PROBABILISTIC METHODS FOR LIQUID CHROMATOGRAPHY MASS SPECTROMETRY DATA PRE-PROCESSING

JOE WANDY

SUBMITTED IN FULFILMENT OF THE REQUIREMENTS FOR THE DEGREE OF
Doctor of Philosophy

SCHOOL OF COMPUTING SCIENCE

COLLEGE OF SCIENCE AND ENGINEERING
UNIVERSITY OF GLASGOW

MARCH 2016

Abstract

This is a dissertation outline using the style guidelines defined by the University of Glasgow.

Acknowledgements

ACK

Dedication. (Is what you need.)

Table of Contents

List of Tables

List of Figures

Todo list

Chapter 1

Introduction

Lorem ipsum dolor sit amet, consectetur adipisicing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris nisi ut aliquip ex ea commodo consequat. Duis aute irure dolor in reprehenderit in voluptate velit esse cillum dolore eu fugiat nulla pariatur. Excepteur sint occaecat cupidatat non proident, sunt in culpa qui officia deserunt mollit anim id est laborum.

Lorem ipsum dolor sit amet, consectetur adipisicing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris nisi ut aliquip ex ea commodo consequat. Duis aute irure dolor in reprehenderit in voluptate velit esse cillum dolore eu fugiat nulla pariatur. Excepteur sint occaecat cupidatat non proident, sunt in culpa qui officia deserunt mollit anim id est laborum.

Lorem ipsum dolor sit amet, consectetur adipisicing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris nisi ut aliquip ex ea commodo consequat. Duis aute irure dolor in reprehenderit in voluptate velit esse cillum dolore eu fugiat nulla pariatur. Excepteur sint occaecat cupidatat non proident, sunt in culpa qui officia deserunt mollit anim id est laborum.

Lorem ipsum dolor sit amet, consectetur adipisicing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris nisi ut aliquip ex ea commodo consequat. Duis aute irure dolor in reprehenderit in voluptate velit esse cillum dolore eu fugiat nulla pariatur. Excepteur sint occaecat cupidatat non proident, sunt in culpa qui officia deserunt mollit anim id est laborum.

Lorem ipsum dolor sit amet, consectetur adipisicing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris nisi ut aliquip ex ea commodo consequat. Duis aute irure dolor in reprehenderit in voluptate velit esse cillum dolore eu fugiat nulla pariatur. Excepteur sint occaecat cupidatat non proident, sunt in culpa qui officia deserunt mollit anim id est laborum.

Lorem ipsum dolor sit amet, consectetur adipisicing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris nisi ut aliquip ex ea commodo consequat. Duis aute irure dolor in reprehenderit in voluptate velit esse cillum dolore eu fugiat nulla pariatur. Excepteur sint occaecat cupidatat non proident, sunt in culpa qui officia deserunt mollit anim id est laborum.

Lorem ipsum dolor sit amet, consectetur adipisicing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris nisi ut aliquip ex ea commodo consequat. Duis aute irure dolor in reprehenderit in voluptate velit esse cillum dolore eu fugiat nulla pariatur. Excepteur sint occaecat cupidatat non proident, sunt in culpa qui officia deserunt mollit anim id est laborum.

Lorem ipsum dolor sit amet, consectetur adipisicing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris nisi ut aliquip ex ea commodo consequat. Duis aute irure dolor in reprehenderit in voluptate velit esse cillum dolore eu fugiat nulla pariatur. Excepteur sint occaecat cupidatat non proident, sunt in culpa qui officia deserunt mollit anim id est laborum.

Lorem ipsum dolor sit amet, consectetur adipisicing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris nisi ut aliquip ex ea commodo consequat. Duis aute irure dolor in reprehenderit in voluptate velit esse cillum dolore eu fugiat nulla pariatur. Excepteur sint occaecat cupidatat non proident, sunt in culpa qui officia deserunt mollit anim id est laborum.

Lorem ipsum dolor sit amet, consectetur adipisicing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris nisi ut aliquip ex ea commodo consequat. Duis aute irure dolor in reprehenderit in voluptate velit esse cillum dolore eu fugiat nulla pariatur. Excepteur sint occaecat cupidatat non proident, sunt in culpa qui officia deserunt mollit anim id est laborum.

1.1 Thesis Statement

1.2 Overview of Thesis and Research Contributions

Chapter 2

Computational Biology Background

This chapter provides the background knowledge necessary to understand the basic principles of mass-spectrometry-based analysis as applied to large-scale untargeted biological studies. A particular emphasis is given to the application of mass spectrometry techniques to the field of metabolomics. For a further reading on mass spectrometry, the reader is directed to a more comprehensive textbooks such as [3] and [4]. Reviews on the necessary data preprocessing steps of mass spectrometry data can be found for e.g. in proteomics [5, 6, 7] and metabolomic [8, 9, 10].

2.1 Computational Biology

Since the discovery of the deoxyribonucleic acid (DNA) as the basic storage of genetic information, the same fundamental principle is found to govern the transmission of hereditary information common to all life on Earth. Following the central dogma of molecular biology:

DNA is transcribed into RNA, which is translated into proteins.

In the central dogma, genetic materials are coded in the DNA, a strand of which consists of a series of nucleotides. The backbone of a nucleotide comprises sugar and phosphate groups attached to one of the four nitrogenous bases of **A**denine, **T**hymine, **G**uanine, and **C**ytosine, forming the four well-known alphabets of the DNA. The start of the transcription process begins with the unwounding of the double strand of the DNA into single strands. The single DNA strand is transcribed by a protein complex (RNA polymerase) into messenger ribonucleic acid (RNA), which can be be thought as nearly identical to DNA, with the crucial difference that the base **U**racil is used in place of thymine. The substitution of thymine to uracil allows RNA to perform its important function as the messenger RNA, which as the name suggests is the intermediate mechanism of messaging for the information contained in

the chemically inert DNA. Messenger RNA is read by the ribosome, a part of the translational apparatus of the cell, and translated into amino acids, which are the building blocks of proteins.

Since its initial proposal, this simple central dogma model has been challenged and expanded to acknowledge other factors that can influence the transcription and translation processes. Nevertheless, the central dogma serves to illustrate the flow of genetic information in a biological system. Different sub-fields of computational biology predominantly study the different entities and processes involved in the central dogma. Genomics is concerned with the large-scale study of the genome (the entire DNA in the organism) and how the genes encoded in the genome interact with each other. Sequencing technologies, in particular next-generation sequencing (NGS) machines such as Illumina and Ion Torrent, have been instrumental in revolutionising genomics by making possible the high-throughput and rapid sequencing of the entire DNA sequence from a sample [11]. Transcriptomics focuses on understanding the transcriptome (the complete set of messenger RNA) and their measurement. Transcriptome relies on DNA micro-array technologies and more recently, have been increasingly performed by NGS sequencing as well. Proteins and their large-scale identifications and quantifications are studied in proteomics, while the complete set metabolites present in the sample and their expressions are the focus of metabolomics. Each successive layer of the -omics hierarchy, which comes closer to the actual physical expression of observable traits (phenotypes), introduces more complexity due to the increased number of ways of putting the building blocks in that layer together (Figure 5.1).

Draw this

Figure 2.1: The building blocks of the genome are the DNA nucleotides. In the transcriptome, the building blocks are the nitrogenous bases that comprise the RNA. In proteomics, the 20 amino acids residues make up the polypeptide comprising a protein molecule. In contrast, the building blocks of metabolites are the atoms (usually CHNOPS: carbon, hydrogen, nitrogen, oxygen, phospor and sulphur) that comprise a large range of compounds, such as lipids, amino acids, vitamins, etc., with varying physical and chemical properties

2.2 Mass spectrometry-based omics

The two -omics closest to the phenotype in Figure 5.1 rely on mass spectrometry, usually coupled to a separation intrument such as chromatography, to perform measurements and quantification of the biological entities of interest. Proteomics and metabolomics are briefly expanded in this section, while details on the set-up of the instruments can be found in Section 2.3.

Proteins, considered the building blocks of life, serve critical roles in an organism by performing cellular maintenance, catalysing chemical reactions, carrying molecules across cell membranes and many other essential functions. The primary building blocks of a protein are amino acids, which results from the translation of messenger RNA. An amino acid consists of the amine group (-NH_2), a carboxylic group (-COOH) and a side chain. Through the loss of water molecule, amino acids can be chained to each other through the peptide bonds, collectively forming a peptide. Each amino acid can be described by a unique letter drawn from a set of 20 chemical alphabets, and consequently a peptide can be succinctly described as a string of letters corresponding to its constituent amino acids. While the sequence of amino acids comprising a protein is largely coded by genes in the genome, this process is far from deterministic. In a process called post-translational modification [12], proteins can be chemically modified after synthesis in a way that completely alters its structure and folding stability, e.g. through phosphorylation, methylation or glycosylation. This results in a large variety of protein diversity present in the biological system, and it is the large-scale characterisation of identities and quantities of proteins that is of particular interest to **proteomics**.

