retention time alignment algorithms for lc/ms data must consider non-linear shifts. *Bioinformatics*, 25(6):758–764, 2009. doi: 10.1093/bioinformatics/btp052.

[13] Tom G. Bloemberg, Jan Gerretzen, Hans J. P. Wouters, Jolein Gloerich, Maurice van Dael, Hans J. C. T. Wessels, Lambert P. van den Heuvel, Paul H. C. Eilers, Lutgarde M. C. Buydens, and Ron Wehrens. Improved parametric time warping for proteomics. *Chemometrics and Intelligent Laboratory Systems*, 104(1):65–74, 2010.

[14] Sara Tulipani, Rafael Llorach, Olga Jáuregui, Patricia López-Uriarte, Mar Garcia-Aloy, Mònica Bullo, Jordi Salas-Salvadó, and Cristina Andrés-Lacueva. Metabolomics Unveils Urinary Changes in Subjects with Metabolic Syndrome following 12-Week Nut Consumption. *Journal of Proteome Research*, 2011. ISSN 15353907. doi: 10.1021/pr200514h.

[15] Rafael Llorach, Mireia Urpi-Sarda, Olga Jauregui, Maria Monagas, and Cristina Andres-Lacueva. An LC-MS-based metabolomics approach for exploring urinary metabolome modifications after cocoa consumption. *Journal of proteome research*, 8(11):5060–5068, 2009. ISSN 1535-3907. doi: 10.1021/pr900470a.

Chapter 8

Publications

During the development of this thesis there were produced a set of publications in national and international conferences as well as scientific papers in indexed journals. It was also developed a set of R packages coding the methods developed in this thesis.

8.1 Indexed Journal Papers

- Francesc Fernández-Albert, Rafael Llorach, Cristina Andres-Lacueva and Alexandre Perera. **Peak Aggregation as an Innovative Strategy for Improving the Predictive Power of LC-MS Metabolomic Profiles.**
  Analytical Chemistry, 2014, 86 (5), pp 2320–2325. DOI: 10.1021/ac403702p

- Francesc Fernández-Albert, Rafael Llorach, Cristina Andres-Lacueva and Alexandre Perera. **An R package to analyse LC/MS metabolomic data: MAIT (Metabolite Automatic Identification Toolkit).**
  Bioinformatics (2014) 30 (13): 1937-1939. DOI: 10.1093/bioinformatics/btu136

- Francesc Fernández-Albert, Rafael Llorach, Mar Garcia-Aloy, Andrey Ziyatdinov, Cristina Andres-Lacueva and Alexandre Perera. **Intensity drift removal in LC/MS metabolomics by Common Variance Compensation.**
  Bioinformatics (2014). DOI: 10.1093/bioinformatics/btu423

- Francesc Fernández-Albert, Rafael Llorach, Cristina Andres-Lacueva and Alexandre Perera. **Correcting time drift efects in Liquid Chromatograpy using**

a new non-linear model-based methodology.
Metabolomics (Submitted)

8.2 Conference Papers

- F.Fernández, A.Perera-Lluna. **Methods and Tools for Liquid Chromatography/Mass Spectrometry in Metabolomics.**
  I Conference on Bioinformatics and Computational Biology.
  Barcelona (Spain) 11/13

- F.Fernández-Albert, S. Kanaan-Izquierdo, Alexandre Perera-Lluna. **Liquid Cromatography / Mass Spectrometry Analysis Methods in Metabolomics**
  7th Annual Conference CIBER-BBN.
  Málaga (Spain) 11/13

- F.Fernández, R.Llorach, C.Andrés-Lacueva, A.Perera-Lluna. **Un nuevo algoritmo para el análisis de estudios de nutrimetabolómica basados en LC-MS.**
  XXIX Conference of the Spanish Society of Biomedical Engineering.
  Cáceres (Spain) 11/11

- F.Fernández, A.Perera-Lluna, C.Andrés-Lacueva, R.Llorach-Asunción. **A new computational assisted bioinformatics workflow to identify polyphenols biotransformation markers on LC-MS based nutrimetabolomics studies.**
  5th International Conference of Polyphenols and Health.
  Sitges (Spain) 10/11

- F.Fernández, A.Perera-Lluna, C.Andrés-Lacueva, R.Llorach-Asunción. **A new computational assisted bioinformatics workflow for the comprehensive analysis of LC/MS based nutrimetabolomic studies.** I International Workshop in Metabolomics.

Bilbao (Spain) 09/11

8.3 Computational Tools and Packages developed

- HCor: R package containing the model-based method to correct non-linear drift effects in the rt (Chapter 7). Available at `http://b2slab.upc.edu/software-and-downloads/retention-time-drift-correction/`.

- intCor: R package containing several intensity drift correction methods. These methods are described in Chapter 6. The package is available at `http://b2slab.upc.edu/software-and-downloads/intensity-drift-correction/` and its vignette can be found in this document at Appendix C

- MAIT: R package that allows to perform end-to-end statistical analysis of LC/MS Metabolomic data. It is described in Chapter 5 and its vignette is found in Appendix B. It is found at `http://b2slab.upc.edu/software-and-downloads/metabolite-automatic-identification-toolkit/`.

- pagR: R package including the peak aggregation techniques described in Chapter 4. It is currently delivered on-demand.

Chapter 9

Results and Conclusions

9.1 Summary of the Results

The effect of applying peak aggregation techniques to LC/MS datasets was evaluated through a repeated random sub-sampling cross-validation stage using a 4-class dataset. We evaluated 4 different peak aggregation techniques and compared them to the situation in which no peak aggregation technique was applied. These peak aggregation techniques were the Spectral Mean method, the Maximum Peak Method, the PCA Decomposition methods and the NMF reduction method. 300 iterations of such classification stage were run using two different classifiers (PLSDA and SVM) and it was computed the classification ratio for each case as quality metrics of the peak aggregation technique. The results show an improvement of the classification ratio compared to not using such approach (differences found when a Fisher's LSD test was run) regardless of which peak aggregation technique was used. The improvement of the classification ratio for the tested data was in a range between 14% and 18%. We found that the PCA Decomposition and the NMF Reduction showed equal performance (no differences were found in a Fisher's LSD test) and they showed larger improvement in the classification ratio compared to the Spectral Mean and the Maximum Peak methods. The peak aggregation showing worse performance was the Maximum Peak method with an improvement of 15.9% (for PLSDA) and 13.8% (for SVM), whereas the improvement of the NMF and PCA was between 18.3% (for PLSDA) and 16.7% (for SVM). All the peak aggregation methods were coded and implemented in an R package called pagR.

We developed an R package called MAIT to perform end-to-end LC/MS Metabolomic data analysis. The MAIT workflow is depicted in Figure 5.1. Section B.7 contains a detailed example using MAIT on the the data distributed through the R package faahKO. In this example it is performed and end-to-end analysis and it is put especial emphasis on how to define a customised data analysis including the definition of user-defined databases or annotation files. It is also shown how MAIT can use an external peak data set computed through any other available computational tool to perform its data analysis, or how the user can define its own statistical test and apply it on a real dataset.

When evaluating the variance of the QC classes, we found that drift effects may be important in large LC/MS Metabolomics data. These effects are evident from Figure 6.2 in which is depicted a PCA Scoreplot of three QC classes (labeled as Spikes, Water and Reference in Figure 6.1) and the class containing the biological variability (labelled as Sample in Figure 6.1). Figure 6.2 depicts how the time elapsed since the first sample injection is an important source of variance despite not being biologically meaningful. We proposed a two step method based on a CPCA drift removal and a median fold change to normalise the LC/MS data. 4 different methods were tested and compared to the proposed method: CC, CPCA, ComBat method and median fold change. The results of the drift corrections were evaluated using the Dunn and Silhouette clustering indices on the QC classes as a quality metric. These measures were computed for the QC classes in their projection on their PC1-PC2 plane. The proposed two-step method gave the highest score for both quality metrics, when removing one CPC (Silhouette equal to 0.794) and two CPCs (Dunn equal to 0.344). Furthermore, a random subsampling stage with different number of samples (10%, 25%, 33% and 50% of the original dataset, taking 100 iterations for each size) were run using all the methods and computing the Silhouette index values. The results shown that the two step method had the highest median value regardless of the dataset size (Figure 6.4) and that its performance is the same regardless of the dataset size (p-value not significant in a t-test).

Retention time drift is a known issue in LC/MS instruments. The retention time values of the same peaks are generally different when comparing different samples. This issue is tackled by performing an alignment stage in the LC/MS data processing workflow, to ensure that the same chromatographic regions for different samples would show similar retention time values. When evaluating the retention time drift in a large LC/MS dataset for typical reverse-phase chromatography experiment run (around 7.5 min), we found that the retention time drift, consistently gave different values depending on the chromatographic region. In particular, chromatographic regions of lower retention time showed greater drift. As a consequence, retention time drift showed a correlation pattern along the rt. It was proposed a novel method to correct the retention time drift taking into account its correlation pattern. The performance of this novel method was compared against 3 other peak alignment methods: piecewise-linear, PTW and LOESS. Peak Correlation across samples and peak kurtosis values for 5 chromatographic regions in which there were located known metabolites, were selected as quality metrics. The methods were run in two different datasets, one having a large retention time drift, and the other small retention time drift. The results show that the proposed method has the higher correlation and kurtosis values (0.901 and 0.083 respectively) for the most unaligned regions of the dataset having large retention time drift, whereas it shows similar performance to LOESS method in the other regions (Table 7.1). On the other hand, the performance of the proposed method is similar to that of the LOESS method for the dataset having small retention time drift, but it consistently gave better retention time drift removal for small retention time drifts (Table 7.2).

Overall, the contributions performed throughout this thesis aim at improving the the data analysis workflow of LC/MS Metabolomic data. All these contributions were coded as packages under the R environment and are completely compatible but they can also be used independently as well. Therefore, after analysing the samples through an LC/MS and recording the data files, the peak alignment step might be run using our model-based method using the H-Cor package to specifically remove the retention time drift. Once the data is aligned, the package intCor can use the output files of the HCor package to remove possible intensity drifts and to normalise in the data using our two-step method through the intCor package. Once the drift sources are removed, the files can be analysed end-to-end using the MAIT package. The optional package pagR, codifies the peak aggregation techniques used to improve the predictive power of the LC/MS data. The pagR package is compatible with the MAIT package, allowing to run an end-to-end analysis using spectra instead of peaks. As the MAIT package contains the HMDB, it is possible to give a metabolite table identification after using the pagR package.

9.2 Discussion of the results and further work

From the main contributions of this thesis, it can be derived that the data analysis pipeline and the available computational tools in LC/MS Metabolomics is a prolific field. Moreover, there are some spots in the workflow that need of newer and more efficient algorithms to tackle LC/MS data analysis. First, it is important to state that, as in many other omic sciences, the quality of experimental procedures used in Metabolomics is crucial. Even small variations of the intensity of a single peak due to modifications of the LC/MS conditions or of the experimental conditions, might affect not only one peak but many other peaks due to the technical correlation structure found in the LC/MS metabolomic data.

An important spot to cover in the data analysis workflow, is the peak intensity normalisation. Although they might be an important source of variance in the data, until very recently (2011), it was not published the first scientific article studying intensity drift effects in LC/MS data. This fact means that data normalisation in LC/MS has not been deeply analysed and taken into account in many of the published papers in Metabolomics. This could be in part due to a lack of open large datasets or to a lack of datasets having an important time span between the injected samples in Metabolomics. In this sense, the contribution of the intCor package with the proposed two-step method aims at performing a quick and robust drift correction of the data regardless of the sample size.

Peak annotation algorithms in LC/MS links the LC/MS data with the biological knowledge, and it is another critical step in LC/MS data analysis. There are currently too few tools and algorithms to deal with peak annotation, despite being a processing step of paramount importance. Although in recent years there has been published an important number of methods and tools with different mathematical approaches, this is still a hot scientific topic as there are not very well defined standards.

Another important topic is the data enrichment in Metabolomics. Most of the data enrichment algorithms performed in LC/MS metabolomics such as over representation analysis or gene-set enrichment analysis are imported from the gene expression field where have been used extensively. However, data enrichment algorithms are extremely dependent on the quality of the annotated data. Whereas the gene expression field has a long tradition of open annotation ontologies such as the Gene Ontology Consoritum [1] linking genes and biological processes, in Metabolomics this is rather new and mainly undone. As the metabolomic annotation is improved, more enrichment algorithms will be used in metabolomic analysis as better biological insight will be retrieved from the data. Therefore, this is going to be a prolific scientific topic in the upcoming years.

The highly specific tasks of many of the available tools and the lack of an open and programmable tool to perform end-to-end statistical analysis in Metabolomics inspired the creation of the MAIT tool. As the computational development in Metabolomics is large in number, the MAIT package was coded as highly modularised tool to allow the future replacements with updated and improved algorithms. A nice addition to the MAIT package would be a module to perform data enrichment in the metabolomic context. Other improvements like adding new classification capabilities (e.g. Random Forest classifiers) are also in the agenda of the new versions of MAIT.

9.3 Conclusions

This section summarises the conclusions of the doctoral thesis.

- Traditional LC/MS data analysis pipelines normally are based on one-peak/one-variable data treatment approaches. However, the peak intensity measures are not independent, as a single metabolite analysed through an LC/MS, in general gives a set of peaks that show a clear correlation pattern in their intensity profiles. In this thesis it was proposed a new step in the metabolomic LC/MS data analysis workflow based on computing aggregated measures using the peaks coming from the same metabolite and redefining these measures as the new variables in the dataset. Therefore, applying this step implies a change in the paradigm of one-peak/one-variable to a one-metabolite/one variable concept. This approach was shown to improve the statistical predictive power of the LC/MS metabolomic data.

[1] http://geneontology.org/

This evidence was found true regardless of the peak aggregation method and of the nature of the classifier (linear PLSDA or non-linear SVM).

- The improvement in the statistical predictive power was computed for several different peak aggregation techniques: Spectral Mean, Maximum Peak, PCA Decomposition or NMF Reduction. All these techniques are based on a dataset decomposition according to $Y = S \times L^T + E$ where Y is the LC/MS dataset, S the scores matrix and L the loadings matrix. Each peak aggregation method would lead to a different loadings and scores matrix. When comparing the statistical predictive power of the LC/MS datasets built using these four different peak aggregation measures, it was found that PCA Decomposition and NMF Reduction showed higher classification ratio than the Spectral Mean and Maximum Peak methods. However, due to the nature of the original LC/MS signals (intensity values), the data built through the NMF Reduction method has better interpretability as it only contains positive values (in both loadings and scores).

- A set of tools was coded as an R package named MAIT which allows to perform end-to-end LC/MS data analysis pipeline. This package has been designed as a modularised and programmable software tool with special emphasis on peak annotation and traceability. The core of the package relies in an S4 class named MAIT class. The objects of this class are designed to contain all the information of a single data analysis. The main functions of the MAIT package have a MAIT object as an input and give another MAIT object as an output updated with the new results. To improve the usability of the package, MAIT also includes a set of functions devoted to plotting results and to writing tables. These functions automatically create the files in an ordered manner by creating subfolders. Finally, MAIT not only supports using raw files as an input, but also is possible to use external sources of data (e.g. peak tables detected using another software). The package and its tutorial can be found here: http://b2slab.upc.edu/software-and-downloads/metabolite-automatic-identification-toolkit/

- The intensity of the LC/MS signal can suffer from severe drift effects. In the thesis it was depicted how these effects might become an important source of variance, leading to correlation patterns not related by the underlying biological variability. In most LC/MS metabolomic datasets, some samples (called Quality Control samples) are injected regularly to check the performance of the LC/MS device and test whether its technical behaviour is correct. Therefore, these samples do not contain any biological variability. Normally the feature pattern of the samples in these classes is known so they can be used to calibrate the experimental device. In the context of the thesis, it was proposed a two-step method to correct the peak intensity drifts in LC/MS data. In this method, the drift effects are considered to be contained in the common variance space of the quality control samples. The common variance space is computed using a Common Principal Components approach (CPCA). The first step of the method is to remove this variance from the data followed by a data normalisation using a median fold correction. The proposed was compared to other four different drift correction methods: Component Correction method, the CPCA method, the median fold correction method and the ComBat method. Taking the Silhouette and the Dunn clustering indices as quality metrics, it was found that the dataset corrected using the proposed method, showed less intensity drift effects.

- All the tested methods to correct the intensity drift effects (including the proposed method) were coded in a single R package called intCor. The package, the vignette and the required files to run the vignette are available here: http://b2slab.upc.edu/software-and-downloads/intensity-drift-correction/.

- The profile of the retention time drifts in LC/MS metabolomic signals shows a non-linear behaviour. In particular, in reverse-phase chromatography, some chromatographic regions show more retention time drift than other regions. In this work, we supposed that even though these time drift measure are different, they show a correlation pattern between them. Based on this hypothesis, a new method was developed to remove the time drift in LC/MS Metabolomic data. This method is based on fitting a non-linear model to capture the time drift effects on the data. To that end, the quality control samples were selected to elute at early, middle and later retention times, to ensure that a metabolite signal will be clearly found in all the range of the dataset. The model is fitted using the peak drifts of the metabolites found in the quality control samples. Finally, the drift is removed sample-wise from the dataset. This method was tested on two datasets of different characteristics: one shows clear non-linear time drift effects and the other showed little time drift effects. The quality metrics were the peak kurtosis and peak correlation of the peaks found in the quality control samples. These measures were compared to other three methods: Linear piece-wise, Parametric Time Warping and LOESS method. The results showed that the proposed method performed better than the other methods for the dataset containing large drift effects and it had similar performance than the LOESS method for the dataset with little drift (both the proposed method and the LOESS method over performed the others for this dataset. The proposed method is coded as an R package called HCor and it can be found at: `http://b2slab.upc.edu/software-and-downloads/retention-time-drift-correction/`.

Appendix A

Supporting Information: Peak aggregation techniques

A.1 Peak Correlations

In order to build the correlation matrix shown in Figure 1, a subset of 50 peaks of the peak data set Y was selected (data was processed as stated in section A.2). The samples used to get such data set are the same that were used in the evaluation of the effect of the peak aggregation methods (section "Effect of the Peak Aggregation") and whose obtaining is detailed in section "Experimental Data". Once the data set Y was gathered, peak pairwise Pearson's correlations were calculated and the corresponding correlation matrix was obtained as a result.

A.2 Metabolomic Data Processing

The XCMS package used in this paper to perform the peak detection step [1, 2] allows the user to extract the peaks from a set of samples using certain instrument-dependant input parameters. In the XCMS run was used the matched filter method along with the parameters $snthresh = 2$, $sigma = 2.123$, $mzdiff = 0.3$, $bw = 3$. Peaks were then detected and grouped across samples. Possible retention time deviations were also corrected following standard XCMS peak alignment procedure. After the peak detection step, the R package CAMERA was used to find the peaks features coming from the same metabolite, for building the spectra and for finding the spectral matrices [3]. The CAMERA package uses a retention time window and peak correlation across samples (and within each peak cluster), to build the spectral matrices. The parameters used in the CAMERA package were $perfwhm = 0.6$ and $cor\_eic\_th = 0.7$. Applying this procedure, 5688 peaks were detected which were grouped into 2821 spectra. 2113 of these spectra had one peak, 659 had between 1 and 10 peaks and 50 spectra had more than 10 peaks per spectra.

Once the spectral matrices were found, one of the five methods (the standard method which uses peaks as variables or one of the four peak aggregation methods described in section ”Peak Aggregation Techniques”) was applied on the peak data set Y. The R package NMF was used to perform the computation of the matrices factorisation in the NMF reduction method[4]. After this step, the spectral data set S was built in accordance with the method used.

The data were scaled in all methods to have unit variance. The data were centred when the PCA peak aggregation method was used. Once the spectral data set was built, an ANOVA test was applied on every variable (spectrum) with threshold $p - value <= 1 \cdot 10^{-3}$. 1301 spectra out of the 2821 were found to be statistically significant. Among these 1301 significant spectra, 362 had more than one peak.

In the validation step, it was used the R package e1071 to perform the SVM classifications [5]. In order to choose the SVM parameters it was used the tune.svm function with a radial kernel. To quantify the differences between the peak aggregation methods it was applied a Fisher Least Significant Difference (LSD) test an adjusted p-value threshold of 0.05 using the R package AGRICOLAE [6].