Apart from proteins, numerous other chemical reactions essential for sustaining life also happen inside a cell. In catabolic reactions, large organic molecules within a cell are broken into energy and smaller molecules. These serve as the input to anabolic reactions, producing the basic building blocks of a cell such as proteins and nucleic acids. Both anabolic and catabolic reactions are usually catalysed by enzymes, and together these two reactions comprise the metabolism of an organism. Metabolites are the molecules involved during or produced as the by-products of metabolism. Through the help of various enzymes, metabolites are transformed from one form to another in a series of chemical reactions as part of the metabolic pathways. Some examples of common metabolites are the various amino acids, fatty acids, and vitamins (e.g. B3 and B12) and minerals (e.g. phosphorus, iron and zinc). The overall set of metabolites that can be found within an organism is collectively called the metabolome. **Metabolomics** studies the metabolome on a large scale, usually for the purpose of identifying and quantifying their differences in the particular organisms or tissues under various experimental or physiological conditions. As metabolomics as a study is considered to be the closest to the phenotype, changes to the metabolome often result in physically observed properties, and indeed changes in the metabolite composition of an organism may be

caused by responses to environmental and genetic factors [13]. Studying the metabolome provides us with an instantaneous 'snapshot' of the chemical activities that are occurring in the cell at that moment.

2.3 Mass Spectrometry

Atoms are small building blocks of matter. An atom has a nucleus at the centre, which consists of positively charged protons and neutrons with no charge. Electrons, having negative charge, are bound to the neucleus through electromagnetic force. The overall charge of the atom is therefore determined by the number of electrons and protons that it has. The atom is called a positive ion when there are more protons than electron, otherwise it is a negative ion. Two or more atoms hold together via chemical bonds comprise a compound. The molecular mass of a compound is the sum of the molecular mass of its elements, measured in Dalton (Da), where one Da is $\frac{1}{12}$ of the molecular mass of the carbon element (^{12}C). Elements in nature occur as isotopes. Isotopes are naturally occurring elements that have the same number of protons (same atomic number) but different number of neutrons (different molecular masses). Each elements has many isotope species, for instance carbon has two isotopes: ^{12}C with molecular mass 12.000000 at 98.890% abundance in nature, and ^{13}C with molecular mass 13.003355 and 1.110% abundance. The term 'mono-isotopic' refers to the most abundant isotope species of an element. The exact mass of a compound can therefore be calculated from the formula sum of the masses of its constituent mono-isotopes. The nominal mass of a compound is similarly calculated by summing the integer masses of the constituent mono-isotopes (e.g. the nominal mass of $H_2O = 1 + 1 + 16 = 18$).

Mass spectrometer (MS) coupled to liquid chromatography, forming the set-up of liquid chromatography mass spectrometry (LC-MS), is the preferred measurement platform for determining the elemental composition and the abundance of the analytes (proteins or metabolites) in proteomics or metabolomics studies. MS instruments can be ranked by the ascending order of their resolving powers of their mass analyser: **(1)** time-of-flight MS, **(2)** quadropole MS, and lastly **(3)** Fourier transform ion-cyclotron MS. A higher resolving power corresponds to a better ability of the instrument to detect small differences in mass-to-charge (m/z) ratios. Having a higher resolving power is generally very useful when trying to identify which metabolites are present in the sample. Modern high-precision MS instruments have very accurate resolving power, with accuracy up to several parts-per-million. The difference between the observed mass-to-charge value to the exact-mass-to-charge value of a compound is the mass accuracy of a mass spectrometry instrument, measured in parts-per-million, i.e. mass accuracy $= 1e6 * \frac{\text{(observed m/z}-\text{exact m/z)}}{\text{exact m/z}}$.

2.3.1 Metabolomics

In recent years, the combination of liquid chromatography coupled to mass spectrometry (LC-MS) has emerged as one of the most widely used techniques in untargeted metabolomic studies. Metabolites in the extracted sample cannot be introduced at once as direct injection into MS due to ion supression effect [14], where compounds 'compete' for charges during the ionisation process inside the MS. Due to this ion supression effect, metabolites present in low abundance might not be ionised and therefore not detected in the resulting mass spectra. As a result, it is necessary for metabolites to be separated before being introduced gradually into the inlet of the ionisation source. Separation techniques such as liquid chromatography LC coupled to MS is commonly used for this purpose. In liquid chromatography, the mobile solvent containing the analytes (metabolites) is introduced and pumped into the stationary phase of the chromatographic column. Metabolites elutes at different time through their interactions with the capillary, based on the hydrophobicity, charge and other chemical properties of the metabolites. The time it takes for these metabolites to elute through the stationary phase of the LC column is called the retention time (RT). LC-MS tends to be easier to automate and suitable for high throughput experiments. Sample preparations for an LC-MS set-up also tends to be simpler compared to the alternative of separation via gas chromatography, while compounds across a wide range of polarity can be separated [3].

Metabolites that elute from liquid chromatography are then vaporised and ionised inside the mass spectrometer. This is usually accomplished through soft-ionisation methods such as atmospheric pressure ionisation or electrospray ionisation (ESI). The distinction between soft- and hard- ionisation methods come from how 'soft' methods do not break the chemical bonds of the compound during the ionisation process, which stands in contrast to hard-ionisation methods, such as the electron impact ionisation, that breaks the chemical bonds in the neutral molecules of compounds. ESI can be directly coupled to LC, so often, it is the preferred method of ionisation. In ESI, the sample analyte is dissolved into a solvent and sprayed through an electrospray. It is the resulting charged aerosol that enters the vacuum of the mass spectrometer, generating charged molecular ions and their corresponding fragment ions. The generated ions are separated by the mass analyser inside the MS instrument according to their m/z (mass-to-charge) ratios and the detected signal abundance for a particular m/z value. The result of this process is a mass spectrum: a two dimensional representation of m/z values to signal intensities. The final raw data produced by an LC-MS setup is called the ion chromatograms: a collection of mass spectra over the range of elution time. The entire raw data can therefore be characterised by a set of vector of m/z, intensity and retention time, and for every slice on the ion chromatogram sharing the same RT value (a scan), a mass spectrum is produced from metabolites that elute at that same retention time. A mass spectrum is the observed m/z and intensity (abundance) values of the peaks that result

from fragmentations of the metabolites during the scan. When the MS instrument is run on the full-scan mode, the entire m/z range is selected for fragmentation.

Redraw this to be simpler.

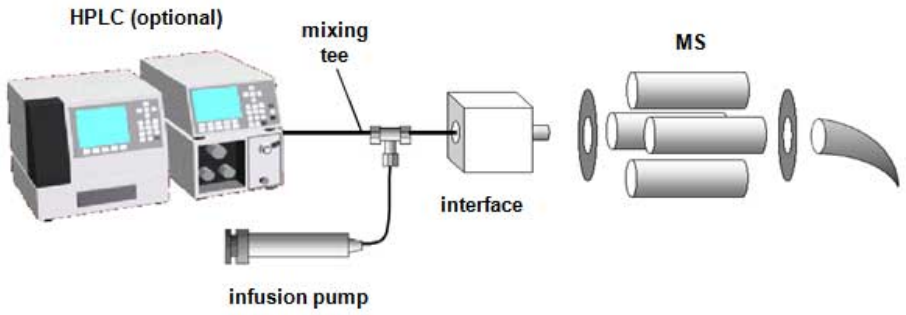


Figure 2.2: A typical LC-MS set-up. High performance liquid chromatography instruments are used to separate metabolites (by their chemical properties) in the sample before they are gradually introduced into the mass spectrometer.

Redraw this to illustrate the point better.

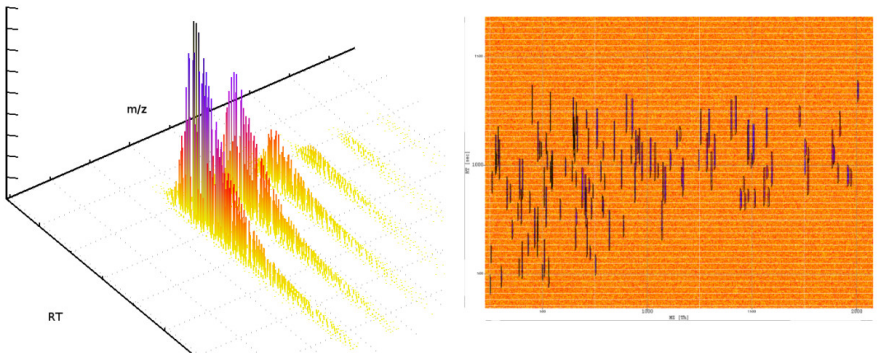


Figure 2.3: The resulting raw data (ion chromatograms) produced from an LC-MS experiment. We can view the data as a 3D profile (left) and as a 2D profile seen from the top (right). A slice of the data on the m/z axis is the mass spectrum. Each mass spectrum is produced by a scan of the mass spectrometer. A collection of mass spectra is produced over the whole range of retention time. A point in the raw data is thus characterised by its intensity value on the m/z and retention time axes.