Bibliography

[1] C A Smith, E J Want, G O'Maille, R Abagyan, and G Siuzdak. XCMS: processing mass spectrometry data for metabolite profiling using nonlinear peak alignment,

matching, and identification. *Analytical Chemistry*, 78(3):779–787, 2006. ISSN 00032700. doi: 10.1021/ac051437y. URL http://pubs3.acs.org/acs/journals/doilookup?in_doi=10.1021/ac051437y.

[2] Ralf Tautenhahn, Christoph Böttcher, and Steffen Neumann. Highly sensitive feature detection for high resolution LC/MS. *BMC Bioinformatics*, 9(1):504, 2008.

[3] C. Kuhl, R. Tautenhahn, C. Boettcher, T. R. Larson, and S. Neumann. Camera: an integrated strategy for compound spectra extraction and annotation of liquid chromatography/mass spectrometry data sets. *Analytical Chemistry*, 84:283–289, 2012. URL http://pubs.acs.org/doi/abs/10.1021/ac202450g.

[4] Renaud Gaujoux and Cathal Seoighe. A flexible R package for nonnegative matrix factorization. *BMC bioinformatics*, 11:367, January 2010. ISSN 1471-2105. doi: 10.1186/1471-2105-11-367.

[5] Evgenia Dimitriadou, Kurt Hornik, Friedrich Leisch, David Meyer, and Andreas Weingessel. *e1071: Misc Functions of the Department of Statistics (e1071), TU Wien*, 2011. URL http://CRAN.R-project.org/package=e1071. R package version 1.6.

[6] Felipe de Mendiburu. *agricolae: Statistical Procedures for Agricultural Research*, 2012. URL http://CRAN.R-project.org/package=agricolae. R package version 1.1-1.

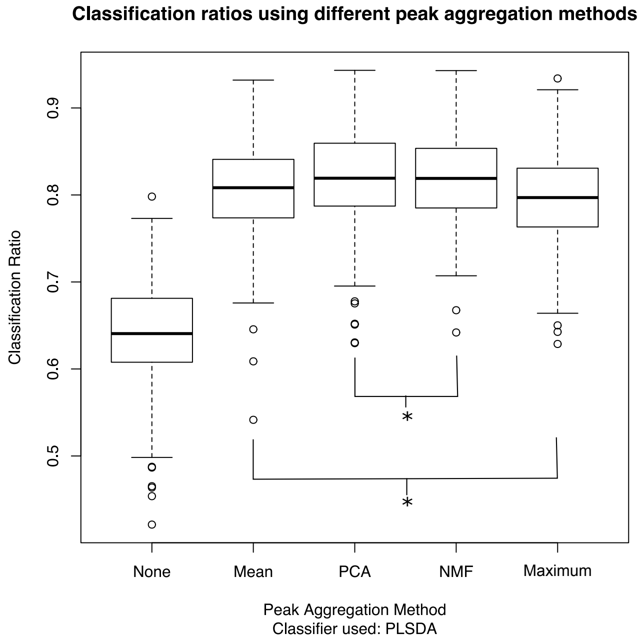


Figure A.1: Boxplot showing the differences in the five methods used when a linear classifier (PLSDA) classifier is used. The method labelled "None" corresponds to the standard method in which no peak aggregation is performed. The lines joining boxes mean that Fisher's LSD test found them to be equal.

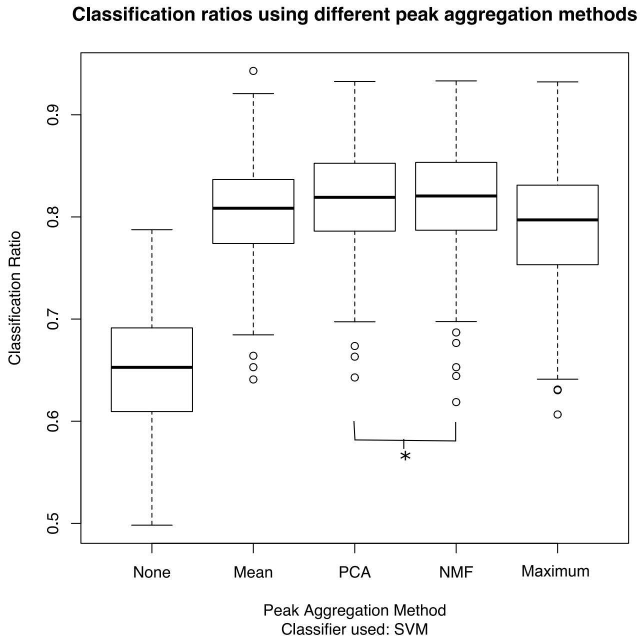


Figure A.2: Boxplot showing the differences in the five methods used when a non-linear classifier (SVM) classifier is used. The method labelled "None" corresponds to the standard method in which no peak aggregation is performed. The line joining boxes mean that Fisher's LSD test found them to be equal.

Appendix B

Supporting Information: MAIT vignette

B.1 Abstract

Processing metabolomic liquid chromatography and mass spectrometry (LC/MS) data files is time consuming. Currently available R tools allow for only a limited number of processing steps and online tools are hard to use in a programmable fashion. This paper introduces the metabolite automatic identification toolkit MAIT package, which allows users to perform end-to-end LC/MS metabolomic data analysis. The package is especially focused on improving the peak annotation stage and provides tools to validate the statistical results of the analysis. This validation stage consists of a repeated random sub-sampling cross-validation procedure evaluated through the classification ratio of the sample files. MAIT also includes functions that create a set of tables and plots, such as principal component analysis (PCA) score plots, cluster heat maps or boxplots. To identify which metabolites are related to statistically significant features, MAIT includes a metabolite database for a metabolite identification stage.

B.2 Introduction

Liquid Chromatography and Mass Spectrometry (LC/MS) is an analytical technique widely used in metabolomics to detect molecules in biological samples [1]. It breaks the

molecules down into pieces, some of which are detected as peaks in the mass spectrometer. Metabolic profiling of LC/MS samples basically consists of a peak detection and signal normalisation step, followed by multivariate statistical analysis such as Principal Components Analysis (PCA) and a wide range of statistical tests such as ANOVA, Welch's test or Kruskal-Wallis test [1, 2].
As analysing metabolomic data is time consuming, a wide array of software tools are available, including commercial tools such as Analyst® software. There are programmatic R packages, such as XCMS [3, 4, 5] to detect peaks or CAMERA package [6] and AStream [7], which cover only peak annotation. Other modularly-designed proposals coded in JAVA such as MZmine or mzMatch are also available [8, 9, 10]. These tools are mainly focused on LC/MS data pre-processing and visualisation. Another category of free available tools consists of those having online access through a graphical user interface (GUI), such as XCMS Online (`http://xcmsonline.scripps.edu`) or MetaboAnalyst [11], both extensively used. These online tools are difficult to use in a programmable fashion. They are also designed and programmed to be used step by step with user intervention, making it difficult to set up metabolomic data analysis workflow.

We introduce a new R package called metabolite automatic identification toolkit (MAIT) for automatic LC/MS analysis. The goal of the MAIT package is to provide an array of tools for programmable metabolomic end-to-end analysis. It consequently has special functions to improve peak annotation through the processes called biotransformations. Specifically, MAIT is designed to look for statistically significant metabolites that separate the classes in the data. MAIT has the following dependencies in terms of R packages: pls, plots, e1071, caret, plsgenomics and agricolae.

B.3 Available Computational Tools

Table B.1 contains a capability comparison between MAIT and some of the most widely used computational tools when processing LC/MS metabolomic data. Among the programable tools, there are R packages such as XCMS which is focused on preprocessing the data but it also is able to perform a simple statistical analysis of the data. MZmine or mzMatch are highly modularised tools based on JAVA and also centred on the data preprocessing and visualisation methods than in the statistical processing of the LC/MS data. On the other hand, the main approach of the online tool MetaboAnalyst is on the statistical analysis of the LC/MS data but it lacks of the programmable approach and off-line capabilities of the previous tools. In this context, MAIT aims at filling the gap of programmatic tools that allow for a full statistical study of LC/MS metabolomic data.

Table B.1: Comparison of some of the main available computational tools for analysing LC/MS data.

Tool	Environment	Programmatic	Statistical Tests	Predictive power methods	Other Statistical Methods and functionalities
XCMS [3]	R package	Yes	Parametric tests: ANOVA, Welch's t test	—	Metabolite Identification
mzMine [8, 9]	Java	Yes	—	—	Clustering Methods Heatmap plots Biotransformations/Neutral losses Annotation Projection Plots (PCA, PLS) Peak Annotation Metabolite Identification
mzMatch [10]	Java	Yes	—	—	Peak Annotation Supports PeakML format
metaboAnalyst [11, 12]	R-based Online Tool	No	Parametric tests: ANOVA, T-test, SAM, EBAM	PLSDA SVM SOM Random Forest	Clustering Methods Heatmap plots Projection Plots (PCA, PLS) Pathway Analysis Metabolite Identification Support for Analysing External Peak Data
MAIT	R package	Yes	Parametric tests: ANOVA, Welch's T test, Student's T test Non-parametric tests: Kruskal-Wallis test Mann-Whitney test Supports user-defined tests	PLSDA SVM KNN	Heatmap plots Projection Plots (PCA, PLS) Peak Annotation Biotransformations/Neutral losses Annotation Metabolite Identification Support for Peak Aggregation Methods Support for Analysing External Peak Data

B.4 MAIT Modularity

Modularity is a highly desirable property of any software tool. A modular package is made of functions each one of which performs highly specific tasks. These functions are used by the package as building blocks to perform more complex procedures. Under the modular design of MAIT, the functions of the package operate with objects of a characteristic class named MAIT-class objects. The main functions of the MAIT workflow fill certain slots of the object and then return the updated MAIT-class object as an output. In this context, Table B.2 shows the slots of the MAIT-class objects that are necessary to run each of the main MAIT functions and also the slots that are filled after the function run. From the same table, it is shown that just a few slots are necessary to run the functions. In particular, considering all the main functions, the required slots are four: @PhenoData@classes, @PhenoData@classNum, FeatureInfo@featureSigID and one of the following slots @RawData@data or @FeatureData@extendedTable. If a certain module is to be added, it is only necessary to fill the slots with the appropriate data. Function getScoresTable returns a peak scores table generated from a xsAnnotate object (see documentation of CAMERA package) if available (it comes from the peakAnnotation function) to extract the peak data. If there is no data in the slot, the table saved in @FeatureData@extendedTable is taken instead. New modules for peak detection and peak annotation stages (see Figure B.1) to be implemented in the MAIT workflow, should create an overload of the function getScoresTable to extract the appropriate data (see help file of the function getScoresTable). Another option would be to use the MAITbuilder function instead (Sections B.6.5 and B.7.9).

Table B.2: Slots of the MAIT-class object filled for each step. Optional slots are labelled with an asterisk.

Step **(Function)**	**Input Slots**	**MAIT-class object Slots filled**
Peak Detection (sampleProcessing())	Raw Sample files	**@RawData@data**: Object containing the raw peak data as an xcms-Set-object. **@PhenoData@classes**: Vector containing the class names. **@PhenoData@classNum**: Vector containing the number of samples for each class. **@PhenoData@resultsPath**: Character containing the path where the results are going to be saved.
Peak Annotation (peakAnnotation())	**@RawData@data** **@PhenoData@classes*** **@PhenoData@classNum***	**@RawData@data** ¡- Object containing the peak annotated data .
Statistical Analysis (spectralSigFeatures())	**@PhenoData@classes** **@PhenoData@classNum** .... **@FeatureData@extendedTable** or **@RawData@data**	**@FeatureData@pvalues** ¡- Vector containing the p-values of the tests **@FeatureData@LSDResults** ¡- Vector containing the results of the Fisher tests **@FeatureData@pvaluesCorrection** ¡- Name of the post hoc method used (if any) **@FeatureData@scores** ¡- Full dataset having rows as variables and columns as samples. **@FeatureData@featureSigID** ¡- Numeric vector with the found significant variables for the dataset saved in the slot @FeatureData@scores
Biotransformations (Biotransformations())	**@FeatureData@featureSigID** .... **@FeatureData@extendedTable** or **@RawData@data**	**@FeatureInfo@biotransformations**: Matrix containing the biotransformations found.
Metabolite Identification (identifyMetabolites())	**@FeatureInfo@biotransformations*** **@FeatureData@extendedTable** or **@RawData@data**	**@FeatureInfo@metaboliteTable**: Dataframe containing the results of the metabolite identification and the statistical analysis.
Predictive Model (Validation())	**@PhenoData@classes** **@PhenoData@classNum** .... **@FeatureData@extendedTable** or **@RawData@data**	**@Validation@classifRatioClasses**: The Classification Ratio per class, classifier and iteration. **@Validation@ovClassifRatioTable**: Table with the mean and standard error of the Classification Ratio per Classifier **@Validation@ovClassifRatioData**: The Classification Ratio per classifier
PLS Model/Plots (plotPLS())	**@PhenoData@classes** **@PhenoData@classNum** .... **@FeatureData@extendedTable** or **@RawData@data**	**@FeatureData@plsModel**: The PLS model of the statistically significant data is sassed in this slot.
PCA Model/Plots (plotPLS())	**@PhenoData@classes** **@PhenoData@classNum** .... **@FeatureData@extendedTable** or **@RawData@data**	**@FeatureData@pcaModel**: The PCA model of the statistically significant data is sassed in this slot.

B.5 Methodology

The main processing steps for metabolomic LC/MS data include the following stages: peak detection, peak annotation and statistical analysis. In the peak detection stage, the objective is to detect the peaks in the LC/MS sample files. The peak annotation stage identifies the metabolites in the metabolomic samples better by increasing the chemical and biological information in the data set. A statistical analysis step is essential to obtain significant sample features. All these 3 steps are covered in the MAIT workflow (see Figure B.1).

B.5.1 Peak Detection

Peak detection in metabolomic LC/MS data sets is a complex issue for which several approaches have been developed. Two of the most well established techniques are matched filter [13] and the centWave algorithm [4]. MAIT can use both algorithms through the XCMS package.

B.5.2 Peak Annotation

The MAIT package uses 3 complementary steps in the peak annotation stage.

- The first annotation step uses a peak correlation distance approach and a retention time window to ascertain which peaks come from the same source metabolite, following the procedure defined in the CAMERA package [6]. The peaks within each peak group are annotated following a reference adduct/fragment table and a mass tolerance window.

- The second step uses a mass tolerance window inside the peak groups detected in the first step to look for more specific mass losses called biotransformations. To do this, MAIT uses a predefined biotransformation table where the biotransformations we want to find are saved. A user-defined biotransformation table can be set as an input following the procedure defined in Section (B.7.6).

- The third annotation step is the metabolite identification stage, in which a predefined metabolite database is mined to search for the significant masses, also using a tolerance window. This database is the Human Metabolome Database (HMDB) [14], 2009/07 version.

B.5.3 Statistical Analysis

The objective of analysing metabolomic profiling data is to obtain the statistically significant features that contain the highest amount of class-related information. To gather these features, MAIT applies standard parametrical and non-parametrical statistical tests on every feature and selects the significant set of features by setting up a user-defined threshold P-value. Depending on the number of classes defined in the data, MAIT can use Student's T-test, Welch's T-tests and Mann-Whitney tests for two classes or ANOVA and Kruskal-Wallis tests for more than two classes. Furthermore, MAIT supports adding user-defined tests in a straightforward way (see section B.7.4 for an example using the Fisher's exact test). Different multiple testing correction methods including false discovery rate and Bonferroni are implemented in MAIT through R function p.adj.

We propose a validation test to quantify how well the data classes are separated by the statistically significant features. The separation is validated through a repeated random sub-sampling cross-validation using partial least squares and discriminant analysis (PLS-DA), support vector machine (SVM) with a radial Kernel and K-nearest neighbours (KNN) [15]. Overall and class-related classification ratios are obtained to evaluate the class-related information of the significant features.

B.5.4 Support for Peak Aggregation Techniques

MAIT optionally supports peak aggregation techniques that might lead to better feature selection [16] through the commercial pagR package.

B.6 MAIT workflow

MAIT accepts LC/MS files in the open formats mzData and netCDF. Sample files should be placed in a folder having a set of subfolders, each of which is going to be a class in the data (see function sampleProcessing() in Section B.7 for details).

The package centrepiece consists of the S4 MAIT-class objects. In terms of traceability, objects belonging to this class are designed to contain all the information related to the processing steps already run. The reason for this design is that using a single R object throughout the workflow improves the traceability of the analysis. The contents of a MAIT-class object are shown below. The slots of the MAIT-class objects are:

```
Formal class 'MAIT' [package "MAIT"] with 5 slots
  ..@ FeatureInfo:Formal class 'MAIT.FeatureInfo' [package "MAIT"] with
   3 slots
  .. .. ..@ biotransformations: logi [1, 1] NA
  .. .. ..@ peakAgMethod      : chr ""
  .. .. ..@ metaboliteTable   :'data.frame': 0 obs. of  0 variables
  ..@ RawData    :Formal class 'MAIT.RawData' [package "MAIT"] with 2 slots
  .. .. ..@ parameters:Formal class 'MAIT.Parameters' [package "MAIT"] with
  10 slots
  .. .. .. .. ..@ sampleProcessing   : list()
  .. .. .. .. ..@ peakAnnotation     : list()
  .. .. .. .. ..@ peakAggregation    : list()
  .. .. .. .. ..@ sigFeatures        : list()
  .. .. .. .. ..@ biotransformations : list()
  .. .. .. .. ..@ identifyMetabolites: list()
  .. .. .. .. ..@ classification     : list()
  .. .. .. .. ..@ plotPCA            : list()
  .. .. .. .. ..@ plotPLS            : list()
  .. .. .. .. ..@ plotHeatmap        : list()
  .. .. ..@ data      : list()
  ..@ Validation :Formal class 'MAIT.Validation' [package "MAIT"] with 3
   slots
  .. .. ..@ ovClassifRatioTable: logi [1, 1] NA
  .. .. ..@ ovClassifRatioData : list()
  .. .. ..@ classifRatioClasses: logi [1, 1] NA
  ..@ PhenoData  :Formal class 'MAIT.PhenoData' [package "MAIT"] with 3 slots
  .. .. ..@ classes    : logi(0)
  .. .. ..@ classNum   : logi(0)
.. .. ..@ resultsPath: chr ""
..@ FeatureData:Formal class 'MAIT.FeatureData' [package "MAIT"] with 12
slots
.. .. ..@ scores            : logi [1, 1] NA
.. .. ..@ featureID         : logi(0)
.. .. ..@ featureSigID      : logi(0)
.. .. ..@ LSDResults        : logi [1, 1] NA
.. .. ..@ models            : list()
.. .. ..@ pvalues           : logi(0)
.. .. ..@ pvaluesCorrection: chr ""
.. .. ..@ pcaModel          : list()
.. .. ..@ plsModel          : list()
.. .. ..@ masses            : num(0)
.. .. ..@ rt                : num(0)
.. .. ..@ extendedTable     :'data.frame': 0 obs. of  0 variables
```

A MAIT-class object is built of 5 different S4 classes:

- FeatureInfo-class: The information regarding the peak annotation is saved in this class.
- RawData-class: This class contains the data imported from the metabolomic LC/MS (xcmsSet-class object or xsAnnotate-class object depending on the last function run)
- Validation-class: This contains the results of the cross-validation classification stage.
- PhenoData-class: All the class-related information and the results path is contained in this class.
- FeatureData-class: This class contains the information related to the features, its P-values and the mathematical models used.

Figure B.1 shows the flowchart of the main functions of the MAIT package, their output files and their functionality. Table B.3 shows the specific outputs of each function shown in Figure B.1.

The MAIT package uses the wrapper function sampleProcessing() to call the required XCMS functions to perform the peak detection step. These functions include xcmsSet(), group(), retcor() and fillPeaks(). The peaks detected are saved as a xcmsSet-class object inside a MAIT-class object.

B.6.1 Peak Annotation

The default tables used to perform all the peak annotation steps are provided in MAIT as an R Data object called MAITtables.RData. When this file is loaded, the following objects can be found:

- posAdducts: The possible annotations for the first annotation step when the polarisation mode in the sample acquisition is set to positive.
- negAdducts: The possible annotations for the first annotation step when the polarisation mode in the sample acquisition is set to negative.
- biotransformationsTable: This table contains the specific biotransformations for the second annotation step.
- Database: The metabolite database table to perform the metabolite identification stage (third peak annotation step). This database is the Human Metabolome Database (HMDB)[14], 2009/07 version.

The MAIT package uses a CAMERA package wrapper function called peakAnnotation() to perform the first step in the peak annotation stage. CAMERA groups the peaks using a retention time window followed by a correlation cut-off approach. An adduct table is required to launch this step. A user-defined adduct table or a MAIT default adduct table (posAdducts or negAdducts) can be selected. The user-defined table should be created following the CAMERA adduct table layout, which is:

```
        name nmol charge  massdiff oidscore quasi ips
1     [M+H]+    1      1  1.007276        1     1 1.0
```

```
2     [M+Na]+    1       1 22.989218         8      1 1.0
3      [M+K]+    1       1 38.963158        10      1 1.0
4    [M+NH4]+    1       1 18.033823        16      1 1.0
5 [M+2Na-H]+    1       1 44.971160        34      0 0.5
6   [M+2K-H]+    1       1 76.919040        60      0 0.5
```

The second peak annotation step is performed by the function called Biotransformations(). The function is codified to perform the procedure defined in Section B.5.2. As is shown in Figure B.1, function Biotransformations() should be launched after detecting the significant features using function spectralSigFeatures() (see Section B.6.2). The first 10 entries of the Biotransformation Table are shown below. When 2 peaks in the same peak group have mass differences (within tolerance) equal to a value of the MASSDIFF column, they are related to each other by that biotransformation and are annotated accordingly.

```
                        NAME MASSDIFF
1              debenzylation -90.0468
2   tert-butyl dealkylation -56.0624
3            decarboxylation -43.9898
4     isopropyl dealkylation -42.0468
5       propylketone to acid -40.0675
6      tert-butyl to alcohol -40.0675
7     alkenes to dihydrodiol  34.0054
8            nitro reduction -29.9742
9       propyl ether to acid -28.0675
10              deethylation -28.0312
```

Likewise, to perform the third peak annotation step, function identifyMetabolites() mines the metabolite database file to find suitable metabolites for each peak. The function outputs a table (see Table B.3) that contains all the possible matches for all the peaks. If no peak aggregation technique was applied through function peakAggregation() (see Section B.6.3), the set of features to be identified are all the significant features found in the statistical tests (Section B.6.2). A user-defined database can be used as an input object as well. To do so, the user file should have the following format:

```
      ENTRY                        NAME    FORMULA       MASS
1 HMDB00001            1-Methylhistidine  C7H11N3O2 169.085129
2 HMDB00002          1,3-Diaminopropane     C3H10N2  74.084396
3 HMDB00005           2-Ketobutyric acid     C4H6O3 102.031693
4 HMDB00008        2-Hydroxybutyric acid     C4H8O3 104.047340
5 HMDB00010            2-Methoxyestrone   C19H24O3 300.172546
6 HMDB00011     (R)-3-Hydroxybutyric acid    C4H8O3 104.047340


                    Biofluid
Blood; Cerebrospinal Fluid; Saliva; Urine
Blood; Urine
Blood; Cerebrospinal Fluid; Urine
Blood; Cerebrospinal Fluid; Urine
Urine
Blood; Cerebrospinal Fluid; Urine
```

Each of these 3 annotation steps is implemented through a function. These 3 functions all have an input parameter where a user-defined table can be used instead of the MAIT default tables. In particular, in function peakAnnotation() there is the argument adductTable, in function Biotransformations(), the argument is called bioTable and the input argument for function identifyMetabolites() is called database.

B.6.2 Statistical Analysis

spectralSigFeatures() performs a univariate statistical test on each feature to gather the statistically significant variables that separate the classes in the data. The results of these statistical tests are saved in the MAIT-class object and are easily retrieved from it by applying function sigPeaksTable(). The validation procedure defined in Section B.5.3 is launched using function Validation(). The overall and class-related classification ratios are saved in boxplots and tables (see Table B.3) in the folder called "Validation". The confusion matrices for each iteration and classifier are saved in the folder named "Confusion_Tables".

B.6.3 Support for Peak Aggregation Techniques

The peak aggregation techniques, optional in MAIT workflow, are applied through function peakAggregation(). This function allows the use of several different methods to obtain the peak aggregation measures. If the chosen method is None, no other packages are required and no peak aggregation technique is applied. Any other valid choice (Mean, Single, PCA, NMF) requires the additional commercial package pagR (see Figure B.1).

B.6.4 Statistical Plots

The package also contains functions that create statistical plots to evaluate analysis results. These plots include 2D PCA score plots and an interactive 3D PCA score plot through function plotPCA(). The interactive 3D PCA score plot is generated by the package rgl [17]. Function plotHeatmap() produces an array of heat maps using different thresholds for the P-values and hierarchical clustering distances (Euclidean and Pearson's; see Table B.3), whereas Function plotBoxplot() makes it possible to create a boxplot for each significant feature found. As is shown in Figure B.1, all 3 functions require the significant features to be found to run the functions correctly and create the plots.

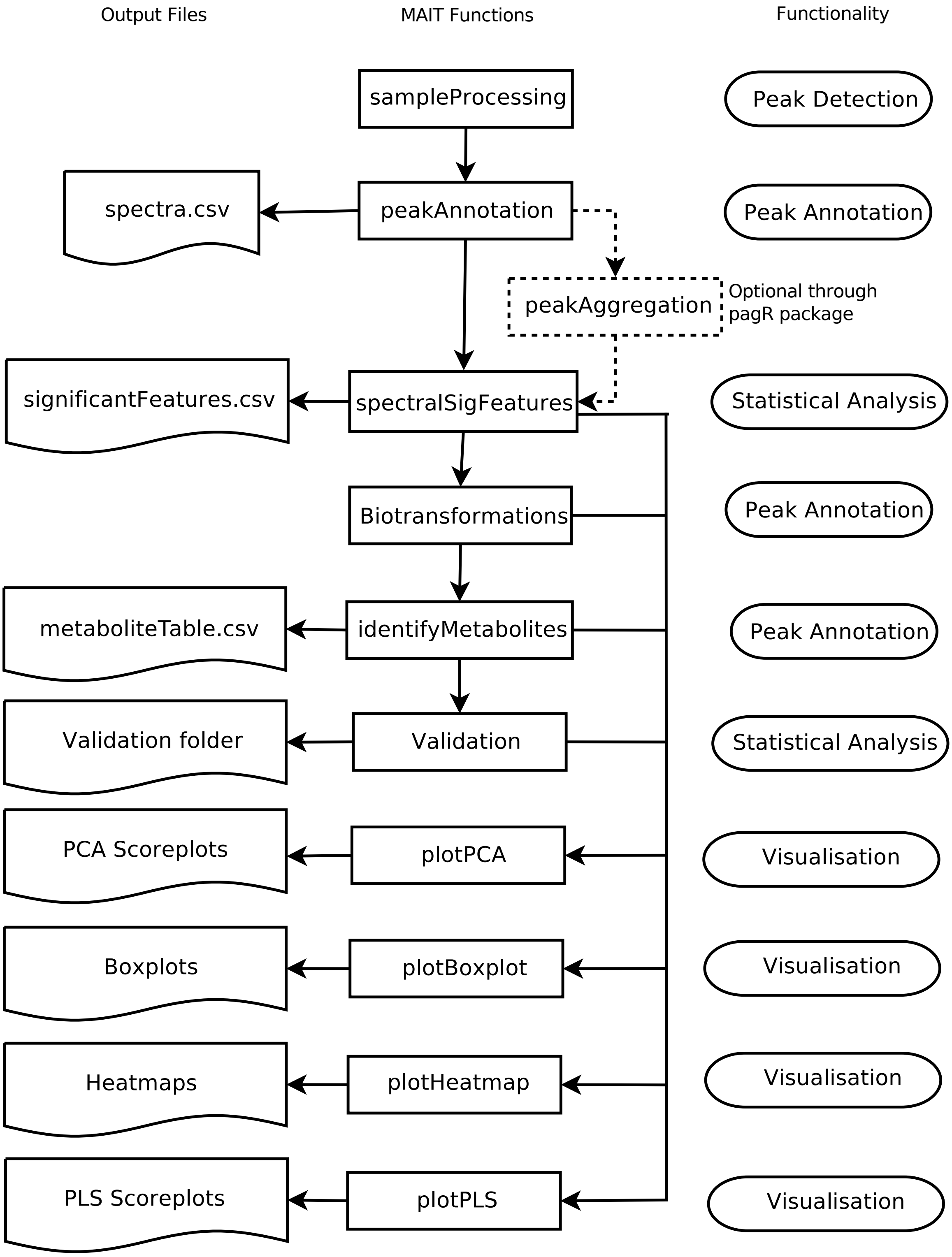


Figure B.1: Flowchart showing the main MAIT functions. Each box refers to a function and each circle points to the functionality of the function in the workflow. Solid arrows refer to possible data processing path. The left column plots contain the output of the functions.

Table B.3: Table showing the output files generated by the main MAIT functions shown in Figure B.1.

MAIT Function	**Output File Name**	**Output type**	**Description**
peakAnnotation	Spectra.csv	Table	This table summarises the correspondence between peaks and spectra
spectralSigFeatures	significantFeatures.csv	Table	In this table the results of the univariate tests performed for every feature and the information of the peak annotation are saved.
identifyMetabolites	metaboliteTable.csv	Table	This table summarises the results of all the previous functions in the workflow (see Figure B.1), including peakAnnotation, spectralSigFeatures and Biotransformations. The possible metabolite identification matches are also included in the table.
plotPCA	Scoreplot_PC12.png	Plot	This file contains the PCA score plot for Principal Component 1 vs Principal Component 2
	Scoreplot_PC13.png	Plot	This file contains the PCA score plot for Principal Component 1 vs Principal Component 3
	Scoreplot_PC23.png	Plot	This file contains the PCA score plot for Principal Component 2 vs Principal Component 3
plotPLS	Scoreplot_PLS_(Ncomps).png	Plots	Depending on the number of components found, this function generates one PLS Scoreplot (1 and 2 components) or three PLS Scoreplots (3 components)
plotHeatmap	X_Distance_Heatmap_pY.png	Plots	This set of files contain the heat maps after applying a hierarchical clustering using X distance (X=Euclidean or Correlation) and Y P-value (Y=0.05, 0.01, 0.001, 1e-4, 1e-5)
plotBoxplot	Boxplot_spectra_X.png	Plots	This set of files contain a boxplot for each significant feature found in the analysis.
Validation	Confusion_Tables	Folder	Folder where the Confusion matrices for every iteration step are saved
	Boxplot_Clases_Classification.png	Plot	Boxplot showing the classification ratio for each class and classifier
	Boxplot_Overall_Classification.png	Plot	Boxplot showing the classification ratio for classifier regardless of the classes.
	ClassificationTable_Class_X.csv	Tables	Table showing the classification ratios for each classifier and for class X. There one of these tables for each class in the data.
	ClassificationTable.csv	Table	Table showing the overall classification ratios for each classifier regardless of the classes.

B.6.5 External Peak Data

MAIT supports importing external peak data through a function called MAITbuilder. This function allows the user to create a MAIT object from a wide variety of data. Table B.4 shows the correspondence between the arguments that the user needs to provide to the MAITbuilder function and the function that the user wants to run. An important point of the MAITbuilder function is the spectraID argument. Whereas this argument is not necessary to run any of the functions related to the statistical processing, it has a big impact on the annotation functions (i.e. Biotransformations and identifyMetabolites). The reason is that, if no spectral information is provided and the flag named spectraEstimation is set to FALSE, the MAITbuilder function considers the provided data as being all in separate spectra (one peak/one spectrum). Therefore the annotation functions will not find any annotation for the provided data. Nevertheless, if the spectraEstimation flag is set to TRUE, MAIT uses a retention time window (defined by the argument rtRange) and a correlation threshold value (defined by corThresh) to estimate a peak grouping into spectra for the provided data.

B.7 Using MAIT

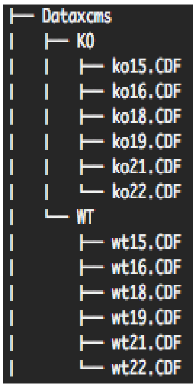
The data files for this example are a subset of the data used in reference [18], which are freely distributed through the faahKO package [19]. In these data there are 2 classes of mice: a group where the fatty acid amide hydrolase gene has been suppressed (class knockout or KO) and a group of wild type mice (class wild type or WT). There are 6 spinal cord samples in each class. In the following, the MAIT package will be used to read and analyse these samples using the main functions discussed in Section B.6. The significant features related to each class will be found using statistical tests and analysed through the different plots that MAIT produces.

B.7.1 Data Import

Each sample class file should be placed in a directory with the class name. All the class folders should be placed under a directory containing only the folders with the files to be analysed. In this case, 2 classes are present in the data. An example of correct file

Table B.4: Correspondence between the necessary arguments of the MAITbuilder and the MAIT functions that can be launched. Given a function, the arguments not mentioned in the should be considered as optional for that function. The argument significantFeatures is a flag that, if it is set to TRUE, the provided features are considered to be statistically significant. A field labelled with an asterisk refers to an optional argument.

MAIT function to be launched	**Necessary arguments of the MAITbuilder function**
spectralSigFeatures()	classes, data
Biotransformations()	masses, significantFeatures=TRUE, spectraID*
identifyMetabolites()	masses,significantFeatures=TRUE, spectraID*
Validation()	classes, data, significantFeatures=TRUE
Plot functions (plotBoxplot,plotHeatmap, plotPCA, plotPLS)	classes, data, significantFeatures=TRUE



```
├─ Dataxcms
|   ├─ KO
|   |   ├─ ko15.CDF
|   |   ├─ ko16.CDF
|   |   ├─ ko18.CDF
|   |   ├─ ko19.CDF
|   |   ├─ ko21.CDF
|   |   └─ ko22.CDF
|   └─ WT
|       ├─ wt15.CDF
|       ├─ wt16.CDF
|       ├─ wt18.CDF
|       ├─ wt19.CDF
|       ├─ wt21.CDF
|       └─ wt22.CDF
```

Figure B.2: Example of the correct sample distribution for MAIT package use. Each sample file has to be saved under a folder with its class name.

distribution using the example data files is shown in Figure B.2.

B.7.2 Peak Detection

Once the data is placed in 2 subdirectories of a single folder, the function sampleProcessing() is run to detect the peaks, group the peaks across samples, perform the retention time correction and carry out the peak filling process. As function sampleProcessing() uses the XCMS package to perform these 4 processing steps, this function exposes XCMS parameters that might be modified to improve the peak detection step. A project name should be defined because all the tables and plots will be saved in a folder using that name. For example, typing project = "project_Test", the output result folder will be "Results_project_Test".

By choosing "MAIT_Demo" as the project name, the peak detection stage can be launched by typing:

```
R> MAIT <- sampleProcessing(dataDir = "Dataxcms", project = "MAIT_Demo",
snThres = 2,rtStep = 0.03)

ko15: 215:366 230:680 245:1014 260:1392 275:1766 290:2120 305:2468 320:2804
335:3150 350:3468 365:3846 380:4182 395:4486 410:4804 425:5110 440:5444
455:5778 470:6136 485:6504 500:6892 515:7296 530:7742 545:8138 560:8620
575:9048 590:9526
ko16: 215:344 230:662 245:1018 260:1378 275:1728 290:2090 305:2434 320:2722
335:3030 350:3352 365:3680 380:4006 395:4310 410:4640 425:4966 440:5276
455:5618 470:6010 485:6370 500:6818 515:7230 530:7662 545:8108 560:8608
575:9110 590:9654

...

wt22: 215:304 230:568 245:872 260:1202 275:1536 290:1838 305:2150 320:2444
335:2758 350:3030 365:3306 380:3576 395:3848 410:4140 425:4420 440:4712
455:5018 470:5364 485:5692 500:6060 515:6472 530:6912 545:7326 560:7786
575:8302 590:8792
Peak detection done
262 325 387 450 512 575
```

```
 Retention Time Correction Groups: 7



Warning:  Span too small, resetting to 0.8

 Retention time correction done
 262 325 387 450 512 575
 Peak grouping after samples done
 ko15
 .
 .
 .
 Peak missing integration done
```

After having launched the sampleProcessing function, peaks are detected, they are grouped across samples and their retention time values are corrected. A short summary in the R session can be retrieved by typing the name of the MAIT-class object.

```
 R> MAIT
A MAIT object built of 12 samples
The object contains 6 samples of class KO
The object contains 6 samples of class WT
```

The result is a MAIT-class object that contains information about the peaks detected, their class names and how many files each class contains. A longer summary of the data is retrieved by performing a summary of a MAIT-class object. In this longer summary version, further information related to the input parameters of the whole analysis is displayed. This functionality is especially useful in terms of traceability of the analysis.

```
 R> summary(MAIT)
A MAIT object built of 12 samples
The object contains 6 samples of class KO
The object contains 6 samples of class WT
```

```
Parameters of the analysis:
                    Value
dataDir             "Data"
snThres             "2"
Sigma               "2.12332257516562"
mzSlices            "0.3"
retcorrMethod       "loess"
groupMethod         "density"
bwGroup             "3"
mzWidGroup          "0.25"
filterMethod        "matchedFilter"
rtStep              "0.03"
nSlaves             "0"
project             "MAIT_Demo"
ppm                 "10"
minfrac             "0.5"
fwhm                "30"
family1             "gaussian"
family2             "symmetric"
span                "0.2"
centWave peakwidth1 "5"
centWave peakwidth2 "20"
```

B.7.3 Peak Annotation

The next step in the data processing is the first peak annotation step, which is performed through the peakAnnotation(). If the input parameter adductTable is not set, then the default MAIT table for positive polarisation will be selected. However, if the adductTable parameter is set to ”negAdducts”, the default MAIT table for negative fragments will be chosen instead. peakAnnotation function also creates an output table (see Table B.3) containing the peak mass (in charge/mass units), the retention time (in minutes) and the spectral ID number for all the peaks detected. A call of the function peakAnnotation may be:

```
 R> MAIT <- peakAnnotation(MAIT.object = MAIT, corrBetSamp = 0.75, perfwhm = 0.6)

 WARNING: No input adduct/fragment table was given. Selecting default MAIT table
 for positive polarity...
Set adductTable equal to negAdducts to use the default MAIT table for negative
polarity
Start grouping after retention time.
Created 1037 pseudospectra.
Spectrum build after retention time done
Generating peak matrix!
Run isotope peak annotation
 % finished: 10  20  30  40  50  60  70  80  90  100
Found isotopes: 15
Isotope annotation done
Start grouping after correlation.
Generating EIC's ..

Calculating peak correlations in 1037 Groups...
 % finished: 10  20  30  40  50  60  70  80  90  100

Calculating peak correlations across samples.
 % finished: 10  20  30  40  50  60  70  80  90  100

Calculating isotope assignments in 1037 Groups...
 % finished: 10  20  30  40  50  60  70  80  90  100
Calculating graph cross linking in 1037 Groups...
 % finished: 10  20  30  40  50  60  70  80  90  100
New number of ps-groups:  2398
xsAnnotate has now 2398 groups, instead of 1037
Spectrum number increased after correlation done
```

```
Generating peak matrix for peak annotation!
Found and use user-defined ruleset!
Calculating possible adducts in 2398 Groups...
 % finished: 10  20  30  40  50  60  70  80  90  100
Adduct/fragment annotation done
```

Because the parameter adductTable was not set in the peakAnnotation call, a warning was shown informing that the default MAIT table for positive polarisation mode was selected. The xsAnnotated object that contains all the information related to peaks, spectra and their annotation is stored in the MAIT object. It can be retrieved by typing:

```
 R> rawData(MAIT)
$xsaFA
An "xsAnnotate" object!
With 2398 groups (pseudospectra)
With 12 samples and 2640 peaks
Polarity mode is set to:  positive
Using automatic sample selection
Annotated isotopes: 15
Annotated adducts & fragments: 16
Memory usage: 7.07 MB
```

B.7.4 Statistical Analysis

Following the first peak annotation stage, we want to know which features are different between classes. Consequently, we run the function spectralSigFeatures().

```
R> MAIT<-spectralSigFeatures(MAIT.object = MAIT, pvalue = 0.05, p.adj = "none",
scale = FALSE)
```