2.3.2 Proteomics and Glyomics

For mass spectrometry analysis of proteins, the samples to be analysed come either in the form of tissues or as body fluids, such as urine, plasma and serum. Different types of samples will demand the appropriate sample handling protocol in the sample preparation stage. Next, cells extracted from the sample are broken down, allowing proteins to be isolated from other constituent parts of the cell, for instance the DNA, lipids and other metabolites

that are present. The purified proteins are then separated. Traditional 2-D gel electrophoresis method allows proteins to be separated according to their size (molecular mass) in one axis and according to their isoelectric points (the pH where the molecule carries no electrical charges) on another. Because 2D-GE approach is tedious and time-consuming, liquid chromatograph mass spectrometry has gotten more popular as the preferred separation technology as it enables the large-scale high-throughput separation of thousands of proteins in a single chromatographic run. Enzymes that can cut the peptide bonds, such as trypsin, are then used to digest proteins into shorter peptide fragments. Using certain enzymes, the cleavage of the peptide bonds happen at specific and predictable spots, allowing well-defined and easily identifiable peptide fragments to emerge. For instance by using trypsin as the digestion enzyme, the cleavage of the protein happens after each arginine or lysine amino acid is encountered, unless a proline amino acid comes next. An initial separation process (prefactionation) can also be performed on the digested peptides using liquid chromatography, resulting in different fractions, which can then be ran separately through the hypenated set-up of LC-MS (Figure 2.2) for mass fragmentation analysis in a manner similar to metabolomics analysis (described in the following paragraphs in Section 2.3.1). This yields the peptide mass fingerprint, which although challenging, can generally be used to match the resulting peptide fingerprints against a database of reference fingerprints for identification of the peptides and correspondingly the entire protein.

2.3.3 Fragmentation

Fragmentation through tandem MS or MS^n instruments can be used to provide further fragmentation information for metabolite identification. As suggested by its name, tandem MS requires two MS analysers operating in tandem. Ions resulting from the initial fragmentation of metabolites in the first MS analyser are selected for further fragmentation in the second MS analyser. The ions selected for the first MS analyser stage are called the precursor ions. In data-dependent acquisition (DDA), precursor ions within some small m/z windows are selected based on some predetermined rules (such as fragmenting the top few most intense precursor peaks in each scan). As a result, typically a small percentage, e.g. less than a fifth of all precursor peaks in the full-scan mode data are selected for MS-MS fragmentation [**?**]. Peaks that are generated from the fragmentation of the precursor ions in the second MS stage are called product ions. Fragmentation spectra of product ions are often used as the unique 'fingerprint' identifiers of the structural composition of the precursor ions. This is described further in Section **??**. An alternative to DDA is the data-independent acquisition (DIA), where no selection of precursor ions needs to be specified as all peaks within a defined m/z range are fragmented. DIA results in a more complex fragmentation spectra due to multiple peptides/metabolites being fragmented together in the same m/z window, and

require sophisticated analysis strategy to deconvolve the signals from the noise.

2.4 Metabolomics Pipeline: From Raw Data to Biological Hypothesis

The raw LC-MS data is noisy, so pre-processing has to take place before analysis can be performed and biological conclusion drawn. The raw LC-MS data has to go through successive pre-processing and transformations along the data pre-processing pipeline before it can be analysed. The main steps of LC-MS data preprocessing generally involve peak detection and the filtering of noise, the matching of identical peaks across samples (alignment), identifications of peaks and lastly, data normalization and visualization. 2.4 shows these key preprocessing steps in the typical LC-MS data processing pipeline, which is elaborated further next.

Redraw this to make it look nicer..?

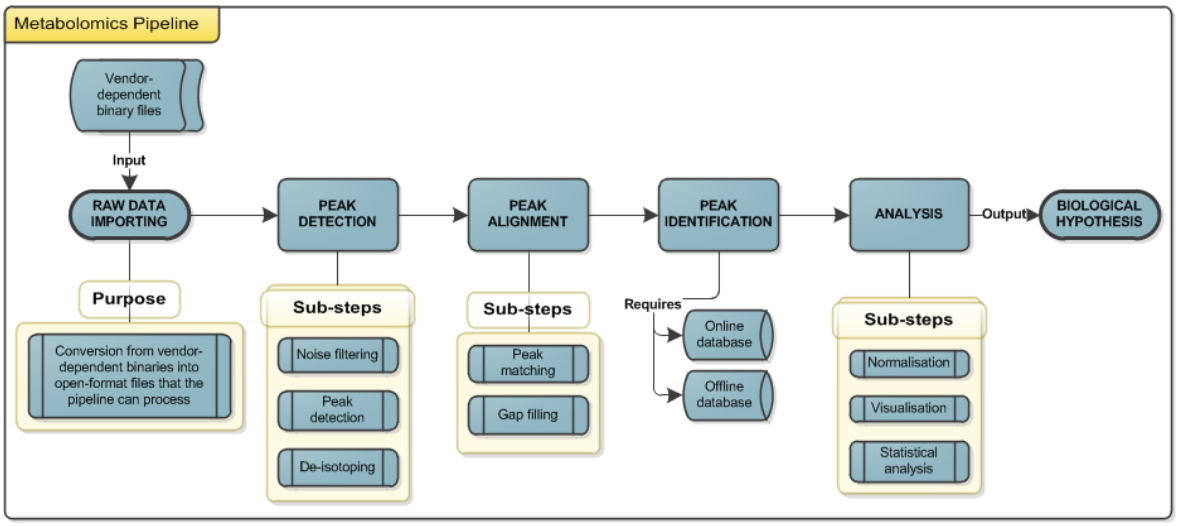


Figure 2.4: Preprocessing pipeline of LC-MS metabolomics data.

2.4.1 Peak Detection

The raw LC-MS data is imported into the pipeline. Beginning as a vendor-proprietary format, the raw data is converted into open XML-based format for storing mass spectrometry data, such as the mzXML or mzML format [15]. Noise filtering is performed as a preliminary filtering to remove noises and artifact signals due to the various chemical noises that also occur during the ionisation process. Peak detection is then performed to identify areas and intensities of peaks. A survey of the different approaches towards peak detection can be

found in [16], but what is important to note is at this stage, additional peaks can potentially be introduced due to peaks falsely detected from chemical noises, e.g. as a result of the contaminants present in the sample, while on the other hand, peaks that should be detected can instead be missing as a consequence of setting incorrect parameters for the detection step, e.g. by setting threshold values that are too low.

Additionally, not all observed peaks would correspond to true precursor ions of the metabolites, since peaks could also be generated by other entities sharing the same identifying mass value, due to the presence of isotopes, contaminants, adducts and other signal artifacts in the sample ([17]). In particular, due to the presence of naturally occurring isotopes (e.g. ^{13}C) and the formation of adducts (the addition of a molecule ion to another), one precursor ion corresponding to a single metabolite alone can produce many observed peaks in the mass spectrum, forming a distribution of isotopic peaks at different m/z values but having similar chromatographic peak shapes in their elution time profiles. As one of the main challenges of peak detection comes from the presence of these isotope peaks, the de-isotoping step is often performed as an integral part of the peak detection step. The presence of multiple peaks that can be traced back to a single metabolite

Write more about related peaks stuff.

2.4.2 Peak Alignment

The next step in the LC-MS data processing pipeline is the peak alignment step, where peaks from different LC-MS runs have to be matched. Experiments in biology usually involve the comparison of multiple samples. Samples can be produced as either biological or technical replicates. Biological replicates are obtained from the same organism studied under varying conditions. The organism studied are usually exposed to different factors (e.g. treatment or no treatment) controlled throughout the course of the experiment. Biological replicates are necessary to determine entities that are differentially expressed across samples. In contrast, technical replicates are obtained from the same samples analysed multiple times. Technical replicates are necessary to account for variability and measurement errors throughout the experiment. Since experiments in biology usually involve a comparison of multiple samples, it is necessary to align the LC-MS data produced from multiple samples in order to compare them. Alignment methods attempt to match peaks in correspondence across replicates.