It is worth mentioning that by setting the scale parameter to TRUE, the data will be scaled to have unit variance. The parameter p.adj allows for using the multiple testing correction methods included in the function p.adjust of the package stats. A summary of the statistically significant features is created and saved in a table called significantFeatures.csv (see Table B.3). It is placed inside the Tables subfolder located in the project folder. This table shows characteristics of the statistically significant features, such as their P-value, the peak annotation or the expression of the peaks across samples. This table can be retrieved at any time from the MAIT-class objects by typing the instruction:

```
R> signTable <- sigPeaksTable(MAIT.object = MAIT, printCSVfile = FALSE)
R> head(signTable)
```

```
        mz mzmin mzmax    rt rtmin rtmax npeaks KO WT      ko15       ...
610  300.2 300.1 300.2 56.36 56.18 56.56     17  6  3 4005711.4       ...
762  326.2 326.1 326.2 56.92 56.79 57.00      9  5  2 3184086.4       ...
885  348.2 348.1 348.2 56.95 56.79 57.15     14  4  2  320468.2       ...
1760 495.3 495.2 495.3 56.93 56.82 57.05     11  3  4  110811.4       ...
935  356.2 356.1 356.3 63.77 63.58 63.92      9  4  4  962224.6       ...
1259 412.2 412.1 412.3 68.61 68.44 68.81     16  4  3  113096.3       ...

    isotopes           adduct pcgroup   P.adjust          p
610                                27 0.01748294 0.01748294     ...
762             [M+H]+ 325.202      31 0.01991433 0.01991433     ...
885            [M+Na]+ 325.202      31 0.16856322 0.16856322     ...
1760                               31 0.96828618 0.96828618     ...
935                                74 0.03310409 0.03310409     ...
1259                               81 0.02240898 0.02240898     ...
      Median Class KO Median Class WT
610      2769931.356        115642.29
762      2353947.791         43006.61
885        40384.825             0.00
1760        6531.515         15969.26
935       848999.980         16836.67
1259      215979.768         34607.95
```

The number of significant features can be retrieved from the MAIT-class object as follows:

```
 R> MAIT
```

```
A MAIT object built of 12 samples and 2640 peaks.
No peak aggregation technique has been applied
106 of these peaks are statistically significant
The object contains 6 samples of class KO
The object contains 6 samples of class WT
```

By default, when using two classes, the statistical test applied by MAIT is the Welch's test. Nevertheless, when having two classes,MAIT also supports applying the Student's t-test and the non-parametric test Mann-Whitney test. For using the Student's t-test on the data, the call to the spectralSigFeatures function should be as:

```
R> MAIT_Student <- spectralSigFeatures(MAIT.object = MAIT, pvalue = 0.05,
p.adj = "none", scale = FALSE, var.equal = TRUE)
R> MAIT_Student
A MAIT object built of 12 samples and 2640 peaks.
No peak aggregation technique has been applied
148 of these peaks are statistically significant
The object contains 6 samples of class KO
The object contains 6 samples of class WT
```

If we want to apply the Mann-Whitney test, in this case is necessary to add some jitter noise in our data. The reason is that the Mann-Whitney test has ties when the data has values equal to zero. Adding a small noise to the data solves this issue. MAIT supports using jitter noise through the flag jitter and the parameter jitter.amount:

```
R> MAIT_MW <- spectralSigFeatures(MAIT.object = MAIT, pvalue = 0.05,
p.adj = "none", scale = FALSE, parametric = FALSE, jitter = TRUE)
```

As an example of its modularity, MAIT supports applying user-defined statistical tests on the data. To use a user-defined test in MAIT, the output of the function should give the p-value of the test given a numeric vector (i.e. the variable values) and a factor vector (the classes of the samples in the parameter group). In the following example, lets suppose that we want to know if a certain metabolite is present or not in the data. To that end, we will define a special exact Fisher's test function with a threshold intensity value. If the intensity of the peak for a particular sample is above this threshold value, the peak is labelled as "present". If the intensity value is below the threshold, the peak is labelled as "absent". If all the labels for a particular peak are the same, the function will not compute the Fisher's test and will throw an NA as a p-value. The function of the Fisher's test, can be defined as follows:

```
R> ftest <- function(x,group){
threshold<-100
x[x>threshold]<-"present"
x[!x>threshold]<-"absent"
x<-as.factor(x)
if(length(summary(x))==1){
out<-NA
}else{
out<-fisher.test(x=x,y=group)$p.value
return(out)
}}
```

And the call to the spectralSigFeatures in this case:

```
R> MAIT_Fisher <- spectralSigFeatures(MAIT.object = MAIT, test.fun = ftest,
namefun = "fisher's test")
R> MAIT_Fisher
A MAIT object built of 12 samples and 2640 peaks.
No peak aggregation technique has been applied
18 of these peaks are statistically significant
The object contains 6 samples of class KO
The object contains 6 samples of class WT
```

```
R> sigPeaksTable(MAIT_Fisher)

         mz mzmin mzmax    rt rtmin rtmax npeaks KO WT     ko15       ko16      ko18
686  314.20 314.1 314.3 58.34 58.26 58.52      9  4  2 53657.59  44311.386  46921.83
743  323.10 323.1 323.2 58.36 58.10 58.70     32  5  6 93579.73 163605.100 269543.46
1879 512.10 512.1 512.1 58.36 58.30 58.36      3  3  0 20781.25   9272.833  15164.85
2398 572.10 572.0 572.2 58.35 58.23 58.54     10  4  4 17548.34      0.000  10066.08
2425 574.15 574.1 574.3 58.36 58.17 58.54      8  3  3     0.00      0.000   7615.29
2484 582.10 582.0 582.2 58.36 57.88 58.75     32  5  6 67578.36  19928.601  20647.05

           ko19      ko21       ko22      wt15       wt16       wt18       wt19
686   21571.953 11447.427  12034.850  10547.28   6815.575  13099.050   8719.591
743  146186.650  4367.423 128649.260 231889.92 223209.690 105094.445 270387.409
1879   9574.670     0.000   2273.451      0.00      0.000      0.000      0.000
2398   4721.605 17117.194      0.000   5033.04   6322.965   9087.955      0.000
2425      0.000  4596.405   5992.385      0.00   8590.285   5377.340      0.000
2484  16169.580  5601.135      0.000  20898.59  63014.725   9707.695   8850.075

           wt19      wt21     wt22 isotopes adduct pcgroup   P.adjust          p
686    8719.591  2475.368     0.00                      98 1.00000000 1.00000000
743  270387.409 26666.035 55103.65                      98         NA         NA
1879      0.000     0.000     0.00                      98 0.01515152 0.01515152
2398      0.000  8717.050 11374.42                      98 1.00000000 1.00000000
2425      0.000     0.000  6685.68                      98 1.00000000 1.00000000
2484   8850.075  5623.045 25839.54                      98 1.00000000 1.00000000

              p Fisher.Test Mean Class KO Mean Class WT Median Class KO Median Class
686  1.00000000          NA     31657.506      6942.811       32941.669        7767.5
743          NA          NA    134321.938    152058.525      137417.955      164152.0
1879 0.01515152          NA      9511.176         0.000        9423.752           0.0
2398 1.00000000          NA      8242.204      6755.905        7393.843        7520.0
2425 1.00000000          NA      3034.013      3442.218        2298.203        2688.6
2484 1.00000000          NA     21654.119     22322.279       18049.090       15303.1
```

All the peaks in the sigPeaksTable are found to be significant for the user-defined Fisher's exact test (note that the column named Fisher.test in the sigPeaksTable refers to the Fisher LSD test performed after an ANOVA test and not to the user-defined Fisher's exact test). This means that If the peak mass related to the fragmentation of the metabolite we are looking for is found in this table, the metabolite would statistically show a different absence/presence behaviour across the classes (WT/KO).

On the other hand, in the call to the spectralSigFeatures function, the argument test.fun contains the function of the user-defined test and the argument namefun is an optional parameter that contains the name of the user-defined function. This name will appear in the parameters slot of the MAIT-class object:

```
R> summary(MAIT_Fisher)
A MAIT object built of 12 samples and 2640 peaks.
No peak aggregation technique has been applied
18 of these peaks are statistically significant
The object contains 6 samples of class KO
The object contains 6 samples of class WT


Parameters of the analysis:
                                     Value
dataDir                              "Data"
snThres                              "2"
Sigma                                "2.12332257516562"
mzSlices                             "0.3"
retcorrMethod                        "loess"
groupMethod                          "density"
bwGroup                              "3"
mzWidGroup                           "0.25"
filterMethod                         "matchedFilter"
```

```
rtStep                               "0.03"
nSlaves                              "0"
project                              "MAIT_Demo"
ppm                                  "10"
minfrac                              "0.5"
fwhm                                 "30"
family1                              "gaussian"
family2                              "symmetric"
span                                 "0.2"
centWave peakwidth1                  "5"
centWave peakwidth2                  "20"
corrWithSamp                         "0.7"
corrBetSamp                          "0.75"
perfwhm                              "0.6"
sigma                                "6"
peakAnnotation pvalue                "0.05"
calcIso                              "TRUE"
calcCiS                              "TRUE"
calcCaS                              "TRUE"
graphMethod                          "hcs"
annotateAdducts                      "TRUE"
peakAggregation method               "None"
peakAggregation PCAscale             "FALSE"
peakAggregation PCAcenter            "FALSE"
peakAggregation scale                "FALSE"
peakAggregation RemoveOnePeakSpectra "FALSE"
fisher's test p-value                "0.05"
fisher's test p-value p.adj          "none"
```

The multiple test corrections are also implemented in this case by changing the p.adj argument of the function:

```
R> MAIT_Fisher <- spectralSigFeatures(MAIT.object = MAIT, test.fun = ftest,
namefun = "fisher's test", p.adj = "fdr")
```

```
Warning message:
In spectralSigFeatures(MAIT.object = MAIT, test.fun = ftest, namefun =
"fisher’s test", : No significative features found with the selected parameters.
```

In this particular case, a warning is thrown as no significant features were found with a false discovery rate-adjusted p-value lower or equal 0.05.

B.7.5 Statistical Plots

Out of 2,402 features, 106 were found to be statistically significant. At this point, several MAIT functions can be used to extract and visualise the results of the analysis. Functions plotBoxplot, plotHeatmap, plotPCA and plotPLS automatically generate boxplots, heat maps PCA score plot and PLS score plot files in the project folder when they are applied to a MAIT object (see Table B.3).

```
R> plotBoxplot(MAIT)
R> plotHeatmap(MAIT)
R> MAIT<-plotPCA(MAIT)
R> MAIT<-plotPLS(MAIT)
```

The plotPCA and plotPLS functions produce MAIT objects with the corresponding PCA and PLS models saved inside. The models, loadings and scores can be retrieved from the MAIT objects by using the functions model, loadings and scores:

```
R> PLSmodel <- model(x=MAIT, type = "PLS")
R> PCAmodel <- model(x=MAIT, type = "PCA")
R> PLSscores <- scores(x=MAIT,model="PLS")
R> PCAscores <- scores(x=MAIT,model="PCA")
R> PLSloadings <- loadings(x=MAIT,model="PLS")
R> PCAloadings <- loadings(x=MAIT,model="PCA")
```

```
R> PLSscores
      Comp 1
1   8.460117
```

```
2   8.238226
3   7.465394
4   6.341839
5   4.958885
6   5.887925
7  -6.577803
8  -6.570983
9  -6.660059
10 -6.363424
11 -7.427228
12 -7.752889
attr(,"class")
[1] "scores"
```

```
R> PCAscores[,1:3]
            PC1         PC2        PC3
 [1,] -8.758728  0.92480221 -6.1406083
 [2,] -8.348530 -0.86569846  0.1783953
 [3,] -7.570347  0.32825445 -1.6159867
 [4,] -6.209758 -0.01281555  3.1104855
 [5,] -4.632576 -0.80459247  5.6779015
 [6,] -5.757966 -0.47710433  0.8561668
 [7,]  6.483476  7.10158291  0.9827710
 [8,]  6.508645  0.44504996 -1.2287543
 [9,]  6.568818  3.66149693 -0.2422269
[10,]  6.311563 -1.97819990 -0.8625683
[11,]  7.518147 -5.26076372 -0.8812214
[12,]  7.887257 -3.06201203  0.1656458
```

```
R> head(matrix(PLSloadings))
            [,1]
[1,]  0.11179158
[2,]  0.10718688
[3,]  0.10167223
```

```
[4,]  0.10124325
[5,] -0.09481443
[6,]  0.10828112
```

```
R> head(PCAloadings[,1:3])
            PC1          PC2         PC3
[1,] -0.1129682  0.008376894 -0.14442144
[2,] -0.1080615 -0.002674411 -0.14786276
[3,] -0.1027608 -0.006700719 -0.10304058
[4,] -0.1009138 -0.010796632  0.09038020
[5,]  0.0950440 -0.212358347 -0.06243794
[6,] -0.1098603  0.054060752 -0.16588612
```

All the output figures are saved in their corresponding subfolders contained in the project folder. The names of the folders for the boxplots, heat maps and score plots are Boxplots, Heatmaps, PCA_Scoreplots and PLS_Scoreplots respectively. Figures B.3 and B.4 depict a heat map, a PCA score plot and a PLS score plot created when functions plotHeatmap, plotPCA and plotPLS were launched. Inside the R session, the project folder is recovered by typing:

```
R> resultsPath(MAIT)
```

B.7.6 Biotransformations

Before identifying the metabolites, peak annotation can be improved using the function Biotransformations to make interpreting the results easier. The MAIT package uses a default biotransformations table, but another table can be defined by the user and introduced by using the bioTable function input variable. The biotransformations table that MAIT uses is saved inside the file MAITtables.RData, under the name biotransformationsTable.

```
R> MAIT <- Biotransformations(MAIT.object = MAIT, peakPrecision = 0.005,
```

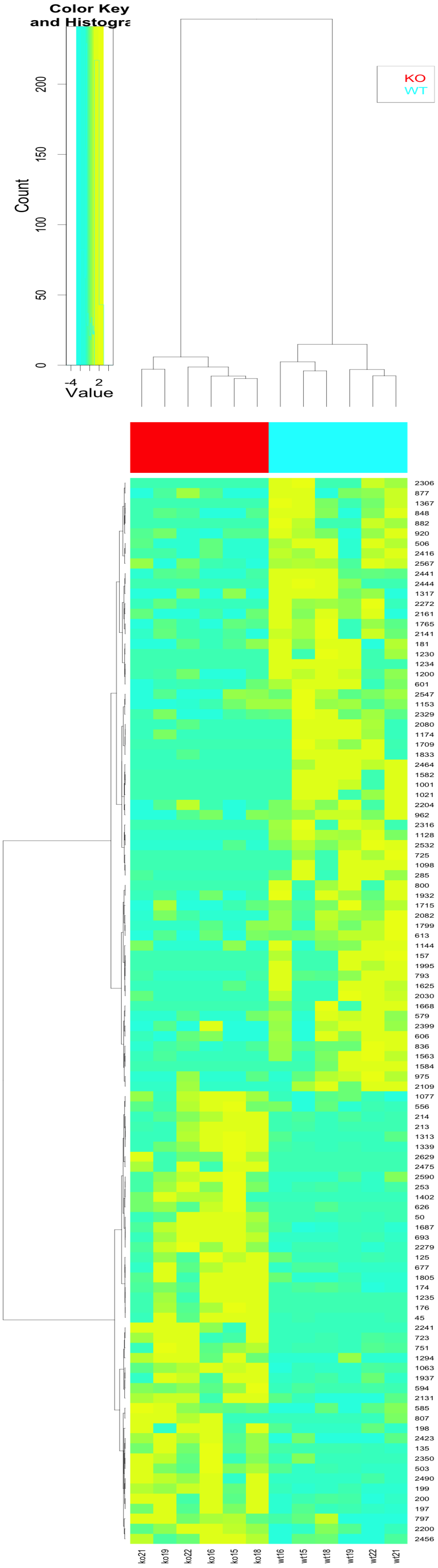


Figure B.3: Heat map created by the function plotHeatmap. Row numbers refer to spectra numbers.

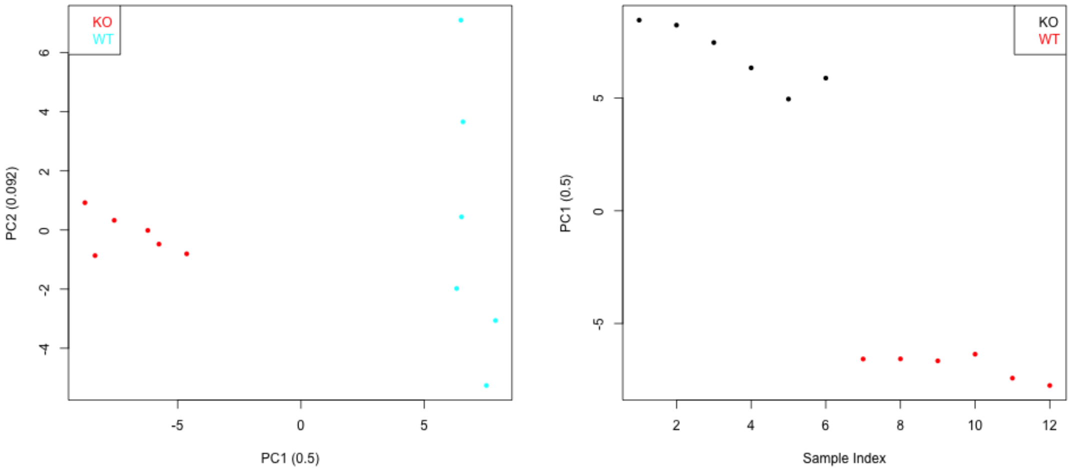


Figure B.4: PCA and PLS score plots (left and right plots respectively) generated by functions plotPCA and plotPLS. The PLS decomposition in this case has just one principal component.

```
adductAnnotation=FALSE)


WARNING: No input biotransformations table was given. Selecting default
MAIT table for biotransformations...
WARNING: No input adduct/fragment table was given. Selecting default MAIT
table for positive polarity...
Set adductTable equal to negAdducts to use the default MAIT table for negative polarity


  % Annotation in progress: 10  20  30  40  60  70  80  90  100
```

The Biotransformations function can also annotate adducts by setting the flag adductAnnotation as TRUE. This is useful when analysing peak data that come from an external source (i.e. peaks and spectra have not been detected by MAIT).

Building a user-defined biotransformations table from the MAIT default table or adding a new biotransformation is straightforward. For example, let's say we want to add a new adduct called ”custom_biotrans” whose mass loss is 105.

```
 R> data(MAITtables)
 R> myBiotransformation<-c("custom_biotrans",105.0)
 R> myBiotable<-biotransformationsTable
 R> myBiotable[,1]<-as.character(myBiotable[,1])
 R> myBiotable<-rbind(myBiotable,myBiotransformation)
 R> myBiotable[,1]<-as.factor(myBiotable[,1])
 R> tail(myBiotable)
```

```
                               NAME  MASSDIFF
45      glucuronide conjugation 176.0321
46 hydroxylation + glucuronide 192.0270
47              GSH conjugation 305.0682
48  2x glucuronide conjugation 352.0642
49                        [C13]   1.0034
50              custom_biotrans    105.0
```

To build an entire new biotransformations table, you only need to follow the format of the biotransformationsTable, which means writing the name of the biotransformations as factors in the NAME field of the data frame and their corresponding mass losses in the MASSDIFF field.