An initial approach towards alignment of replicates would be to spike a known amount of internal standards into each sample before running them through the LC-MS instruments. The peaks generated from the standards can be used as 'landmark' peaks to linearly shift the retention time in each sample, usually against a reference sample. Alternatively, labelling experiment can also be done by chemically labelling metabolites in two samples with isotopic reagents. The samples are then mixed before the LC-MS experiment and run through a single LC-MS run. The same metabolites from two samples would generally appear at close

retention time, making alignment easy. However, labelled experiments consume expensive reagents, are more difficult to prepare and harder to compare across laboratories and to various mass spectral databases online for identification. Consequently, it is common for LC-MS experiments to be performed label-free without relying on such labelling information. This is called label-free experiments. To be comparable, the results from these label-free experiments need to be aligned, using peak alignment methods.

Broadly speaking, the main challenge in the peak alignment stage of label-free experiments is caused by the poor reproducibility of retention time, with potentially large non-linear shifts and distortions across LC-MS runs produced from different analytical platforms. Consequently, most alignment methods correct for those shifts and distortions by finding – either explicitly or implicitly – a mapping function f that maps time t in one replicate to $f(t)$ in another. The mapping function f should be a monotonically smooth and increasing function, since elution orders of peaks that come out from the liquid chromatography instrument are generally preserved across replicates, at least for the data produced from the same LC-MS instruments. Alignment methods can therefore be broadly divided into two categories: warping and direct matching methods [18]. Warping methods perform RT correction of peak features before establishing their correspondences across replicates. Warping methods attempt to correct the RT drifts present across runs, by fitting an RT correction function (typically a regression model), using either the full LC-MS profile data or the peak feature data alone. Early warping approaches, such as dynamic time warping [19], correlation optimised warping [20] and parametric time warping [21], are predominantly based on dynamic programming, and use only the time information present in the Total Ion Chromatogram, although more recent warping approaches have started to consider the m/z dimension as well [22]. Once the time warping resulting in RT shifts have been corrected, the correspondence of peaks can be found through any method that matches peak features across runs.

The alternative approach towards alignment is the direct-matching methods, where the warping step is skipped and peak features are directly matched across replicates to establish their correspondences. Direct approaches therefore require that the peak (i.e. feature) extraction step has already been completed. Direct matching methods can be preferred in certain cases due to their simplicity, while still offering good performance [23]. The majority of direct matching approaches consist of two stages: computing feature similarity and using this similarity to match the features. A wide range of feature similarity measures have been proposed to compare the m/z and RT values of two peaks, including normalised weighted absolute difference [24], cosine similarity [25], Euclidean distance [26], and Mahalanobis distance [27]. Once similarity has been computed, feature matching can be established through either a greedy or combinatorial matching method.

Many approaches have been proposed for direct matching of peak features. Greedy direct-matching methods work by making a locally optimal choice at each step, in the hope that

this will lead to an acceptable matching solution in the end. RTAlign in MSFACTs [28] merges all runs and greedily groups features into aligned peaksets within a user-defined RT tolerance. Join Aligner [24] in MZmine2 merges successive runs to a master peaklist by matching features greedily according to their similarity scores within user-defined m/z and RT windows. Similarly, MassUntangler [26] performs nearest-distance matching of features, followed by various intermediate filtering and conflict-resolutions steps. Recent advances in direct matching methods have also posed the matching task as a combinatorial optimisation problem. Simultaneous Multiple Alignment (SIMA) [27] uses the Gale-Shapley algorithm to find a stable matching in the bipartite graph produced by joining peaks (nodes) from one run with peaks from another run that are within certain m/z and RT tolerances. [29] explores the application of the classical Hungarian algorithm to find the maximum weighted bipartite matching. BIPACE [25] establishes correspondence by finding the maximal cliques in the graph. SMFM [30] uses dynamic programming to compute a maximum bipartite matching under a relaxed bijective mapping assumption for time mapping.

Alignment methods can also be categorised depending on whether they require a user-defined reference run to be specified. When such reference is necessary, the full alignment of multiple runs is constructed through successive merging of pairwise runs towards the reference run (e.g. MZmine2's Join aligner in [24]. Alternatively, methods that do not require a reference run can either operate in a hierarchical fashion – where the final multiple alignment results are constructed in a greedy manner by merging of successive pairwise results following a guide tree (e.g. SIMA, described in [27]) – or by pooling features across runs and grouping similar peaks in the combined input simultaneously (e.g. the *group()* function of XCMS in [31]).

Label-free experiments pose many challenges in analysing replicates from different LC-MS runs. In particular, peaks from different runs can experience a potentially non-linear shift in retention time across chromatograms [32]. There is often a large amount of variations in the retention times across the replicates. Retention time variation could be due to instrument-specific factors (the condition of the chromatographic column itself. including flow rate variations, gradient slope and temperature [22]) or experiment-specific factors (e.g. instrument malfunctions or columns that need be replaced mid-experiment). Both factors are difficult to control, even in a careful experimental setting. Consequently, a single peak from one replicate can have several potential matches peaks in another replicate, whilst having no matches in another replicate. This is exacerbated by the uncertainties introduced due to parameter selections in the preceding steps of the pipeline. As a result, replicates produced by different LC-MS platforms or from different laboratories cannot be easily aligned to each other. In particular, the non-linear variation in retention time makes aligning technical replicates (which contains the same composition of metabolites) difficult and aligning biological replicates (which may not contain the same composition of metabolites) even more challenging.

Since large-scale untargeted metabolomics study can generate a huge number of samples (see [33, 34]), having a reliable and accurate peak alignment step during data preprocessing is important. Peaks that are improperly aligned can lead to false positives, and especially for untargeted label-free metabolomic experiments, the presence of even relatively small errors in any steps preceding the identification stage (including alignment) can result in significant differences to the final analysis and biological conclusions [35]. Errors or uncertainties inadvertently produced in any sub-step before identification would be carried forward forward in the pipeline. Improper preprocessing steps can also introduce variabilities that obscure important biological variations of metabolites themselves.

2.4.3 Peak Identification

The problem of identification of LC-MS data from peptides is referred to as peptide mass fingerprinting. As proteins are cleaved into peptides that are unique, the resulting fragmentation spectra are also expected to be unique to a protein. The theoretical peptide spectra can then matched against a reference spectra library. In practice, the resulting fragmentation spectra are not entirely unique and multiple hits can be returned from the spectra library, particularly in the case of libraries that have a large number of records. Identification is more difficult for metabolites due to the inherent complexity in metabolomics samples. A complete characterisation of the entire metabolome of any species is very difficult (for instance the human plasma and serum metabolome is still not fully characterised [36]).

Untargeted identification Metabolite identification is challenging in untargeted metabolomic studies due to the vast number of metabolites present in sample and the diversity in elements that comprise a metabolite. Unlike the genome that has four nucleotide bases as its sole alphabets, or proteins with twenty one amino acids as their building blocks, metabolites are harder to characterise structurally. The basic building blocks of a metabolite are atoms (commonly CHNOPS) that can be arranged in a variety of configurations in a single molecule alone. Similar to proteomics, the primary metabolite identification techniques relies on matching the accurate mass information of compounds to the set of chemical alternatives in a mass spectral database. The goals of the identification process are to distinguish between (in increasing levels of difficulty): **(1)** metabolites with different nominal masses, **(2)** metabolites with the same nominal masses, but different formula and monoisotopic masses, and finally **(3)** metabolites with the same nominal and monoisotopic masses, but different chemical structures (including chirals and isomers, such as leucine and isoleucine) [37].

Having a high mass accuracy is crucial here as it reduces the size of possible alternatives. However, even at the very high mass accuracy of 1 ppm, the number of possible formulae matched by accurate mass is still too large to allow for definite metabolite identifications [38]. Identification is particularly difficult for metabolites present in low abundance in the

samples. Consequently, widely-used metabolomics analysis tools like mzMine [24] employ sophisticated heuristics (such as the Seven Golden Rules) to narrow the formulae space based on various chemical constraints. Additional information such as the isotope patterns of compounds, and their fragmentation patterns (obtained from tandem MS), can also be used to help in accurate metabolite identification. Identification can also be performed on the basis of groups of peaks that have been gathered together in the ionisation product clustering step. For instance, the software tool CAMERA (Collection of Algorithms for MEtabolite pRole Annotation, [39]), can be used to perform the annotations of ionisation product species on groups of peaks, based on constructing a similarity graph and detecting highly-connected subgraphs in the graph. The same principle is used in the more recent probabilistic approach of MetAssign [40] that performs identifications of metabolites based on how well observed peaks fit to the relationship between theoretical distributions of adduct and isotopes.

2.4.4 Analysis

The last step in preprocessing of LC-MS data is the normalisation and visualisation of data. Normalisation is essential for removing any possible variation and systematic bias to allow for comparisons of differential levels of expressions of metabolites across samples. Statistical analysis is performed with visualizations in order to draw useful inferences from data – a step that is crucial in confirming or rejecting biological hypotheses. At this stage, the data is normalised to correct for systematic variations before statistical analysis. Spiked-in compounds that do not occur naturally are used for this purpose. Since the spiked-in compounds are expected to have equal concentration in all samples, they can be used to normalise peak areas in samples. Statistical analysis, such as t-test, ANOVA and principal component analysis, can then be performed on the normalised peaks across samples. The goal of statistical analysis is to answer biological hypothesis posed by life-science researchers. During the analysis, it is common to place the result obtained from metabolomic studies on the larger biological context by mapping them onto some biological pathways ([41, 42]) or in relation to other -omics studies ([43, 44]).