B.7.7 Metabolite Identification

Once the biotransformations annotation step is finished, the significant features have been enriched with a more specific annotation. The annotation procedure performed by the Biotransformations() function never replaces the peak annotations already done by other functions. MAIT considers the peak annotations to be complementary; therefore, when new annotations are detected, they are added to the current peak annotation and the identification function may be launched to identify the metabolites corresponding to the statistically significant features in the data.

```
R> MAIT <- identifyMetabolites(MAIT.object = MAIT, peakTolerance = 0.005,
 polarity="positive")
```

```
 WARNING: No input database table was given.
 Selecting default MAIT database...
 Metabolite identification initiated
```

```
% Metabolite identification in progress: 10  20  30  40  50  60  70
80  90  100
Metabolite identification finished
```

By default, the function identifyMetabolites() looks for the peaks of the significant features in the MAIT default metabolite database. The input parameter peakTolerance defines the tolerance between the peak and a database compound to be considered a possible match. It is set to 0.005 mass/charge units by default. The argument polarity, refers to to the polarity in which the samples were taken (positive or negative). It is set to "positive" by default but it should be adjusted changed to "negative" if the samples were recorded in negative polarisation mode. To check the results easily, function identifyMetabolites creates a table containing the significant feature characteristics and the possible metabolite identifications. Such a table is recovered from the MAIT-class object using the instruction:

```
R> metTable <- metaboliteTable(MAIT)
R> head(metTable)
```

```
  Query Mass Database Mass (neutral mass)    rt Isotope Adduct                Name spectra
1      300.2                      Unknown 56.36                            Unknown      27
2      588.2                      Unknown 46.65                            Unknown      91
3      537.4                      Unknown 64.41                            Unknown    1869
4      451.2                   450.193634 61.88                  Geranylgeranyl-PP    1891
5      325.2                      Unknown 60.95                            Unknown    1901
6      395.1                      Unknown 51.19                            Unknown    1921

       Biofluid     ENTRY       p.adj           p Fisher.Test Mean Class KO Mean Class WT
1       unknown   unknown 0.017482939 0.017482939          NA  2258350.1365    128461.054
2       unknown   unknown 0.193607894 0.193607894          NA     1998.5050     28919.323
3       unknown   unknown 0.024657677 0.024657677          NA      521.9275      3261.594
4 Not Available HMDB04486 0.003172073 0.003172073          NA     8853.1464      1629.177
5       unknown   unknown 0.019582285 0.019582285          NA     7781.1248     16818.493
6       unknown   unknown 0.025496645 0.025496645          NA     1463.7786      6408.485
```

```
  Median Class KO Median Class WT KO WT        ko15        ko16        ko18       ko19
1    2769931.3564     115642.2922  6  3 4005711.400 3115027.656 2726906.080 2812956.63
2          0.0000      10033.2150  2  4       0.000       0.000       0.000       0.00
3          0.0000       3751.3050  1  3       0.000       0.000       0.000       0.00
4       9644.3125        835.6261  5  0   10878.315    1943.378   12670.240    9634.14
5       7676.3250      17783.4658  5  6    9563.384    7485.395    3538.465   11418.24
6        900.5959       6702.1125  0  4       0.000    1801.192    3595.172       0.00
       ko21        ko22      wt15      wt16      wt18       wt19       wt21       wt22
1 57169.450  832329.600 192385.450 94036.332 48410.145 137248.252 213368.607  85317.540
2  2837.345    9153.685  40378.565     0.000     0.000   6696.635  13369.795 113070.941
3     0.000    3131.565   3306.845     0.000  4255.525   1844.086   4195.765   5967.345
4  8338.320    9654.485   1671.252  3877.383     0.000      0.000   4226.428      0.000
5  6814.010    7867.255  17009.985 18556.947 27223.175   7555.820  11949.359  18615.675
6  3386.308       0.000   4895.743  9045.700 11105.240   5371.080      0.000   8033.145
```

This table provides useful results about the analysis of the samples, such as the P-value of the statistical test, its adduct or isotope annotation and the name of any possible hit in the database. Note that if no metabolite has been found in the database for a certain feature, it is labelled as ”unknown” in the table. The table also includes the median and mean values per class and feature.

B.7.8 Validation

Finally, we will use the function Validation() to check the predictive value of the significant features. All the information related to the output of the Validation() function is saved in the project directory in a folder called ”Validation”. Two boxplots showing the overall and per class classification ratios are created, along with every confusion matrix corresponding to each iteration (see Table B.3).

```
R> MAIT <- Validation(Iterations = 20, trainSamples= 3, MAIT.object = MAIT)
```

```
Iteration 1 done
Iteration 2 done
Iteration 3 done

...
```

```
 Iteration 19 done
 Iteration 20 done
```

A summary of a MAIT object, which includes the overall classification values, can be accessed:

```
 R> summary(MAIT)
```

```
A MAIT object built of 12 samples and 2640 peaks. No peak aggregation technique has been applied
106 of these peaks are statistically significant
The object contains 6 samples of class KO
The object contains 6 samples of class WT
The Classification using 3 training samples and 20 Iterations gave the results:

               KNN PLSDA SVM
mean             1     1   1
standard error   0     0   0


Parameters of the analysis:
                                     Value
dataDir                             "Data"
snThres                             "2"
Sigma                               "2.12332257516562"
mzSlices                            "0.3"
retcorrMethod                       "loess"
groupMethod                         "density"
bwGroup                             "3"
mzWidGroup                          "0.25"
filterMethod                        "matchedFilter"
rtStep                              "0.03"
nSlaves                             "0"
project                             "MAIT_Demo"
ppm                                 "10"
minfrac                             "0.5"
fwhm                                "30"
family1                             "gaussian"
family2                             "symmetric"
span                                "0.2"
```

```
centWave peakwidth1                  "5"
centWave peakwidth2                  "20"
corrWithSamp                         "0.7"
corrBetSamp                          "0.75"
perfwhm                              "0.6"
sigma                                "6"
peakAnnotation pvalue                "0.05"
calcIso                              "TRUE"
calcCiS                              "TRUE"
calcCaS                              "TRUE"
graphMethod                          "hcs"
annotateAdducts                      "TRUE"
peakAggregation method               "None"
peakAggregation PCAscale             "FALSE"
peakAggregation PCAcenter            "FALSE"
peakAggregation scale                "FALSE"
peakAggregation RemoveOnePeakSpectra "FALSE"
Welch pvalue                         "0.05"
Welch p.adj                          "none"
peakPrecision                        "0.005"
Biotransformations adductAnnotation  "0"
peakTolerance                        "0.005"
polarity                             "positive"
Validation Iterations                "20"
Validation trainSamples              "3"
Validation PCAscale                  "0"
Validation PCAcenter                 "1"
Validation RemoveOnePeakSpectra      "0"
Validation tuneSVM                   "0"
Validation scale                     "1"
PCA data logarithm                   "FALSE"
PCA data centered                    "TRUE"
PCA data scaled                      "TRUE"
```

It is also possible to gather the classification ratios per class, classifier used and iteration number by using the function classifRatioClasses():

```
R> classifRatioClasses(MAIT)
```

The classification ratios are 100% in all the iterations; the set of significant features separates the samples belonging to these classes.

B.7.9 Using External Peak Data

Taking advantage of the modularised design of MAIT, it is possible to use the function MAITbuilder to import peak data and analyse it using the MAIT statistical functions. As stated in section B.6.5, there are certain arguments that should be provided depending on which function is wanted to be launched. In this section we will show an example of this data importation procedure using the same data that we have been using in the tutorial so far. Let's say we have a peak table recorded in positive polarisation mode with the peak masses and retention time values such as:

```
R> peaks <- scores(MAIT)
R> masses <- getPeaklist(MAIT)$mz
R> rt <- getPeaklist(MAIT)$rt/60
```

We want to perform an annotation stage and metabolite identification on these data. To that end, we can launch the function MAITbuilder to build a MAIT-class object with the data in the table:

```
R> importMAIT <- MAITbuilder(data = peaks, masses = masses, rt = rt,
significantFeatures = TRUE, spectraEstimation = TRUE, rtRange=0.2,corThresh=0.7)
```

We have selected the option spectraEstimation as TRUE because we do not know the grouping of the peaks into spectra. As we want to annotate and identify all the peaks in the data frame, we set the flag significantFeatures to TRUE. At this point, we can launch the Biotransformations function:

```
R> importMAIT <- Biotransformations(MAIT.object = importMAIT, adductAnnotation = TRUE,
peakPrecision = 0.005, adductTable = NULL)
```

We set the adductAnnotation flag to TRUE as we want to perform an adduct annotation step. The parameter adductTable set to NULL implies that a positive polarisation adduct annotation stage will be performed. To run a negative annotation, the argument should be set to negAdducts. The metabolite identification stage is launched as in the previous case:

```
R> importMAIT <- identifyMetabolites(MAIT.object = importMAIT, peakTolerance=0.005,
polarity="positive")
```

The annotation of the Biotransformations and the adducts is given in the Adduct field of the metabolite table. The identification procedure can be performed for LC/MS data gathered in negative polarisation mode by setting polarity = ”negative”. If the class information is also introduced in the MAITbuilder, it is also possible to launch the computation of statistical tests (through function spectralSigFeatures), the validation and the functions regarding the statistical plots and models.

B.8 Conclusions

MAIT package is a new R package that analyses LC/MS metabolomic data files. The package provides functions yielding a programmable environment that is especially focused on performing an end-to-end metabolomic analysis. Special emphasis is given to peak annotation and statistical result validation using a predictive approach. MAIT also supports peak aggregation techniques to improve the predictive power of the features. The package is capable of producing a set of post-processing plots, such as PCA score plots, and summary tables to evaluate the results of the analysis. In short, MAIT is an easy, quick-to-use package for performing a complete automatic analysis of LC/MS metabolomic data files.

Bibliography

[1] Georgios a Theodoridis, Helen G Gika, Elizabeth J Want, and Ian D Wilson. Liquid chromatography-mass spectrometry based global metabolite profiling: a review. *Analytica chimica acta*, 711:7–16, January 2012. ISSN 1873-4324. doi: 10.1016/j.aca.2011.09.042.

[2] Sara Tulipani, Rafael Llorach, Olga Jáuregui, Patricia López-Uriarte, Mar Garcia-Aloy, Mònica Bullo, Jordi Salas-Salvadó, and Cristina Andrés-Lacueva. Metabolomics Unveils Urinary Changes in Subjects with Metabolic Syndrome following 12-Week Nut Consumption. *Journal of Proteome Research*, 2011. ISSN 15353907. doi: 10.1021/pr200514h. URL http://www.ncbi.nlm.nih.gov/pubmed/21905751.

[3] C A Smith, E J Want, G O'Maille, R Abagyan, and G Siuzdak. XCMS: processing mass spectrometry data for metabolite profiling using nonlinear peak alignment, matching, and identification. *Analytical Chemistry*, 78(3):779–787, 2006. ISSN 00032700. doi: 10.1021/ac051437y. URL http://pubs3.acs.org/acs/journals/doilookup?in_doi=10.1021/ac051437y.

[4] Ralf Tautenhahn, Christoph Böttcher, and Steffen Neumann. Highly sensitive feature detection for high resolution LC/MS. *BMC Bioinformatics*, 9(1):504, 2008. URL http://www.ncbi.nlm.nih.gov/pubmed/19040729.

[5] H Paul Benton, Elizabeth J Want, and Timothy M D Ebbels. Correction of mass calibration gaps in liquid chromatography-mass spectrometry metabolomics data. *Bioinformatics*, 26 (19):2488–2489, 2010. URL http://www.ncbi.nlm.nih.gov/pubmed/20671148.

[6] Carsten Kuhl, Ralf Tautenhahn, and Steffen Neumann. LC-MS Peak Annotation and Identification with CAMERA. *Camera*, pages 1–14, 2011.

[7] Arnald Alonso, Antonio Julia, Antoni Beltran, Maria Vinaixa, Marta Díaz, Lourdes Ibañez, Xavier Correig, and Sara Marsal. Astream: an r package for annotating lc/ms metabolomic data. *Bioinformatics*, 27(9):1339–1340, 2011. URL http://www.ncbi.nlm.nih.gov/pubmed/21414990.

[8] Mikko Katajamaa, Jarkko Miettinen, and Matej Oresic. MZmine: toolbox for processing and visualization of mass spectrometry based molecular profile data. *Bioinformatics (Oxford, England)*, 22(5):634–636, 2006. ISSN 1367-4803. doi: 10.1093/bioinformatics/btk039.

[9] Tomás Pluskal, Sandra Castillo, Alejandro Villar-Briones, and Matej Oresic. MZmine 2: modular framework for processing, visualizing, and analyzing mass spectrometry-based molecular profile data. *BMC bioinformatics*, 11:395, 2010.

[10] Richard A Scheltema, Andris Jankevics, Ritsert C Jansen, Morris A Swertz, and Rainer Breitling. PeakML/mzMatch: a file format, Java library, R library, and tool-chain for mass spectrometry data analysis. *Analytical chemistry*, 83(7):2786–2793, 2011.

[11] Jianguo Xia, Nick Psychogios, Nelson Young, and David S. Wishart. Metaboanalyst: a web server for metabolomic data analysis and interpretation. *Nucleic Acids Research*, 37(suppl 2):W652–W660, 2009. doi: 10.1093/nar/gkp356. URL http://nar.oxfordjournals.org/content/37/suppl_2/W652.abstract.

[12] Jianguo Xia, Rupasri Mandal, Igor V Sinelnikov, David Broadhurst, and David S Wishart. MetaboAnalyst 2.0–a comprehensive server for metabolomic data analysis. *Nucleic acids research*, 40(Web Server issue):W127–33, July 2012. ISSN 1362-4962. doi: 10.1093/nar/gks374. URL http://nar.oxfordjournals.org/cgi/content/long/gks374v1.

[13] Rolf Danielsson, Dan Bylund, and Karin E Markides. Matched filtering with background suppression for improved quality of base peak chromatograms and mass spectra in liquid chromatography-mass spectrometry. *Analytica Chimica Acta*, 454(2):167–184, 2002. URL http://linkinghub.elsevier.com/retrieve/pii/S0003267001015744.

[14] David S Wishart, Craig Knox, An Chi Guo, Roman Eisner, Nelson Young, Bijaya Gautam, David D Hau, Nick Psychogios, Edison Dong, Souhaila Bouatra, and et al. Hmdb: a knowledgebase for the human metabolome. *Nucleic Acids Research*, 37(Database issue):D603–D610, 2009. URL http://www.pubmedcentral.nih.gov/articlerender.fcgi?artid=2686599&tool=pmcentrez&rendertype=abstract.

[15] T. Hastie, R. Tibshirani, and J. H. Friedman. *The Elements of Statistical Learning*. Springer, corrected edition, July 2003. ISBN 0387952845. URL http://www.worldcat.org/isbn/0387952845.

[16] Francesc Fernández-Albert, Rafael Llorach, Cristina Andrés-Lacueva, and Alexandre Perera-Lluna. Peak Aggregation as an Innovative Strategy for Improving the Predictive Power of LC-MS Metabolomic Profiles. *Analytical chemistry*, 2014. ISSN 1520-6882. doi: 10.1021/ac403702p. URL http://www.ncbi.nlm.nih.gov/pubmed/24471770.

[17] Daniel Adler and Duncan Murdoch. *rgl: 3D visualization device system (OpenGL)*, 2012. URL http://CRAN.R-project.org/package=rgl. R package version 0.92.894.

[18] Alan Saghatelian, Sunia A Trauger, Elizabeth J Want, Edward G Hawkins, Gary Siuzdak, and Benjamin F Cravatt. Assignment of endogenous substrates to enzymes by global metabolite profiling. *Biochemistry*, 43(45):14332–14339, 2004. URL http://www.ncbi.nlm.nih.gov/pubmed/15533037.

[19] Colin A. Smith. *faahKO: Saghatelian et al. (2004) FAAH knockout LC/MS data*, 2012. URL http://dx.doi.org/10.1021/bi0480335. R package version 1.2.13.

Appendix C

intCor vignette document

C.1 abstract

Liquid Chromatography coupled to mass Spectrometry (LC/MS) has become widely used in Metabolomics. Several artefacts have been identified during the acquisition step in large LC/MS metabolomics experiments, including ion suppression, carryover or changes in the sensitivity and intensity. Several sources have been pointed out as responsible for these effects. In this context, the drift effects of the peak intensity is one of the most frequent and may even constitute the main source of variance in the data, resulting in misleading statistical results when the samples are analysed. In this paper, we propose the introduction of a methodology based on a common variance analysis prior to the data normalisation to address this issue. This methodology was tested and compared with four other methods by calculating the Dunn and Silhouette indices of the Quality Control classes. The results showed that our proposed methodology performed better than any of the other four methods. As far as we know, this is the first time that this kind of approach has been applied in the metabolomics context.

C.2 Using intCor

intCor is an R package focused on drift removal and data normalisation for LC/MS metabolomic data. The package includes five different methods to correct drift effects in the data. It is mainly based on two functions, one for data importation and a second one for correcting the drift effects. Functions to perform graphical analyses (PCA, heat map plots) and to create output cdf files are also included in the package.

C.2.1 Importing data

The intCor package uses the function importData to read the samples or data sets that contain the data to be analysed. This function, accepts three different input formats: a set of external files (e.g. cdf or mzXML), a matrix or an xcmsSet object. The output of the package could be either a set of cdf files (if the input of the importData was a set of files) or a data matrix (regardless of the type of input).

C.2.1.1 Using an External Data Matrix

One possibility is to import the data through a data table and a class vector. The data matrix should have the samples as columns and variables (time or masses) as rows. The class vector should be a character with the class names. Each one of the components in the vector should have the same ordering than the columns of the data matrix (e.g. the first vector component should refer to the class of the first column of the data matrix). intCor contains the data used as cdf in the previous subsection as an RData:

```
R> data(intCorData)
```

The normInt object might be constructed as follows:

```
 R> intCor_table<-importData(data=dataMatrix,classes=classes)

The data matrix provided includes peak masses or not? (please answer yes or no)
1: no
Read 1 item
Starting the class-wise outlier detection...
Outlier detection for class Reference in progress...
Outlier removal in drift training data in progress...
Possible score Outliers: 1 2 3 5 6 7 9 10 36
Possible orthogonal Outliers: 1 6

Select sample ID number to be removed
1: 1
2: 6
```

```
3:
Read 2 items
Possible score Outliers: 1 2 3 4 5 6 7 8 34
Possible orthogonal Outliers: 2 34

Select extra sample ID number to be removed
1:
Read 0 items
Outlier removal finished
Outlier Removal for class Reference done

Outlier detection for class Water in progress...
Outlier removal in drift training data in progress...
Possible score Outliers: 1 2 3 4 5 6 7 8 12 15 16 22 23 56 86 89
Possible orthogonal Outliers: 15 23 68

Select sample ID number to be removed
1: 15
2: 23
3: 68
4:
Read 3 items
Possible score Outliers: 1 2 3 4 5 6 7 8 15 21 29 46 54 79 83 84 86
Possible orthogonal Outliers: 83

Select extra sample ID number to be removed
1:
Read 0 items
Outlier removal finished
Outlier Removal for class Water done

Outlier detection for class QC in progress...
Outlier removal in drift training data in progress...
Possible score Outliers: 1 2 3 8 46 47 48
Possible orthogonal Outliers: 47

Select sample ID number to be removed
1: 47
```

```
2:
Read 1 item
Possible score Outliers: 1 2 3 4 8 9 10 11 47
Possible orthogonal Outliers:

Select extra sample ID number to be removed
1:
Read 0 items
Outlier removal finished
Outlier Removal for class QC done

Outlier Removal stage QC finished!
```

C.2.1.2 Using an xcmsSet object

intCor also supports using a xcmsSet object (from the XCMS package [1]) as an input for the function importData. In this case, if the class information is not provided (argument classes of the function importData), the class assignment would be retrieved from the xcmsSet object. intCor also includes an xcmsSet object of the provided sample files:

```
R> data(intCorXCMS)
R> xcg
An "xcmsSet" object with 180 samples

Time range: 17.7-451.2 seconds (0.3-7.5 minutes)
Mass range: 70.0626-683.9618 m/z
Peaks: 99333 (about 552 per sample)
Peak Groups: 526
Sample classes: QC, Reference, Water

Profile settings: method = bin
                  step = 0.1

Memory usage: 8.79 MB
```

The data importation in this case can be computed as:

```
R> normInt_xcms<-importData(data=xcg)
Starting the class-wise outlier detection...
Outlier detection for class QC in progress...
Outlier removal in drift training data in progress...
Possible score Outliers: 1 2 3 4 5 6 7 8 9 10 11 12 19
Possible orthogonal Outliers: 2

Select sample ID number to be removed
1: 2
2:
Read 1 item
Possible score Outliers: 1 2 4 5 6 7
Possible orthogonal Outliers: 1 2 3 4 7 18

Select extra sample ID number to be removed
1:
Read 0 items
Outlier removal finished
Outlier Removal for class QC done

Outlier detection for class Reference in progress...
Outlier removal in drift training data in progress...
Possible score Outliers: 1 2 3 4 5 6 9 10 16 27 36
Possible orthogonal Outliers: 1 5 6 10 16

Select sample ID number to be removed
1: 1
2: 5
3: 6
4: 10
5: 16
6:
Read 5 items
Possible score Outliers: 1 2 3 4 5 6 7 22 31
Possible orthogonal Outliers: 1 3 22 31

Select extra sample ID number to be removed
1:
```

```
Read 0 items
Outlier removal finished
Outlier Removal for class Reference done

Outlier detection for class Water in progress...
Outlier removal in drift training data in progress...
Possible score Outliers: 3 4 8 17 18 19 20 23 24 27 28 29 30 33 42 57 58 85 86 91 92 93 94
Possible orthogonal Outliers: 4

Select sample ID number to be removed
1: 4
2:
Read 1 item
Possible score Outliers: 3 6 7 22 23 26 27 28 29 35 40 41 84 85
Possible orthogonal Outliers:

Select extra sample ID number to be removed
1:
Read 0 items
Outlier removal finished
Outlier Removal for class Water done

Outlier Removal stage Water finished!
```

C.2.1.3 Using Sample files

When cdf or mzXML files are provided, the intCor package performs the drift correction on the chromatograms of the samples. To run the following example, download the data available in our website (http://b2slab.upc.edu/software-and-downloads/intensity-drift-correction/). The zip file contains the data of three different classes (QC, Reference and Water) in cdf format along with a csv file containing the metadata of the samples. If we set the R working directory to where the file sampTable.csv is, we can take a look on the provided metadata:

```
R> tab <- read.csv("sampTable.csv")
R> head(tab)
```

```
                                         FileName     Date  Time     Class     Column Batch
1 DM_20121211_POS_pl4_1-M00000000_0_X_X_POS_pl4 12/11/12 14:41 Reference LC-014-MET     4
2         DM_20121211_POS_pl4_1-MQCx1_1_POS_pl4 12/11/12 19:47     Water LC-014-MET     4
3         DM_20121211_POS_pl4_1-MQCx2_1_POS_pl4 12/11/12 20:02        QC LC-014-MET     4
4         DM_20121211_POS_pl4_1-MQC_1_1_POS_pl4 12/11/12 20:16     Water LC-014-MET     4
5         DM_20121211_POS_pl4_2-MQCx1_2_POS_pl4 12/12/12 01:23     Water LC-014-MET     4
6         DM_20121211_POS_pl4_2-MQCx2_2_POS_pl4 12/12/12 01:37        QC LC-014-MET     4
```

we can see that the data frame relates the sample IDs (column FileName) with information regarding the sample recording date and time (columns Date and Time), the class of the sample, the column ID of the chromatograph and the batch number of the sample. Only the columns relating the sample IDs and the classes are mandatory (columns FileName and Class) for the intCor package.

A normInt object is created by giving the name of the data frame (sampTable.csv) and the extension of the sample files (cdf) to the importData function. The function performs a class-wise user-supervised outlier removal stage by default (argument removeOutliers) that can be set as automatic (argument automaticOutlierRemoval). The function uses the Hotelling method to detect the outliers. For each class, the function gives an estimation of the score and orthogonal outliers and ask to the user for the samples to be removed. Once the sample removal is performed, the Hotelling distances are computed again to look for more outliers of that class. The loop ends when no outliers (blank input) is given. For example in the following, we are going to import the data through the cdf files and remove the samples 1 and 6 of the Reference class, the samples 15, 23 and 68 of the Water class and the sample 47 of the QC class.

```
R> library(intCor)

R> normInt<-importData(dataDir="data", fileType="cdf", tabName="sampTable.csv")

Reading input table...Done
Extraction of the Total Ion Chromatograms initiated...
 % done: 10  20  30  40  50  60  70  80  90  100  Extraction of the Total Ion Chromatograms F
Starting the class-wise outlier detection...
Outlier detection for class Reference in progress...
Outlier removal in drift training data in progress...
Possible score Outliers: 1 2 3 5 6 7 9 10 36
Possible orthogonal Outliers: 1 6

Select sample ID number to be removed
1: 1
2: 6
3:
Read 2 items
Possible score Outliers: 1 2 3 4 5 6 7 8 34
Possible orthogonal Outliers: 2 34

Select extra sample ID number to be removed
1:
Read 0 items
Outlier removal finished
Outlier Removal for class Reference done

Outlier detection for class Water in progress...
Outlier removal in drift training data in progress...
Possible score Outliers: 1 2 3 4 5 6 7 8 12 15 16 22 23 56 86 89
Possible orthogonal Outliers: 15 23 68

Select sample ID number to be removed
1: 15
2: 23
3: 68
4:
Read 3 items
Possible score Outliers: 1 2 3 4 5 6 7 8 15 21 29 46 54 79 83 84 86
Possible orthogonal Outliers: 83

Select extra sample ID number to be removed
1:
Read 0 items
Outlier removal finished
Outlier Removal for class Water done

Outlier detection for class QC in progress...
Outlier removal in drift training data in progress...
```

```
Possible score Outliers: 1 2 3 8 46 47 48
Possible orthogonal Outliers: 47

Select sample ID number to be removed
1: 47
2:
Read 1 item
Possible score Outliers: 1 2 3 4 8 9 10 11 47
Possible orthogonal Outliers:

Select extra sample ID number to be removed
1:
Read 0 items
Outlier removal finished
Outlier Removal for class QC done

Outlier Removal stage QC finished!
```

C.2.2 Correcting the drift in the data

Once the run of the importData function is finished, the data have been imported into a normInt object along with the metadata stored in the table:

```
R> normInt
normInt object with a retention time range of [0.333,7.502] seconds containing:
93 samples of the class Water
47 samples of the class QC
34 samples of the class Reference
The data has not been normalised
```

We can also retrieve the information regarding the outlier removal stage by running a summary of the object:

```
R> summary(normInt)
```

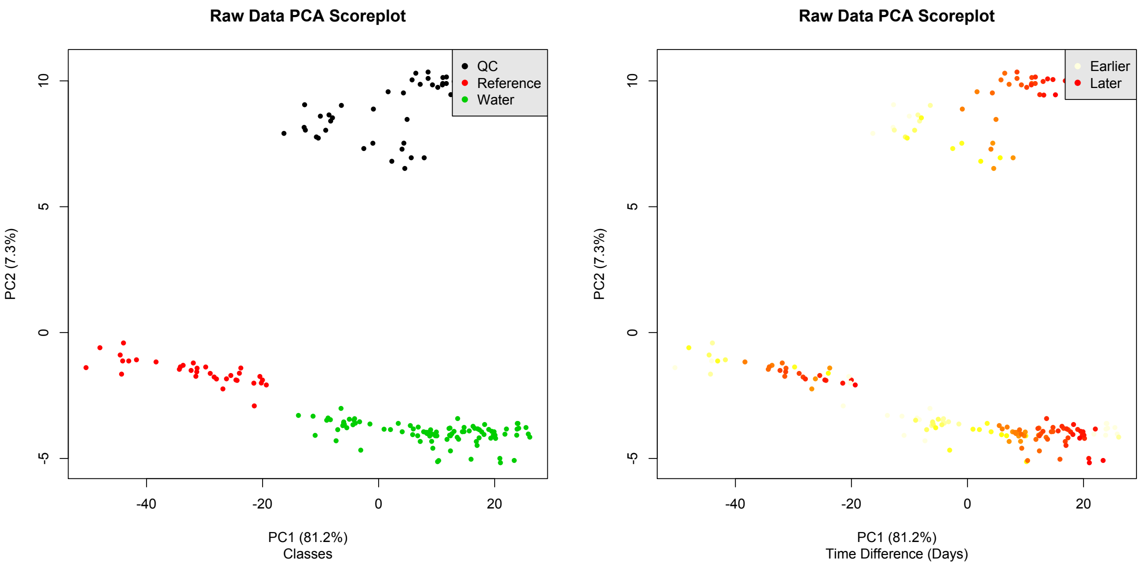


FIGURE C.1: Raw Data score plots. The left plot depicts the samples into the plane PC1-PC2 using the class labels whereas the right plot shows the same PCA score plot but using the time labels.

```
normInt object with retention a time range of [0.333,7.502] seconds containing:
93 samples of the class Water
47 samples of the class QC
34 samples of the class Reference

There were removed the following outliers:
2 samples of class Water
3 samples of class QC
1 samples of class Reference
The data has not been normalised
```

It is also straightforward to run a PCA Scoreplot of the data (see Figure C.1 for the PCA score plots regarding the raw data and their class and time labels) by running the pcaPlot function on the normInt object:

```
R> pcaPlot(normInt)
```

At this point, we can correct the drift effects in the data though the corrModel function. The function supports five different methods: Component Correction (function argument method=”cc”), Common Principal Components analysis (function argument method=”cpca”), Common Principal Components Analysis + Median Normalisation (function argument method=”cpcaMed”), Median Normalisation (function argument method=”medians”) and Batch compensation through the ComBat function (function argument method=”batch”) [2].

In the next command, we correct the data using the Common Principal Components Analysis + Median Normalisation method. One common principal component was selected for removal and all the classes were taken for generating the model:

```
normInt_cpcaMed_1C <- corrModel(normInt=normInt,method="cpcaMed",
             modClasses=c("Water","QC","Reference"),nComps=1)

Detected available classes to select for correction:
       QC Reference     Water
       47        34        93
Selected classes to compute the drift model:
[1] "Water"     "QC"        "Reference"

Computing CPC model...Model done!

Computing CPC explained variance...
CPC1
 0.7
Correcting Intensity Drift...
Computing clustering indices ...
Raw Data
, , kmeans

                      3
Connectivity 14.5682540
Dunn          0.0242528
Silhouette    0.5038728

, , hierarchical

                      3
Connectivity 3.64246032
Dunn         0.07857024
```

```
Silhouette   0.49989042


Corrected Data
, , kmeans


                     3
Connectivity 0.0000000
Dunn         0.9661138
Silhouette   0.8929673


, , hierarchical


                     3
Connectivity 0.0000000
Dunn         0.9661138
Silhouette   0.8929673
```

Figure C.2 depicts the PCA score plots for the corrected data set. The comparison between the clustering indices can be retrieved by running a summary of the normInt object:

```
R> summary(normInt_cpcaMed_1C)
normInt object with retention a time range of [0.333,7.502] seconds containing:
93 samples of the class Water
47 samples of the class QC
34 samples of the class Reference

There were removed the following outliers:
2 samples of class Water
3 samples of class QC
1 samples of class Reference

The data was normalised using the cpcaMed method
The variance captured for each component was
0.7 for component 1

The computed Dunn indices before and after the normalisation:
Raw Dunn index =  0.0242528
```

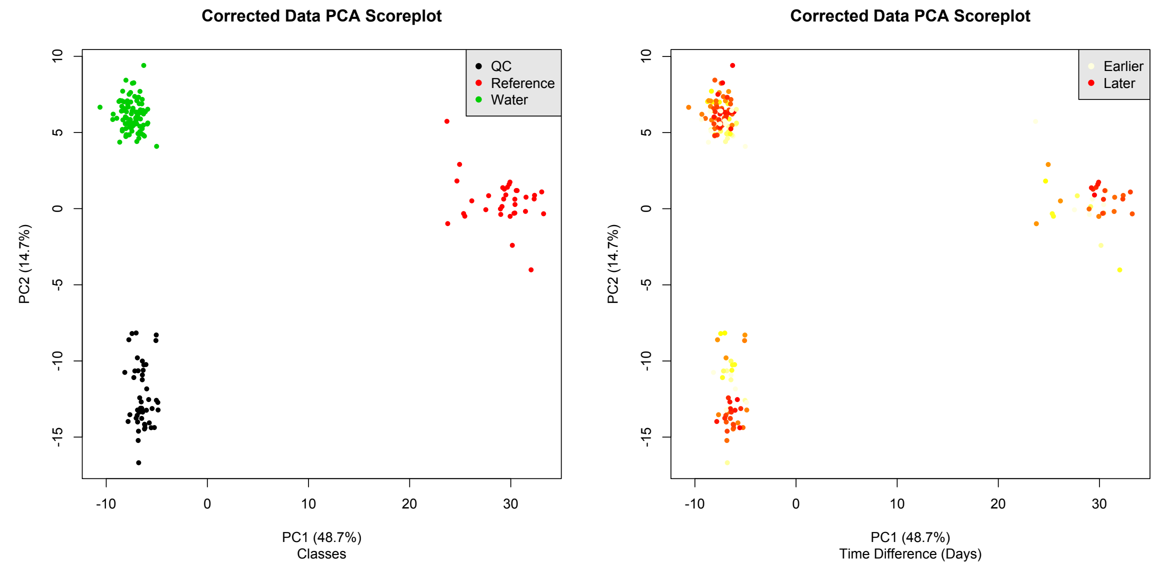


FIGURE C.2: PCA score plots of the corrected data using the Common Principal Components + Median Normalisation method. The class assignment (left) and the time elapsed since the first sample injection (right) were used as labelling.

```
Corrected Dunn index =  0.9661138


The computed Silhouette indices before and after the normalisation:

Raw Silhouette index =  0.5038728

Corrected Silhouette index =  0.8929673
```

The clustering indices show an important improvement after the data correction. The other methods are launched in a similar way, for example the method regarding the ComBat function:

```
normInt_comBat<-corrModel(normInt=normInt,method="batch")

Defining Model Matrix

Correcting Batch effects...Found 12 batches

Found 2  categorical covariate(s)

Standardizing Data across genes

Fitting L/S model and finding priors

Finding parametric adjustments

Adjusting the Data

Done

Computing explained variance...

[1] 0.881


Computing clustering indices ...
```

```
Raw Data
, , kmeans

                        3
Connectivity 14.5682540
Dunn          0.0242528
Silhouette    0.5038728

, , hierarchical

                        3
Connectivity 3.64246032
Dunn         0.07857024
Silhouette   0.49989042

Corrected Data
, , kmeans

                       3
Connectivity 0.0000000
Dunn         0.6363887
Silhouette   0.8027747

, , hierarchical

                       3
Connectivity 0.0000000
Dunn         0.6363887
Silhouette   0.8027747
```

gave the PCA score plots in the Figure C.3, or the medians method:

```
R> normInt_medians<-corrModel(normInt=normInt,method="medians")

Computing clustering indices ...
Raw Data
, , kmeans
```

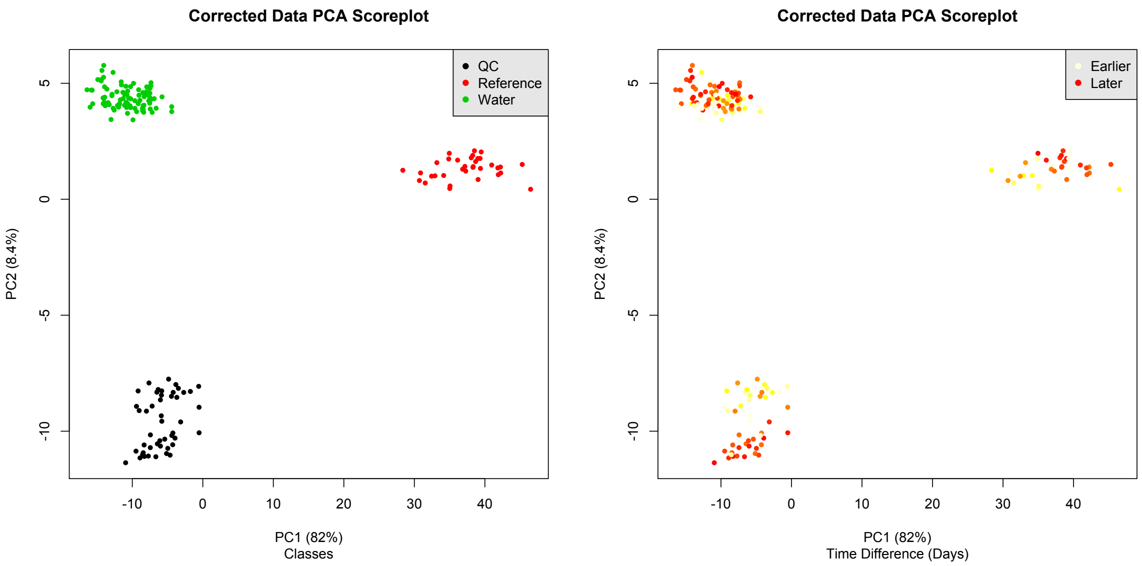


FIGURE C.3: PCA score plots of the corrected data using the Batch compensation (ComBat function) method. The class assignment (left) and the time elapsed since the first sample injection (right) were used as labelling.

```
                      3
Connectivity 14.5682540
Dunn          0.0242528
Silhouette    0.5038728


, , hierarchical


                      3
Connectivity 3.64246032
Dunn         0.07857024
Silhouette   0.49989042


Corrected Data
, , kmeans


                     3
Connectivity 0.0000000
Dunn         0.7873576
Silhouette   0.8622102


, , hierarchical
```

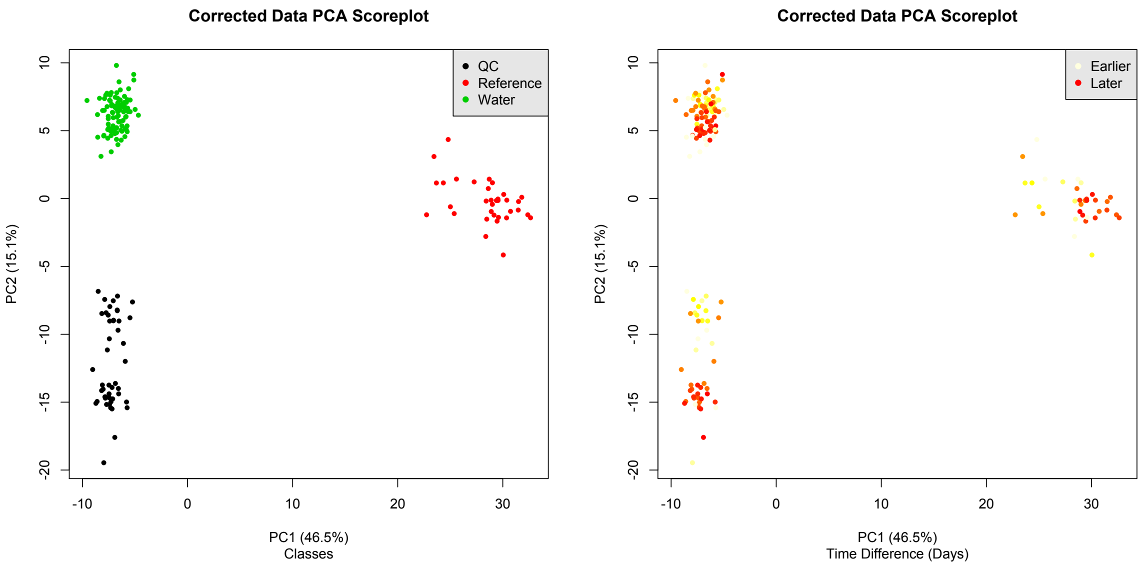


Figure C.4: PCA score plots of the corrected data using the Median Normalisation method. The class assignment (left) and the time elapsed since the first sample injection (right) were used as labelling.

```
                     3
Connectivity 0.0000000
Dunn         0.7873576
Silhouette   0.8622102
```

which gave the PCA score plots depicted in Figure C.4

C.2.3 intCor Output

If the data was imported through external files, there is the possibility of printing cdf files back as an output. For example, to print the cdf files of the corrected data through the cpcaMed method, we can use the function cdfFileCreator as:

```
R> cdfFileCreator(dataMatrix=getCorrData(normInt_cpcaMed_1C),dir2print="correctedCDFs",
               dataDir="data",fileType="cdf",scanRange=NULL)
```

Finally, another possibility that can be used regardless of the data importation method, is to retrieve the corrected table using the function getCorrData and (optionally) print a CSV file:

```
R> write.csv(file="corrData_cpcaMed_1C.csv",getCorrData(normInt_cpcaMed_1C))
```

Bibliography

[1] C A Smith, E J Want, G O'Maille, R Abagyan, and G Siuzdak. XCMS: processing mass spectrometry data for metabolite profiling using nonlinear peak alignment, matching, and identification. *Analytical Chemistry*, 78(3):779–787, 2006. ISSN 00032700. doi: 10.1021/ac051437y. URL http://pubs3.acs.org/acs/journals/doilookup?in_doi=10.1021/ac051437y.