While targeted metabolomics focuses on a handful of specific metabolites, untargeted studies (such as in [33] and [34]) attempt to perform a global analysis of metabolites in the samples under study. Understanding the metabolome in an untargeted study is a challenging task due to the complex interactions of metabolites in the metabolome. Identification of specific metabolites are frequently not the final goal in untargeted metabolomics, rather it is the discovery of metabolites or groups of metabolites that are differentially expressed or correlated to the expression of specific physical traits being studied. Of particular interest is the detection of metabolites that act as disease biomarkers. The presence or absence of such metabolites can provide an indication to the corresponding presence or absence of dis-

ease in the organism [45]. Differences caused by genetic variations are also highly visible as changes in the metabolite composition of an organism. These could be quantified through differential analysis that compares the expression levels (abudance) of metabolites across samples. The resulting differential analysis provides biologists with a better understanding of the metabolic pathways in the cell and how they respond to perturbations. Differential analysis also underpins many practical applications of systems biology, such as nutritional research [46], drug discovery [47] and even in an integrative approach that combines genomics and metabolomics to obtain a more comprehensive picture of living organisms [44].

2.5 Conclusion

Software toolkit that deals with metabolomics data usually operate in a modular manner, where successive transformation of the raw LC-MS data happen by particular modules in the pipeline. However, it is important to highlight that despite the (apparently) serial pre-processing manner shown in 2.4, the actual workflow employed by life scientists is often iterative. For example, it is often the case that there are some peaks of low intensities that should be present but are found to be missing from a replicate. This requires the life scientist to go back to the peak detection stage, reduce the threshold used for noise filtering and repeat the pre-processing stages again from that point onwards. Another challenge common in bioinformatics data analysis in general is the lack of interoperability of different toolkits that deal with different parts of the pipeline. This often requires the user to 'hack' together an ad-hoc solution to perform data preprocessing that suits the needs of the research purpose. However, despite its many challenges, metabolomics is an exciting field with many open research problems.

Chapter 3

Machine Learning Background

Note:*[Machine learning stuff, around 10 pages ..?]*

3.1 Probabilities

Random variable

Marginalisation

Inference

3.2 Markov chain Monte Carlo methods

In Gibbs sampling, each model parameter is sampled in turn from its full conditional distribution until the random walk converges to the target distribution. The advantage of MCMC methods is that we obtain distributions over the model parameters, which allow us to quantify our uncertainties on them – as opposed to MLE method that provides only the most likely parameter values.

3.3 Mixture model clustering

In the clustering problem, we are presented with a list of feature vectors as input, and our goal is to separate those data points (features) into groups. Clustering is an instance of unsupervised learning where the learning algorithm tries to find hidden structure in unlabeled data – in constrast to supervised learning, where each data point comes with a class label. Many clustering algorithms exist, the simplest of which is k-means clustering. In k-means,

we assume that the data contains a fixed set of K clusters. Features are then assigned to the nearest cluster centroids based on some distance function. Each cluster centroid is updated by computing the average of all the features assigned to it. This process is repeated until convergence.

An alternative way to cluster data is through statistical model-based clustering. In mixture model clustering, each cluster is represented by a statistical distribution. The normal (Gaussian) distribution is commonly used to model continuous data, while the multinomial distribution is frequently used to model discrete data (for e.g. as topics in a document). The entire dataset can therefore be modeled by a finite mixture of mixture of several probability distributions, e.g. a Gaussian mixture model of two components can be used to model the distribution of heights in males and females from the sampled data. To illustrate with an example, here we construct a one-dimensional Gaussian mixture model on the retention time (RT) of our peak features. Each mixture component ideally corresponds to a metabolite, since peaks that share close RT values should originate from the same metabolite. This can be represented as the weighted finite sum of its K component distributions

$$p(\mathbf{y}|\boldsymbol{\mu}, \boldsymbol{s}, \boldsymbol{\pi}) = \sum_{k=1}^{K} \pi_k \mathcal{N}(\mu_k, s) \tag{3.1}$$

where $\boldsymbol{\mu} = \{\mu_1, ..., \mu_k\}$ are the component means, s is a fixed variance common to all components, and $\boldsymbol{\pi} = \{\pi_1, ..., \pi_k\}$ are the mixing proportions where $\pi_k > 0$ and $\sum_{k=1}^{K} \pi_k = 1$. The data points are represented as $\boldsymbol{y} = \{y_1, y_2, ..., y_n\}$ where y_n is the RT value of a peak feature. In this model, each retention time value is 'generated' by its k-th component Gaussian (a crucial modelling assumption here is all peaks are generated by a metabolite, which might not be true in the presence of noisy signals, ionisation products and other artefacts in the data). We denote this by the indicator variable z_{nk} where $z_{nk} = 1$ if feature n is assigned to component k, and 0 otherwise. Collectively, the indicator variable z_{nk} for all $n \in N$ and $k \in K$ can be stored inside the matrix $\mathbf{Z}$ of size N by K. Let $\theta = \{\boldsymbol{\mu}, \boldsymbol{\pi}, \mathbf{Z}\}$ denotes all the parameters of interest in the model. We now have $p(\boldsymbol{y}|\lambda)$ and we would like to infer $p(\lambda|\boldsymbol{y})$, in particular all the indicator variables in $\mathbf{Z}$ that tells us which data point goes into which mixture component (cluster), i.e. the cluster memberships. We get this by applying Bayes' rule:

$$p(\lambda|\boldsymbol{y}) = \frac{p(\lambda)p(\boldsymbol{y}|\lambda)}{\int p(\lambda)p(\boldsymbol{y}|\lambda)d\lambda} \tag{3.2}$$

The aim of inference here is to estimate model parameters from the posterior joint distribution $p(\lambda|\boldsymbol{y})$ of model parameters λ given the data $\boldsymbol{y}$. For non-trivial models, this often involves solving the complex integration on the denumerator on the right hand side of the equation above, which can be difficult (it's not analytically tractable). Instead, parameter estimations can be done through maximum likelihood estimation (MLE), usually through the

Expectation-Maximization algorithm, which finds model parameters maximising the likelihood of the model given the data. Alternatively, parameter estimations can also be done through Markov chain Monte Carlo (MCMC) methods. MCMC sampling allows us to approximate a target distribution via random walks obeying the Markovian property, where the current state in the random walk depends only on the previous state. When direct sampling of the posterior distribution is difficult but the full conditional distribution of each model parameter (e.g. $p(\mu_k|...)$ where ... denotes every other model parameter and the data) is easier to sample from, Gibbs sampling, an instance of MCMC methods, is often used.

3.4 Dirichlet Process mixture model clustering

We can avoid specifying the number of cluster K *a-priori* by assuming that the data is generated by a mixture of infinite number of components (taking the limit as K goes to ∞). Dirichlet Process is a stochastic process that describes a distribution of probabilty distributions, and is often used in Bayesian non-parameteric models – particularly as a prior distribution in Dirichlet Process (DP) mixture model. In non-parametric models, the model structure (e.g. the number of mixture components) is not fixed in advance *a priori*, but is instead determined based on the observed data. To do this, we place a Dirichlet process (DP) prior on the component parameters. Let θ_k denotes the component parameter of the k-th cluster. The DP can be viewed as an infinite dimensional generalisation of the Dirichlet distribution, where draws from the DP is itself a probability distribution. Following the example above, the RT data points can thus be explained by the following generative model:

$$G|\alpha, H \sim DP(\alpha, H) \tag{3.3}$$

$$\mu_k|G \sim G \tag{3.4}$$

$$y_n|\mu_k \sim N(\mu_k, s) \tag{3.5}$$

Similar to the finite case, $N(\mu_k, s)$ denotes the distribution of the data point y_n,which is a Gaussian distribution parameterised by mean μ_k and variance s (which is fixed, so we will not infer). The component parameter (the μ_ks) are conditionally independent given G, which is a discrete distribution drawn from the Dirichlet Process, and the data point y_n are conditionally independent given a component parameter μ_k. H is the base distribution that provides the prior on the μ_ks, while the parameter α can be seen as the inverse variance, with larger values of α producing smaller variance in the distributions drawn from the GP from H. The DP prior induces a partitioning on the data points, where the probability of a newly arriving data point to join an existing cluster is proportional to the number of data points already in that cluster. However, with a probability proportional to α, the data point will form a new cluster on its own. Additional details on Dirichlet process mixture model

clustering can be found in [48].