[2] Jeffrey T. Leek, W. Evan Johnson, Hilary S. Parker, Andrew E. Jaffe, and John D. Storey. The sva package for removing batch effects and other unwanted variation in high-throughput experiments. *Bioinformatics*, 28(6):882–883, 2012.

Appendix D

Supporting information: Intensity drift removal in LC/MS metabolomics by Common Variance Compensation

D.1 Dataset size effects on the drift removal

To measure the effect of having a smaller dataset on the performance of the different methods we performed a random sub-sampling stage using different sample size. We picked 100 random sets for four different sample sizes (10%, 25%, 33% and 50% of the original dataset). For each one of these pics, it was performed a drift correction stage using all the tested methods and there were taken the Silhouette clustering indices. Figure 4 depicts a summary of the Silhouette index values for the five methods. For the methods involving a different number of components to model the drift (CC, CPCA and CPCA_Medians), there were taken the cases having highest mean Silhouette index. All the three methods showed best performance when three components were considered.

Table D.1 shows the results of fitting a linear model for each of the methods using the dataset size as a cofactor. We consider that a Slope value is different than 0 (null hypothesis) if the p-value is equal or less than 0.05.

Among the five methods tested, the CPCA and the CPCA + Median showed no variation depending on the dataset size. The CC and Median Fold Change method had a negative slope.

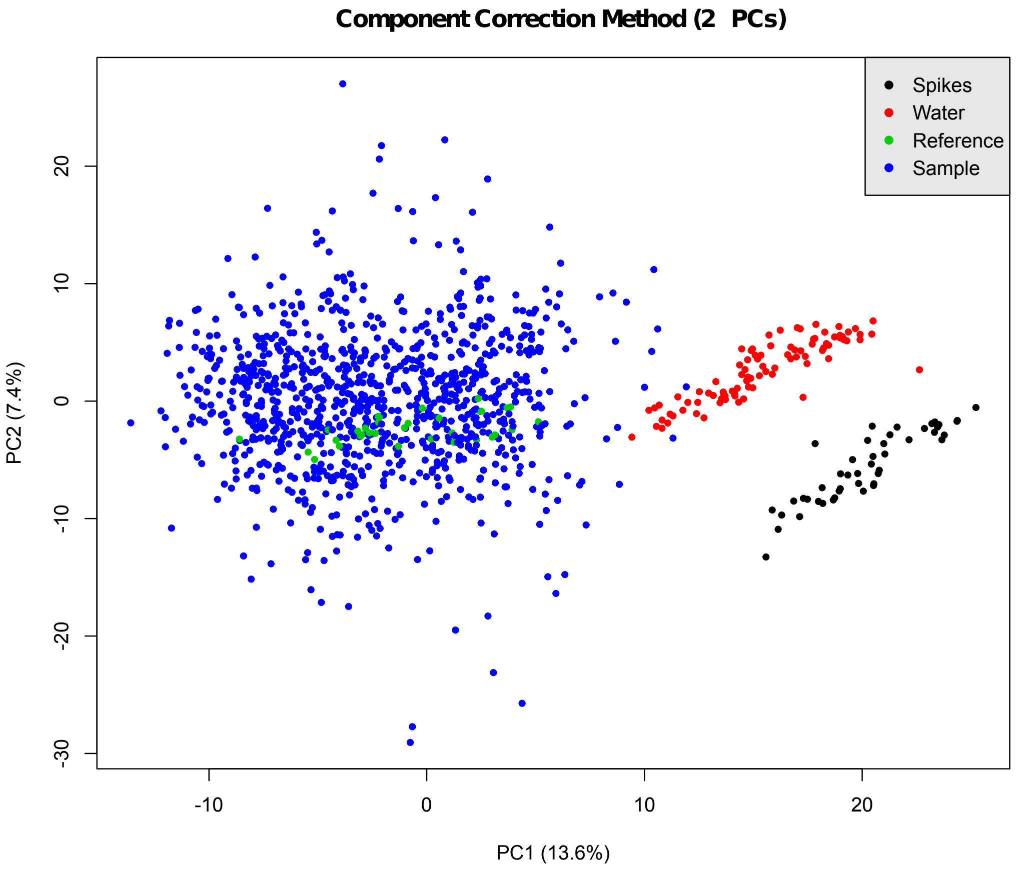


Figure D.1: PCA Scoreplot showing the corrected data when the Component Correction method was applied extracting two PCs.

Table D.1: Results of a linear model fitting and an ANOVA test on each of the methods depicted in Figure 4. The Slope is related to the dataset size (cofactor of the linear model). The Standard Error measures the uncertainty of the Slope in the linear model. The p-value contains the p-values of a statistical test for the Slope.

Method	**Slope**	**Standard Error**	**p-value**
None	-0.00075	0.00018	$3.78 \cdot 10^{-5}$
CC (3PC)	-0.0018	0.0003	$4.43 \cdot 10^{-11}$
CPCA (3CPC)	-0.0003	0.0004	$3.31 \cdot 10^{-1}$
CPCA (3CPC)+Median	-0.0004	0.0003	$2.50 \cdot 10^{-1}$
ComBat	0.0010	0.0002	$1.64 \cdot 10^{-6}$
Median Fold Change	-0.0020	0.0003	$2.20 \cdot 10^{-9}$

This result means that the correction of these methods worse when the dataset is larger. On the other hand, the ComBat method showed the opposite behaviour as its correction improves for larger datasets. Overall, the CPCA_Medians method showed the highest mean value regardless of the dataset size.

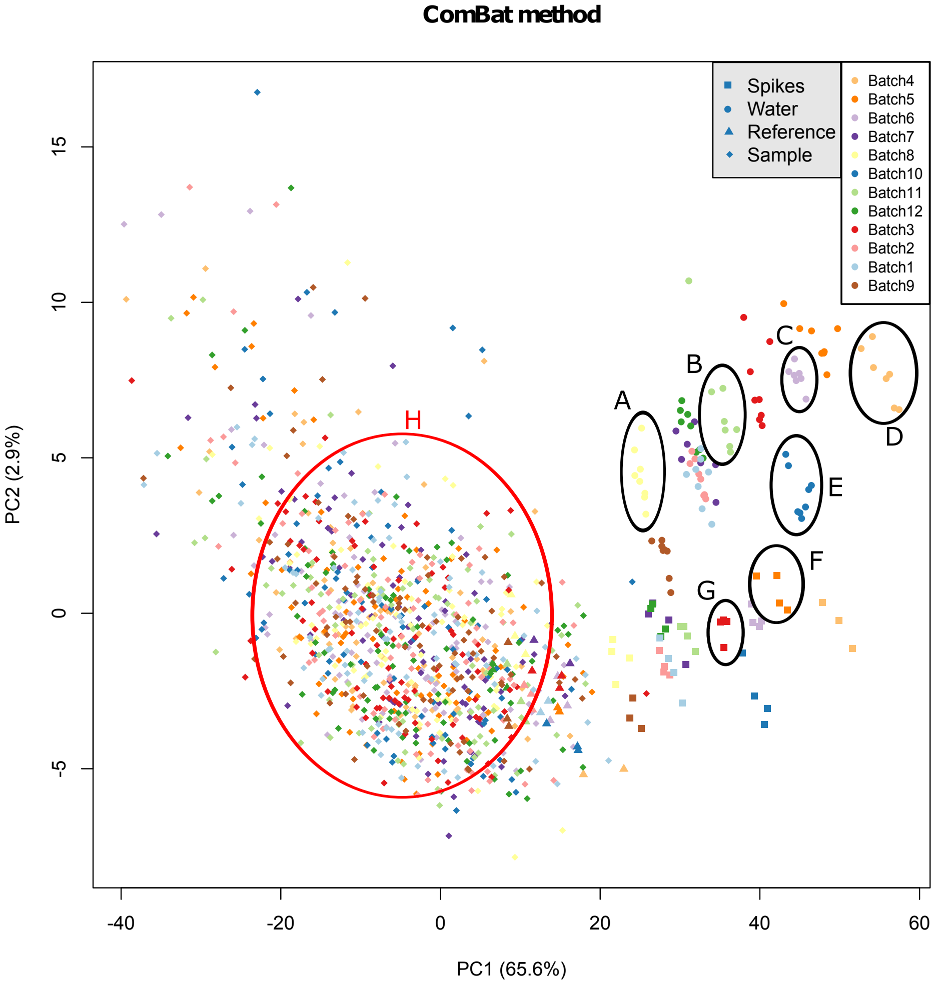


FIGURE D.2: PCA Scoreplot showing the batch effects for the corrected data set using the ComBat method Z^{ComBat}. Colours refer to different batches (the order in the legend correspond to the real injection order of the samples despite the numbers) whereas the geometrical shape of the sample points, refer to the class.

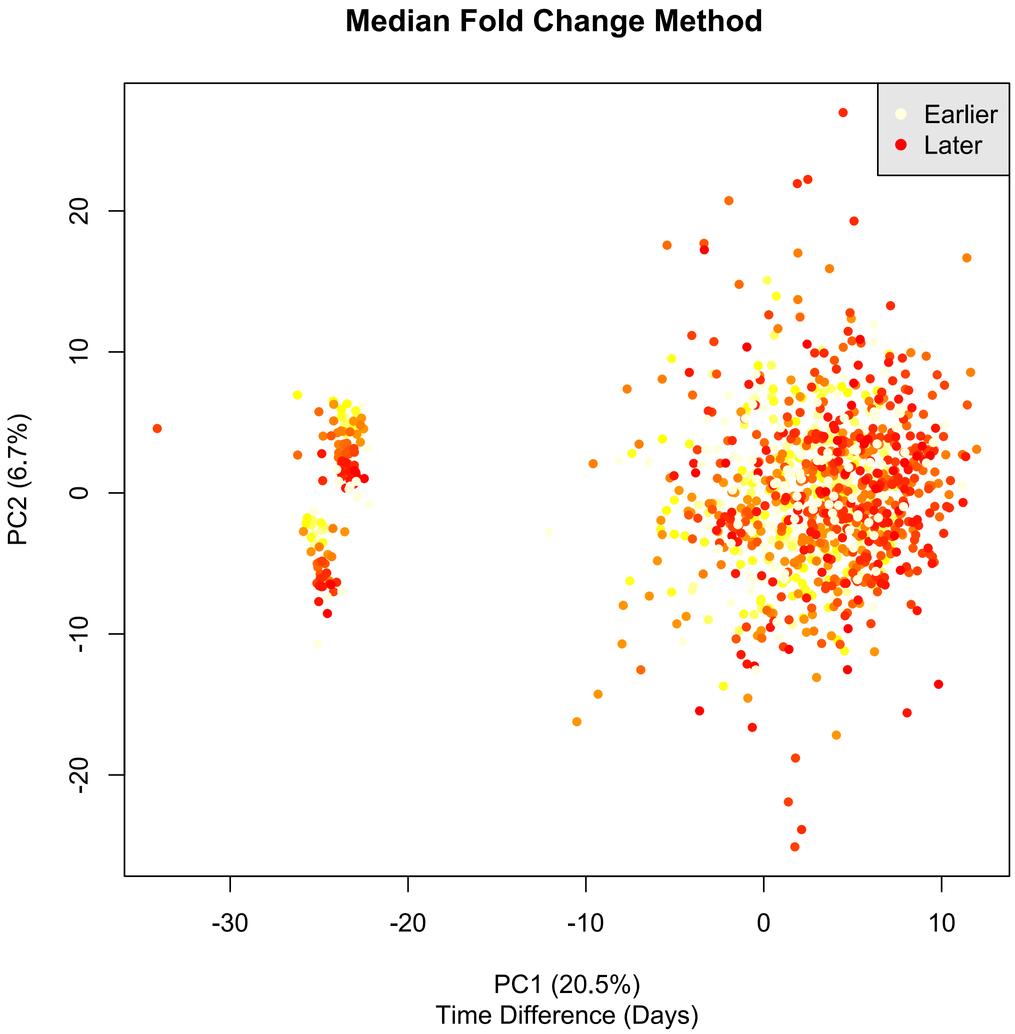


FIGURE D.3: PCA Scoreplot showing the time elapsed effect for the corrected data set using the Median Fold Change Method method $Z^{Medians}$. Colours refer to the time elapsed since the first sample injection. A drift component is observed in the PC2 direction for the clusters corresponding to the Water and Spikes classes.

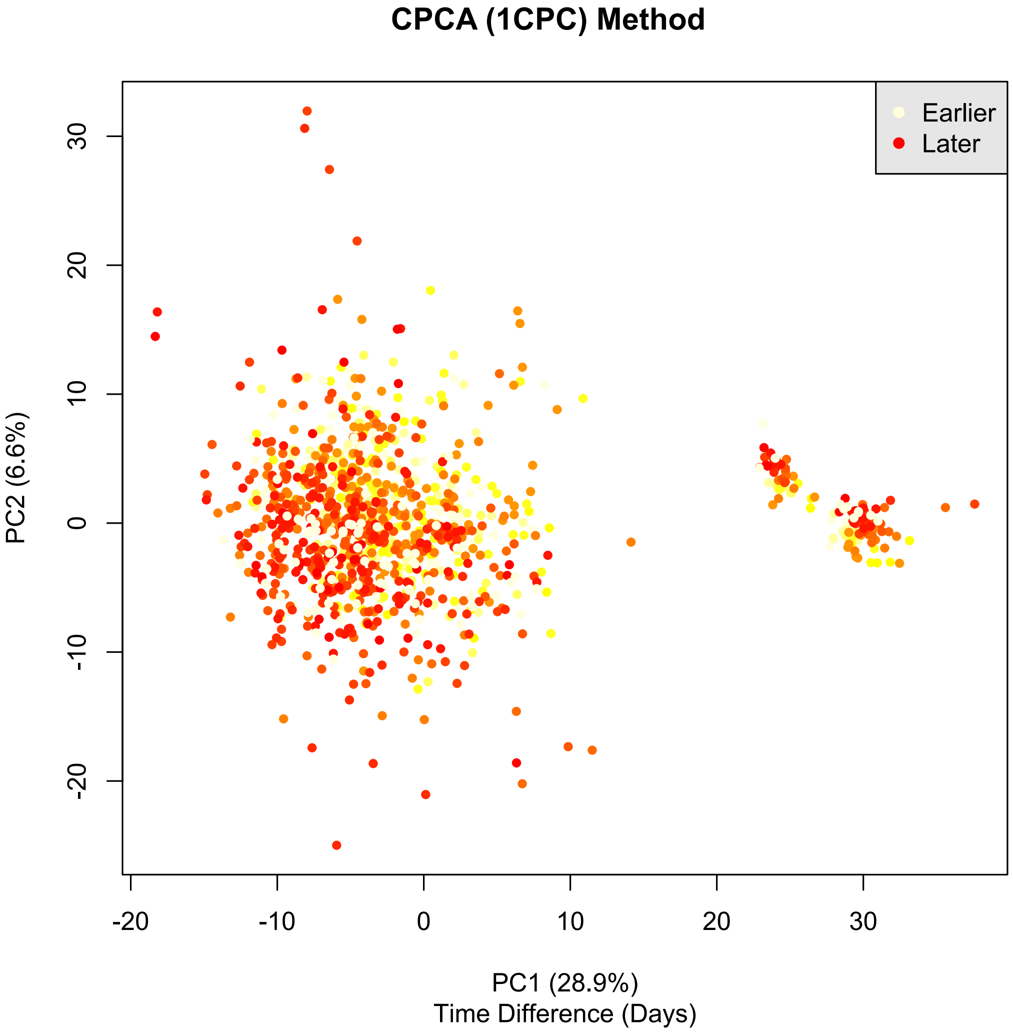


FIGURE D.4: PCA Scoreplot showing the time elapsed effect for the corrected data set using the CPCA method with one CPC removed. Colours refer to the time elapsed since the first sample injection.

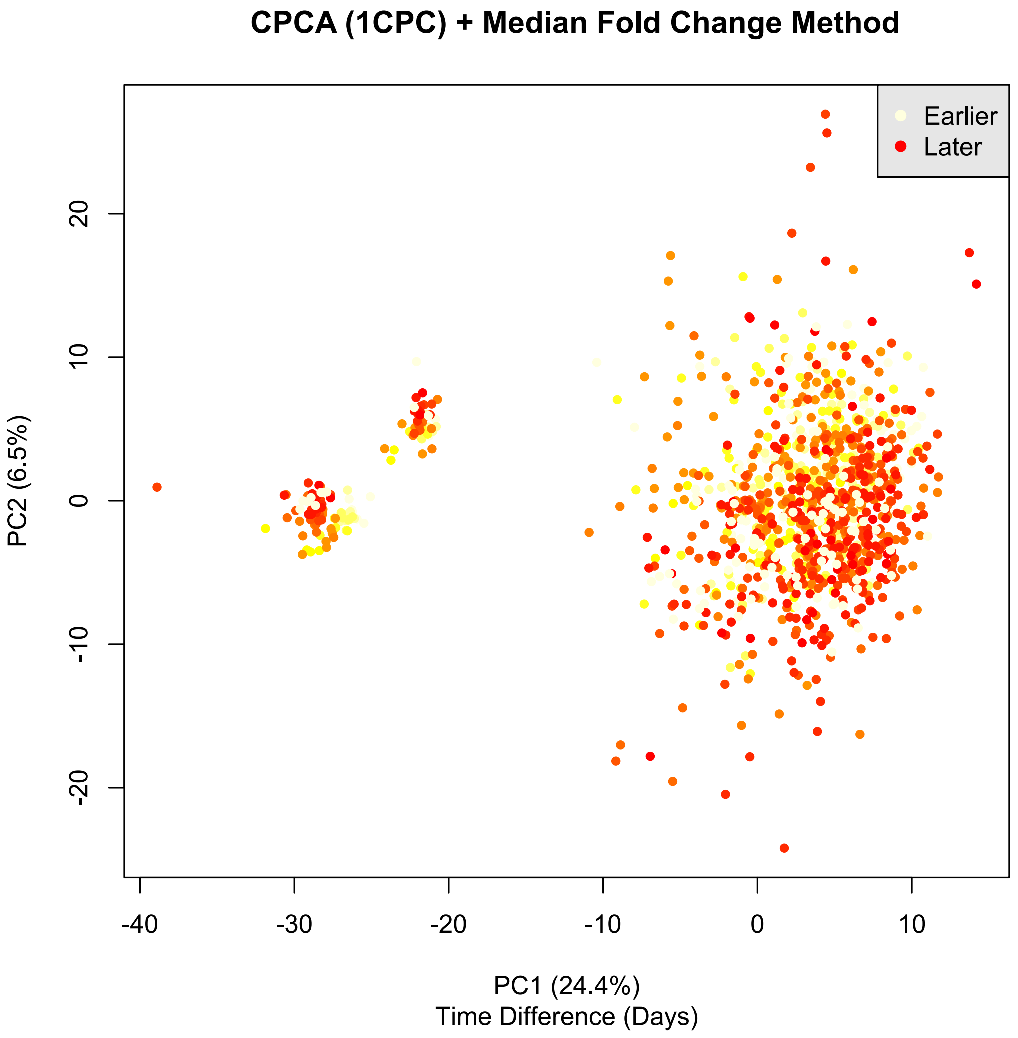


Figure D.5: PCA Scoreplot showing the time elapsed effect for the corrected data set using the CPCA with one CPC removed and a Median Fold Change method. Colours refer to the time elapsed since the first sample injection.

Appendix E

Supporting Information: Correcting time drift effects in Liquid Chromatography using a new non-linear model-based methodology

E.1 Alignment algorithms and Packages

R package XCMS [1] was used to perform both the LOESS and the piecewise Linear regression alignment methods, whereas the PTW method was applied through the R package ptw [2]. The package ptw was applied using the functions baseline.corr to correct the baseline of the signal and the function ptw that performs the alignment using a master sample. The XCMS parameters used in the functions to detect and group the peaks were the same for piecewise Linear and LOESS (snthresh=4, bw=5). As the LOESS algorithm is sensitive to the span parameter[3], several values for the span parameter of the retcor function were checked in the range of [0.7,1.0]. The lowest span tested value (0.7) was suggested as a warning call by the retcor function of the XCMS package. The higher tested span value, was selected to be the value from which making bigger the span, the quality values showed no changes.