3.5 Hierarchical Dirichlet Process mixture model clustering

While the DP mixture model allows us to cluster related peaks together within each run, we would also like such clusterings to be shared across runs. This is reasonable to expect because if a cluster represents related peaks derived from a metabolite / compound, then we can expect to discover the same clusters across similar runs. The idea here is that: (1) peaks put together in the same 'global' clusters are basically aligned, and (2) we can define a model with entities that are meaningful in the biological sense, and thus discover insights from such models.

Suppose we have J runs to align. A Hierarchical Dirichlet process (HDP) is a distribution over a set of random probability measures, where each replicate has its associated random probability measure G_j. The global measure G_0 is distributed as a Dirichlet process, and the random measures G_j for each replicate is also distributed according to a DP, conditionally independent on G_0

$$G_0|\alpha, H \sim DP(\alpha, H) \tag{3.6}$$

$$G_j|\alpha_0, G_0 \sim DP(\alpha_0, G_0) \tag{3.7}$$

$$\mu_k|G_j \sim G_j \tag{3.8}$$

$$y_n|\mu_k \sim N(\mu_k, s) \tag{3.9}$$

Notice that the difference between DP mixture and HDP mixture is the fact that we are adding another level of hierarchy to the model, where the probability measure G_j for each replicate j is in turn drawn from the global measure G_0. A draw from the DP is a discrete probability measure, so G_0 and G_j are discrete distributions of point masses from the base distribution H. By drawing G_j, the j-th file specific measure, from a common global measure G_0, this makes it possible for us to share clustering parameters across different runs. In the popular Chinese Restaurant Franchise analogy described in [**?**], we have a Chinese restaurant franchise with a menu of dishes shared across all its restaurants (the global measure G_0). At each table (replicate cluster parameter) in a restaurant, one dish (global cluster parameter) is ordered from the menu by the first customer (data point) who sits there. The dish is shared by all customers who sit on that table. Newly arriving customer joins existing tables with a probability proportional to the number of people already sitting there, or sits on a new table by himself with a probability proportional to α_0. Existing dishes are also ordered based on its popularity across the franchise (the number of tables ordering it), or a new dish is created

with a probability proportional to α. In this hierarchical DP process, cluster parameter values are shared across runs and also within run.

3.6 Latent Dirichet Allocation

Latent Dirichlet Allocation (LDA), proposed in [**?**], is a probabilistic topic model widely used for unsupervised topic discovery. In the standard LDA model applied to text mining, documents comprise of some topics, each of which may produce the observed words in that document. Given a corpus of documents, the goal of inference in LDA is to approximate the posterior distributions of documents to topics and words to topics.

For the purpose of substructure discovery in MS2 data, a topic – explained as the set of recurring words shared in many documents – can be seen as corresponding to a substructure shared by many metabolites. Each topic then produces the observed MS2 fragment/loss words in an MS1 document. We assume the bag-of-word word model, where within each MS1 document, the observed MS2 fragment/loss word features are exchangeable. i.e. their ordering do not matter, only their observed counts matter. The input to LDA is therefore a matrix of the counts of occurences of MS2 word for each MS1 document. This can produced by concatenating the count matrices of the fragment words and the loss words produced in section **??** row-wise, e.g. if there are N_f unique fragment words and N_l unique loss words, both of which are shared across D MS1 peaks, the input matrix to LDA is a D-by-$(N_f + N_l)$ matrix. Entries in the matrix are the observed counts of words in the document, so they are the discretised intensity values of the fragment and loss words for each MS1 peak – produced according to Section **??**. We restrict the input to the standard LDA to take into account only the fragment and loss words because the counts of both fragment and loss words are derived from the normalised intensity values of the MS2 peaks.

The standard LDA model – as applied to substructure discovery – is now briefly described here. Given K predefined topics (indexed by $k = 1, ..., K$) corresponding to metabolite substructures, the observation of the n-th MS2 fragment/loss word in the d-th document (MS1 peak) can be described by the following generative process.

$$w_{dn}|\boldsymbol{\phi}_{z_{dn}} \sim Multinomial(\boldsymbol{\phi}_{z_{dn}}) \tag{3.10}$$

$$z_{dn}|\boldsymbol{\theta}_d \sim Multinomial(\boldsymbol{\theta}_d) \tag{3.11}$$

$$\boldsymbol{\theta}_d|\alpha \sim Dirichlet(\alpha) \tag{3.12}$$

$$\boldsymbol{\phi}_k|\beta \sim Dirichlet(\beta) \tag{3.13}$$

In other words: observation on the n-th MS2 word in the d-th MS1 peak (w_{dn}) is conditioned

on the assignment of word w_{dn} to some known k-th multinomial distribution (corresponding to a substructure). This assignment is denoted by the indicator variable z_{dn}, so $z_{dn} = k$ if w_{dn} is assigned to a k-th multinomial. The k-th multinomial distribution that an MS2 word is assigned to is characterised by the parameter vector $\boldsymbol{\phi}_{z_{dn}}$. However, $\boldsymbol{\phi}_{z_{dn}}$ is itself drawn from a prior Dirichlet distribution having a symmetric parameter β. The probability of seeing certain substructures (topics) for each d-th MS1 peak is then drawn from a multinomial distribution with a parameter vector $\boldsymbol{\theta}_d$. This parameter vector $\boldsymbol{\theta}_d$ is in turn drawn from a prior Dirichlet distribution having a symmetric parameter α. Figure **??** is the plate diagram of the standard LDA model, which shows the conditional dependencies between the random variables in the model.

Chapter 4

Incorporating Clustering Information into Peak Alignment

4.1 Introduction

According to [49], the objective function used for alignment can be improved by operating on the groupings of related IP peaks rather than at individual peak level alone. However, one of the alignment tools surveyed in Section 2.4.2 take into account this structural dependencies between related peaks produced by the same metabolite when solving the correspondence problem. Such information could potentially be used to improve the alignment process since a set of related peaks in one run should generally be aligned to another set of related peaks in the other run. As described in Section 2.4.1, related peaks are defined to be all those peaks that appear in a run due to the presence of one compound (peptide/metabolite) in the sample being analysed. Examples of related peaks are isotope peaks, multiple adduct and deduct peaks, and fragment peaks, elaborated further in Section 2.4.1. Such peaks should co-elute from the column and have similar chromatographic shapes and RT values. The related peak information can come from any peak grouping methods, of which clustering via RT is one instance, but one key assumption is that groups of co-eluting peaks corresponding to the same metabolite are generally preserved across runs.

In this chapter, we propose clustering the related peaks sharing similar RT values together, and using the information from the clustering process to modify the similarity score matrix used for the alignment (matching) of peaks across runs. This idea is illustrated in Figure 4.1 and further introduced in Sections 4.2 and 4.3. As shown in Figure 4.1, initial weights are computed between pairs of peaks in the two runs that are within m/z and RT tolerances (e.g. W_{AE} and W_{AJ}). When related peak information is added, the similarity between peaks A and E is increased due to peak A being related to another peak (B) that is similar to a peak (G) related to E. On the other hand, the similarity between A and J is not increased as

J does not have any related peaks that could potentially be matched to peaks related to A. In other words, we are proposing using the structural dependencies present between peaks in each run to modify the similarity scores and improve alignment performance: the more peaks related to A that could be matched to peaks related to E, the more likely it becomes that A should be matched to E.

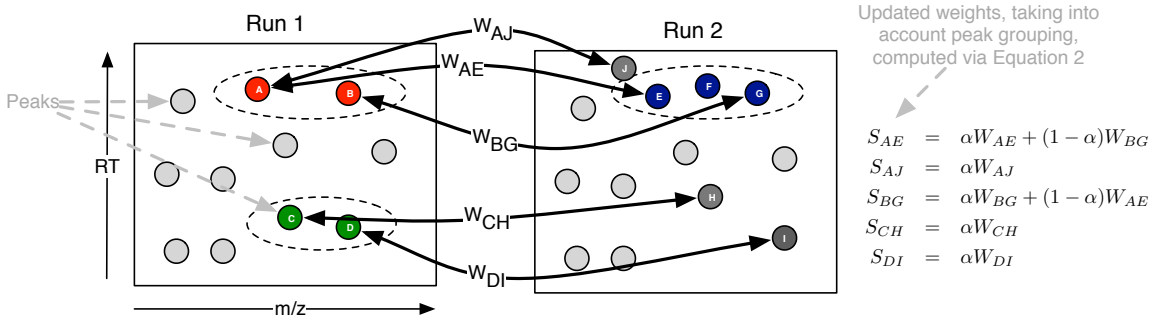


Figure 4.1: Illustrative example of the incorporation of grouping information into the similarity score. Each node in the figure is a peak feature, and dotted ovals represent groups of related peaks, e.g. isotopes, fragments, etc. Initially weights (e.g. W_{AE}) are computed for pairs of peaks (one from each run) with m/z and RT within pre-defined thresholds. These weights are converted into an overall score by incorporating grouping information. For example, peak pairs (A, E) and (B, G) are both within the threshold. Because A and B are in the same group, and E and G are in the same group, the weights between pairs (A, E) and (B, G) are upweighted. Peak J is not related to any peaks that could be matched with A's related peaks and the similarity between A and J is therefore downweighted (because $\alpha \leq 1$). The same applies to similarities between pairs (C, H) and (D, I).