E.2 Peak Drifts Computation

To compute the drift of a given peak of a set of samples, first a sample was defined as a master sample. The sample drift was then defined as a pairwise measure between the master sample and another sample that computes how much the sample has moved compared to the master sample. The peak drift is the sample drift considering only the region of the signals where the peak is. In our case the master sample for each dataset was chosen to be the first sample.

To obtain the retention time region of the peaks A to E depicted in the bottom-left part of Figure 6.2, we plotted the fifteen highest intensity points of each sample in a density plot. The density plot corresponding to the dataset 1 is shown in Figure E.8. At this point, a density threshold value was defined and the intersection of the density with this threshold value is taken as the initial peak region. In a second stage, both sides of this initial peak region are enlarged by 30% of the region's total length to ensure that the entire peak falls inside the interval. Making the assumption that the drift between both signals is not very large, we set a maximum drift of 5 temporal units in both directions. Under these conditions, the peak drift is the amount of units that the signal peak has to be moved to have maximum cross-correlation with the same peak of the master sample.

Once the peak retention time range is obtained, the peak drift is computed through the cross-correlation of the two signals in the peak retention time range [4].

E.3 Peak quality measures

The quality measures to evaluate the different alignment algorithms were the peak kurtosis and the peak correlation. Both measures were computed through R functions. For computing the kurtosis it was used the kurtosis function contained in the e1071 package [5]. To compute the peak quality measures, we used the peak retention time interval as obtained following the procedure defined in E.4. These peak quality measures were computed for each of the 5 highest chromatographic regions (see Figure 6.2) and for all the tested alignment methods (the four aligned datasets using the different alignment methods tested and the raw dataset).

E.4 Fitting and applying the H-Models

The model defined in (6.1) was fitted for each sample using the peak drift measures of the peaks A, B and D shown in Figure 6.2. As these peaks show high intensity and they will be found in all the samples, these peak drift measures are the most reliable peak drift measures in the data. These features make them good candidates to be used as fitting points for the model (see a fitting example in Figure E.9).

The parameters of the model (a and b) were computed by minimising the square error of the model (6.1) in these three peak drift measures. The b parameter is limited to have values greater than 0.5 to ensure that the function pole will not affect the range of our signals ($rt \geq 0$) and make the corrected signals diverge. To perform the optimisation procedure it was used the R function optim [6]. We chose L-BFGS-B as the optimising method as it allows to set minimum or maximum bounds to the fitting parameters.

Once the models are fitted, the residuals of these models are extracted and used as a new dataset. The peak drifts measures of this new dataset are obtained again by the same procedure (see section E.3). For each sample of the new dataset, it is computed the overall peak drift error by adding the square of the peak drift measures for the peaks A, B and D. The overall peak drift error is also obtained for the original dataset and both values (before and after correction) are compared to perform a quality measure for each sample. If the overall peak drift error for the corrected sample is bigger than the not-corrected, then the correction is not applied and the original sample values are recovered. On the other hand, if the new overall peak drift error is lower, the correction is applied for that sample. Proceeding this way we ensure that the correction is only applied on those samples in which exists a nonlinear drift component over the retention time.

TABLE E.1: Spike in or endogenous metabolites used in the fitting of the model and its correspondence with the chromatographic peaks shown in Figure 6.2

Retention Time (min)	**Compound**	$[M + H]^+$	**Description**	**Peak correspondence**
1.33	L-Phenylalanine-15N L-Phenylalanine-15N fragment	167.08328 121.07780	Internal Control (Spike In)	Peak A
6.09	L-Indole-3-acetic-2,2-d2 acid L-Indole-3-acetic-2,2-d2 acid fragment L-Indole-3-acetic-2,2-d2 acid isotope L-Indole-3-acetic-2,2-d2 acid Na-adduct L-Indole-3-acetic-2,2-d2 acid K-adduct	178.08318 132.07700 179.08318 200.06480 214.0901	Internal Control (Spike In)	Peak D
3.71	Hippuric acid fragment Hippuric acid	105.03 180.06551	Urine endogenous metabolite	Peak B

Bibliography

[1] C A Smith, E J Want, G O'Maille, R Abagyan, and G Siuzdak. XCMS: processing mass spectrometry data for metabolite profiling using nonlinear peak alignment, matching, and identification. *Analytical Chemistry*, 78(3):779–787, 2006. ISSN 00032700. doi: 10.1021/ac051437y.

[2] Tom G. Bloemberg, Jan Gerretzen, Hans J. P. Wouters, Jolein Gloerich, Maurice van Dael, Hans J. C. T. Wessels, Lambert P. van den Heuvel, Paul H. C. Eilers, Lutgarde M. C. Buydens, and Ron Wehrens. Improved parametric time warping for proteomics. *Chemometrics and Intelligent Laboratory Systems*, 104(1):65–74, 2010.

[3] Katharina Podwojski, Arno Fritsch, Daniel C. Chamrad, Wolfgang Paul, Barbara Sitek, Kai Stühler, Petra Mutzel, Christian Stephan, Helmut E. Meyer, Wolfgang Urfer, Katja Ickstadt, and Jörg Rahnenführer. Retention time alignment algorithms for lc/ms data must consider non-linear shifts. *Bioinformatics*, 25(6):758–764, 2009. doi: 10.1093/bioinformatics/btp052.

[4] John G. Proakis and Dimitris K. Manolakis. *Digital Signal Processing (4th Edition)*. Prentice Hall, 4 edition, April 2006. ISBN 0131873741.

[5] David Meyer, Evgenia Dimitriadou, Kurt Hornik, Andreas Weingessel, and Friedrich Leisch. *e1071: Misc Functions of the Department of Statistics (e1071), TU Wien*, 2012. R package version 1.6-1.

[6] R Core Team. *R: A Language and Environment for Statistical Computing*. R Foundation for Statistical Computing, Vienna, Austria, 2012. ISBN 3-900051-07-0.

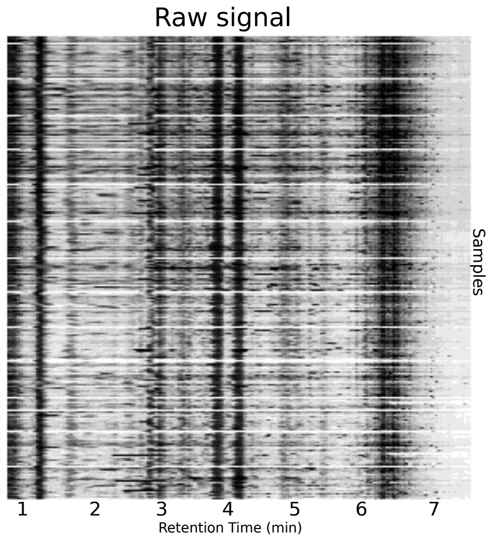


FIGURE E.1: Raw chromatograms for the second dataset. Each row refers to a sample whereas the columns are the retention time. Darker grey means higher signal intensity.

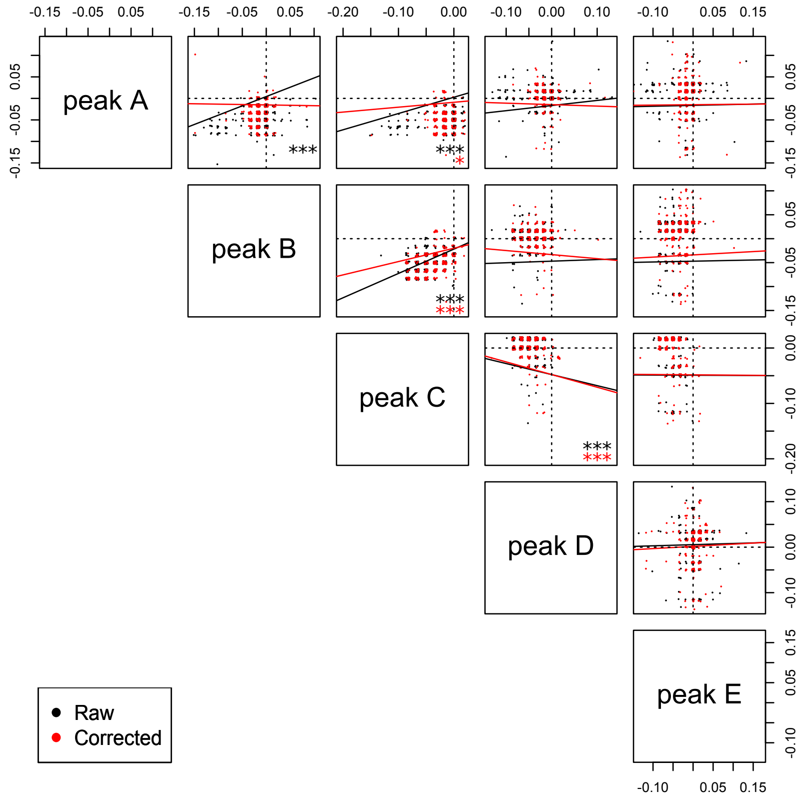


FIGURE E.2: Peak Drifts before and after applying the H-Cor method to Dataset 1 labelled Raw and Corrected respectively. The peak names refer to the same peaks depicted in Figure 6.2. The asterisks refer to statistically significant model following an ANOVA test. The numerical p-values can be found at table E.2. The linear models were fitted using the minimum squared approach.

Table E.2: P-values on the slopes for the peak drifts regressions depicted in Figure E.2

P-values of the ANOVA tests on the slopes linear models					
	Peak A	**Peak B**	**Peak C**	**Peak D**	**Peak E**
Peak A	**Raw** **Corrected**	$1.13 \cdot 10^{-7}$ *** 0.512	$1.1 \cdot 10^{-4}$ * ** 0.016 *	0.192 0.138	0.754 0.607
Peak B		**Raw** **Corrected**	$< 2.2 \cdot 10^{-16}$ *** $4.38 \cdot 10^{-07}$ ***	0.400 0.508	0.703 0.314
Peak C			**Raw** **Corrected**	$1.94 \cdot 10^{-4}$ *** $7.68 \cdot 10^{-4}$ ***	0.788 0.439
Peak D				**Raw** **Corrected**	0.469 0.117

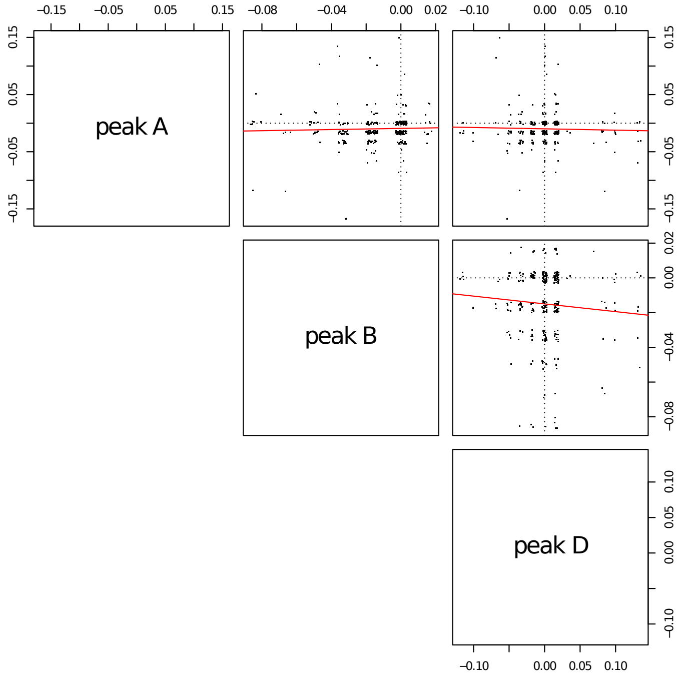


Figure E.3: Peak Drifts for Dataset 2. The red line corresponds to a linear fit using minimum squares approach. The peak names refer to the same peaks depicted in Figure 6.2

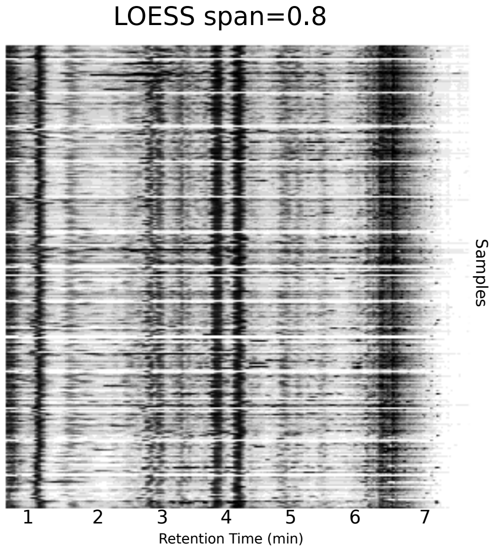


FIGURE E.4: Aligned chromatograms for the first dataset using the LOESS method with $span = 0.8$. Each row refers to a sample whereas the columns are the retention time in minutes. Darker grey means higher signal intensity. The chromatogram at the top of the image corresponds to the average chromatogram of all the samples of the dataset.

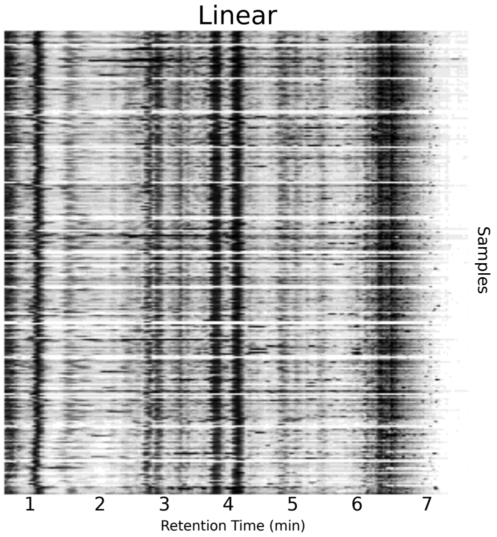


FIGURE E.5: Aligned chromatograms for the first dataset using the piecewise linear method. Each row refers to a sample whereas the columns are the retention time in minutes. Darker grey means higher signal intensity. The chromatogram at the top of the image corresponds to the average chromatogram of all the samples of the dataset.

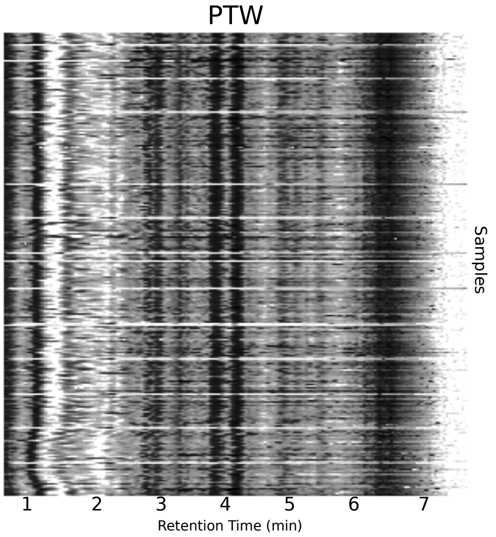


Figure E.6: Aligned chromatograms for the first dataset using the PTW method. Each row refers to a sample whereas the columns are the retention time in minutes. Darker grey means higher signal intensity. The chromatogram at the top of the image corresponds to the average chromatogram of all the samples of the dataset.

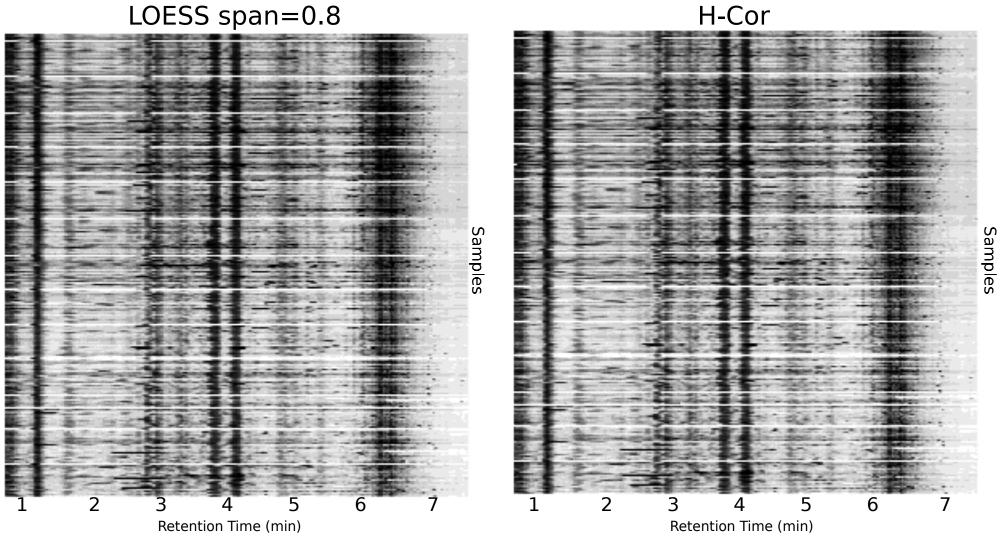


Figure E.7: Aligned chromatograms for the second dataset using the H-cor and LOESS methods (span=0.8). Each row refers to a sample whereas the columns are the retention time in minutes. Darker grey means higher signal intensity.

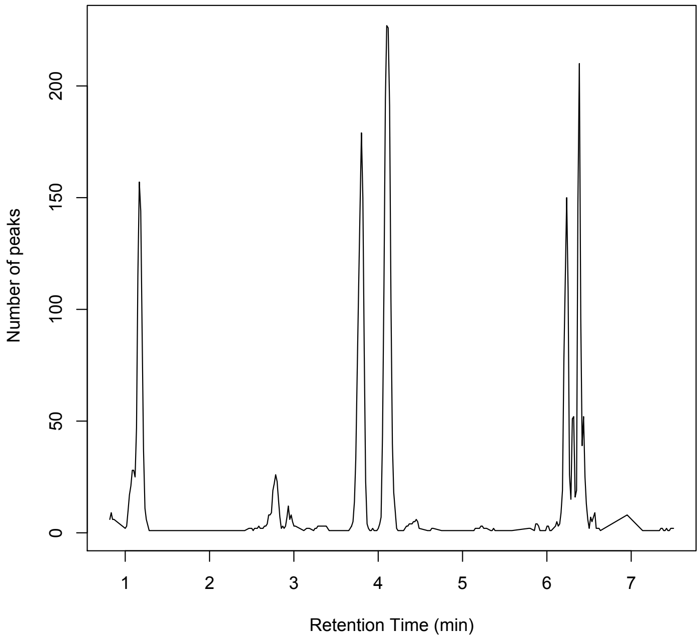


FIGURE E.8: Peak density for dataset 1. The 15 most intense points of each sample were picked and plotted according to their retention time. The peaks show regions of high intensity peaks (i.e. density of high intensity peaks) rather than chromatographic peaks.

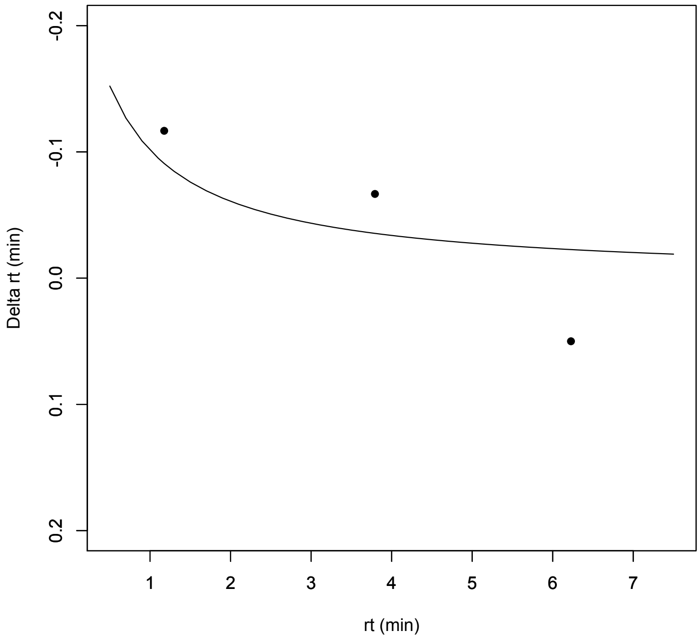


FIGURE E.9: The line depicts the H-Cor model fitted for a sample of the first dataset. The points show the peak drifts for peaks A, B and D which were used to adjust the model expressed as shown in (6.1).