Statement of Original Work

The idea of constructing alignment via approximate maximum weighted matching was proposed by the author. Simon Rogers conceived the idea of using the clustering information of related peaks to modify the similarity matrix used for matching. Code implementation and performance evaluation was carried out by the author.

4.2 Direct Matching

Our proposed alignment method combines a novel similarity score with maximum weighted bipartite matching. This results in pairwise alignments which can be, if desired, extended to multiple alignments with hierarchical merging strategy. In such merging strategies, having an accurate initial pairwise alignments is important because of its influence on the final multiple alignment results. In the following sections, we describe each step in more detail.

4.2.1 Feature Matching

A peak feature refers to a tuple of $(m/z, RT)$ produced as output after the initial peak detection stage of LC-MS data. Here, m/z is the mass-to-charge value and RT the retention time value of a peak feature. Suppose we wish to align run A containing N_A peaks with run B containing N_B peaks. Alignment between two runs can be represented as a matching problem on a bipartite graph G, where nodes in the graph are the features, edges are the potential correspondence between features and the weights on the edges are the similarity scores (entries in S) between features. In SIMA [27], the Gale-Shapley algorithm [50] is used to find a stable matching in G. A matching is stable if there are no two features in different runs that would prefer to be matched to each other than to their currently matched partners. Since the stable matching is computed based on ranked preference, valuable information could be discarded as distances between features are converted to ranks. As such, we prefer to use a method that maximises the total sum of similarity scores of matched features (maximum weighted matching).

The benefit of maximum weighted bipartite matching in solving the peak correspondence problem has been studied in [29] in their LWBMatch tool. LWBMatch shows that such matching method, coupled to a local regression method, is able to align runs having large and systematic drifts in RT values. The well-known Hungarian algorithm [51] attributed to Kuhn and Munkres is used in LWBMatch to solve this problem. The time complexity of the Hungarian algorithm is $O(n^3)$, where n is the number of peaks in the larger set. While the Hungarian algorithm's implementation can be improved to $O(n^2 log\, n)$ by using Fibonacci heaps for the shortest path computation, the polynomial time complexity required in this scheme is often too slow to be practical for alignments of the large number of runs produced in large-scale untargeted LC-MS studies. Consequently, we compute an approximation of the maximum weighted matching using a simple greedy algorithm that runs in $O(m\, log\, n)$ time, where n and m denote the number of vertices and edges in the bipartite graph G to be solved. The greedy algorithm is straightforward to describe: pick the heaviest edge e in G, where e represents a potential match between nodes (features). Add e to the matching solution M and remove all other edges adjacent to e from G. Repeat until all edges in G have been exhausted. This simple greedy algorithm is known to provide a lower bound of at least 1/2 of the maximum weight in the matching [52].

4.2.2 Feature Similarity

To define a similarity measure between peak features, we follow SIMA [27] in using the Mahalanobis distance between two peaks $\boldsymbol{p}_i \in A$, $\boldsymbol{p}_j \in B$ where each peak is a vector of its

m/z and RT values $\boldsymbol{p}_i = [m_i, t_i]^\mathsf{T}$ and $\boldsymbol{p}_j = [m_j, t_j]^\mathsf{T}$. The distance is given as:

$$D(\boldsymbol{p}_i, \boldsymbol{p}_j) = \sqrt{(\boldsymbol{p}_i - \boldsymbol{p}_j)^\mathsf{T} \boldsymbol{\Sigma}^{-1} (\boldsymbol{p}_i - \boldsymbol{p}_j)},$$

where the covariance matrix $\boldsymbol{\Sigma}$ is a diagonal matrix of mass-to-charge tolerance σ_m^2 and retention time tolerance σ_t^2. The diagonal covariance matrix $\boldsymbol{\Sigma}$ assumes independence between the σ_m^2 and σ_t^2 components. To reduce the computational burden, entries in $\boldsymbol{D}$ are only computed when the peaks' m/z and RT values are within σ_m and σ_t. We now define the similarity score between two peaks as one minus their normalised distance:

$$W(\boldsymbol{p}_i, \boldsymbol{p}_j) = 1 - \frac{D(\boldsymbol{p}_i, \boldsymbol{p}_j)}{D_{max}}, \tag{4.1}$$

where D_{max} is the maximum computed distance between peaks in the two runs being aligned. Collectively, we call the $N_A \times N_B$ matrix of similarity scores between all peaks in run A and B to be $\boldsymbol{W}$.

4.3 Incorporating Related Peak Groups

4.3.1 Combining Scores

The similarity score matrix $\mathbf{W}$ can now be combined with related peak information to obtain a final score, $\mathbf{S}$:

$$\boldsymbol{S} = \alpha \boldsymbol{W} + (1 - \alpha) \boldsymbol{L} \tag{4.2}$$

where $\boldsymbol{L}$ is the cluster similarity score between the two peaks in a single run (described below), and α ($0 \leq \alpha \leq 1$) is a parameter controlling the relative influence of the two components. To compute $\boldsymbol{L}$, we require related peak groupings from the two runs being aligned. This takes the form of an $N_A \times N_A$ matrix $\mathbf{C}^A$ for run A and an $N_B \times N_B$ matrix $\mathbf{C}^B$ for run B. Entries in $\boldsymbol{C}^A$ and $\boldsymbol{C}^B$ can be either binary (0, 1) or probability values, depending on the peak grouping algorithm used. For example, if a greedy clustering approach is applied to the features in run A, the ij-th element of $\mathbf{C}^A$ will be either 1 or 0, depending on whether the i-th and j-th features (peaks) in A are clustered together (1) or not (0). Note that in the following, we define the diagonal components of both matrices to be zero to avoid double counting. We then compute $\mathbf{L}$ as follows:

$$\boldsymbol{L} = \boldsymbol{C}^A \cdot \boldsymbol{W} \cdot \boldsymbol{C}^B. \tag{4.3}$$

The resulting matrix gives cluster similarity scores such that each element L_{ij} of $\boldsymbol{L}$ is the sum of weight from peaks in the same cluster as i in run A to peaks in the same cluster as

j in run B. This allows us to use the matrix $\boldsymbol{L}$ to upweight the similarity scores between peaks in the same cluster in one run that also have more potential matches to peaks in the same cluster in the other run of the matching. Computation of Equation 4.3 is illustrated in Figure 4.1. The ratio parameter α controls how much clustering information we bring into the overall similarity score matrix $\boldsymbol{S}$, with its value bounded in $0 \leq \alpha \leq 1$. Setting $\alpha = 1$ results in a matching that uses only information from $\boldsymbol{W}$, the similarity score matrix. Setting $\alpha = 0$ means that the matching is performed based only on the cluster similarity score $\boldsymbol{L}$. From our experience, a reasonable range of values for α lies between 0.2 to 0.4.

Our proposed approach is independent of the method used to group related peaks in each run. For comparison, we call our method that does not use the cluster similarity score ($\alpha = 1$) to be Maximum-Weighted (MW). We then demonstrate the performance improvement from incorporating related peaks information using two different clustering algorithms: a greedy RT clustering approach (described in Section 4.4) and a statistical mixture model (Section 4.5). The combination of matching with the greedy clustering is called MWG, while the alternative approach that uses the probabilities coming from the mixture model is called MWM.

4.4 Greedy Clustering of Related Peaks

In the greedy clustering method, the most intense peak in the dataset is selected and clustered with other candidate peaks inside a retention time window g_{tol}. The next most intense peak that has not already been clustered is processed, and the grouping process is repeated until all peaks are exhausted. If chromatographic peak shapes information is available (such as for the Metabolomic dataset used in section 4.7.2), the Pearson correlation coefficient between the chromatographic peak signals of the most intense peak and the candidate peaks are computed. Only candidate peaks with Pearson correlation values greater than some threshold c are accepted into the newly-formed cluster. This greedy clustering process results in binary grouping matrices $\mathbf{C}^A$ and $\mathbf{C}^B$ that can be used in eq. 4.3.

4.5 Mixture Model Clustering of Related Peaks

We can also group related peaks together by their RT values using a mixture model. Our observation consists of a vector of N observed peak's RT values $\mathbf{y} = (y_1, y_2, ..., y_n)$, and our aim is to partition each set of peaks into K groups of related peaks (clusters) by their RT values. We used a Gaussian mixture model with Dirichlet Process prior (described further in Section 3.4) to model the data. A peak is indexed by the variable $n = 1, ..., N$ and a cluster indexed by the variable $k = 1, ..., K$. Each Gaussian mixture component has some mean

μ_k are assumed to have a fixed precision (inverse variance) δ, corresponding to the fixed retention time tolerance for each group of related peaks. Let the indicator $\boldsymbol{z}_{nk} = 1$ denotes the assignment of peak n to RT cluster k. Then:

$$\begin{aligned} \boldsymbol{\pi}|\alpha &\sim GEM(\gamma) && (4.4)\\ \boldsymbol{z}_{nk} = 1|\boldsymbol{\pi}_k &\sim \boldsymbol{\pi}_k && (4.5)\\ \mu_k|\mu_0, \tau_0 &\sim \mathcal{N}(\mu_k|\mu_0, \tau_0^{-1}) && (4.6)\\ y_n|\boldsymbol{z}_{nk} = 1, \mu_k &\sim \mathcal{N}(\mu_k, \delta^{-1}) && (4.7) \end{aligned}$$

where $\boldsymbol{\pi}$ is the mixing proportions, distributed according to the GEM (Griffiths, Engen and McCloskey) distribution. The GEM distribution over $\boldsymbol{\pi}$ is parameterised by the concentration parameter γ and is described through the stick-breaking construction:

$$\begin{aligned} \beta_k &\sim Beta(1, \gamma) && (4.8)\\ \boldsymbol{\pi}_k &= \beta_k \prod_{l=1}^{k-1}(1 - \beta_l) && (4.9) \end{aligned}$$

The mixture component mean μ_k is drawn from a base Gaussian distribution with mean μ_0 and precision τ_0. We set μ_0 to the mean of the observed data, while τ_0 is set to a broad value of 5E-3. Analytical inference is not tractable here, so we use the Gibbs sampling scheme for inference. To do this, we need the conditional probability of $p(\boldsymbol{z}_{nk} = 1, ...)$ of peak n to be in an existing cluster k (or k^* if a new cluster is to be created), given any other parameters in the model. This conditional probability is given by:

$$P(\boldsymbol{z}_{nk} = 1|\mathbf{y}_n, \ldots) \propto \begin{cases} c_k \cdot p(\mathbf{y}_n|\boldsymbol{z}_{nk} = 1, ...) \\ \gamma \cdot p(\mathbf{y}_n|\boldsymbol{z}_{nk^*} = 1, ...) \end{cases} \quad (4.10)$$

where c_k is the current number of members (peaks) in an existing cluster k. $p(\mathbf{y}_n|\boldsymbol{z}_{nk} = 1, ...)$ is the likelihood of peak $\mathbf{y}_n$ in an existing cluster k. We can marginalise over all mixture components and get:

$$p(\mathbf{y}_n|\boldsymbol{z}_{nk} = 1...) = \mathcal{N}(\mathbf{y}_n|\mu_k, \lambda_k^{-1}) \quad (4.11)$$

where $\lambda_k = ((\tau_0 + \sigma c_k)^{-1} + \delta^{-1})^{-1}$ and $\mu_k = \frac{1}{\lambda_k}\left[(\mu_0\tau_0) + (\delta \sum_n \mathbf{y}_{n\in k})\right]$. Here, $\mathbf{y}_{n\in k}$ denotes the RT values of any peak n currently assigned to cluster k, and c_k the count of such peaks. The conditional probability of peak n to be in a new cluster k^* is:

$$p(\mathbf{y}_n|\boldsymbol{z}_{nk^*} = 1...) = \mathcal{N}(\mathbf{y}_n|\mu_0, \lambda_{k^*}^{-1}) \quad (4.12)$$

where $\lambda_{k^*} = (\tau_0^{-1} + \sigma^{-1})^{-1}$. In a step of the Gibbs sampling procedure, we perform the assignment of peak n to cluster k, creating new cluster k^* if necessary. Using the posterior summaries across all samples drawn $S^* = \frac{1}{R}\sum_{r=1}^{R} s_r$, where s_r is the r-th posterior sample collected after a suitable burn-in period and R is the total number of samples taken (excluding burn-in samples), we can obtain the marginal posterior of the probability of two features (peaks) to be in the same cluster k averaged across all samples. These probabilities comprise the elements of $\mathbf{C}^A$ and $\mathbf{C}^B$ (i.e. the ij-th element of $\mathbf{C}^A$ is the proportion of samples from run A in which peaks i and j were in the same cluster), which can be used in eq. 4.3.

4.6 Evaluation Study

Performance evaluation of alignment methods is difficult due to the lack of gold standard and evaluation criteria for benchmarking [8, 53]. Relatively few works, such as [23], exists that provide a comprehensive ground truth for evaluation. In fact, despite the numerous alignment methods that exist, most methods remain unevaluated, evaluated against a small number of alternatives or evaluated based on highly subjective criteria [18]. In particular, evaluation of alignment quality through manual visual inspection of superimposed profile images and some selected chromatograms is problematic and is not a systematic approach towards performance evaluation. While straightforward, the visual inspection of alignment quality is tedious and do not work for evaluation of a large number of aligned peaksets produced by the alignment of a large number of samples. It is also often subjective and might suffer from dissimilar interpretations across different experiments and datasets.

In this chapter, the performance of the proposed methods and other benchmark methods is evaluated using precision and recall on LC-MS datasets from proteomic, metabolomic and glycomic experiments. The proteomic datasets are obtained from [23] while the glycomic dataset comes from [1]. These datasets provide the ground truth for alignment and have used to benchmark alignment performance in other evaluation studies [23, 24, 26, 27, 1]. Additionally, we also introduce a metabolomic dataset generated from the standard runs used for the calibration of chromatographic columns [34]. The runs were produced from different LC-MS analyses separated by weeks, representing a challenging alignment scenario.

Many direct matching methods work in a pairwise fashion and produce an overall results via some merging strategies of intermediate results. Pairwise performance therefore limits overall performance, and as such, in this chapter, we focus on evaluation using only pairs of runs. Some (P2, metabolomic and glycomic) of the datasets selected for evaluation in our experiments have more than 2 runs, so we select only 2 runs each to form a training and testing set. The procedure for doing so is described in the respective following sections for each dataset.

4.6.1 Proteomic Datasets

[23] introduces two benchmark LC-MS proteomic sets (P1, P2) constructed to evaluate the ability of alignment tools in dealing with large retention time drifts. Both the P1 and P2 datasets were analysed using an automated LC-LC/MS-MS platform. Each dataset comes in multiple chromatography salt-step fractions, obtained by bumping the salt level at every 10 minutes interval during chromatographic separation. P1 was produced from *E. coli* samples digested by trypsin, and comes in 2 runs for each fraction. P2 was obtained from *M. smegatis* protein extracts, similarly digested by trypsin, and contains 3 runs for each fraction. P2 was constructed to be a greater challenge to align with runs separated by weeks. Alignment ground truth is established in [23] by means of peptides that can be reliably identified during the identification stage. Only identification annotations with SEQUEST Xcorr score >1.2 is included. Annotations are then filtered by their retention times and matched across runs.

For the proteomic datasets, each fraction in P1 has two runs used for alignment, while each fraction in P2 has three runs (we use only the first two to establish pairwise alignments). Tables 4.1 shows the number of features for each run of the P1 and P2 datasets used for evaluations. Both P1 and P2 represent challenging alignment cases, with large deviations in RT values across runs. This is especially true for P2 with LC-MS runs separated by weeks and large differences in the number of features per run. Further details on the nature of the datasets can be found in [23].

Fraction	# runs	# features per run (P1)	# features per run (P2)
000	2	5824	5054
		4782	5100
020	2	1114	3271
		1021	529
040	2	1230	1483
		958	678
060	2	1902	-
		1440	-
080	2	1183	474
		903	438
100	2	745	401
		581	429

Table 4.1: No. of features in the proteomic (P1 and P2) datasets. Note that fraction 060 is not present in P2.

4.6.2 Metabolomic Datasets

We use a metabolomic dataset generated from a mixture of 104 standard metabolites used for the calibration of chromatographic columns (details in [34]). These runs were produced by

ZIC-HILIC chromatography (Merck Sequant, Darmstadt, DE) on an UltiMate 3000 RSLC system (Thermo, Hemel Hempstead, UK), coupled to an Orbitrap Exactive mass spectrometer (Thermo, Hemel Hempstead, UK) in positive mode. The metabolomic dataset is available in different 11 runs, produced from different LC-MS analyses separated by weeks. While these runs are not true technical replicates, they are similar enough to be treated as replicates for the purpose of performance evaluation, and they represent a realistic and fairly challenging alignment scenario. The output from each of these runs is available in PeakML format, which were then converted into a suitable format using the mzMatch suite [54]. Both the original PeakML files and the converted text files can be found in our site. To generate the actual training and testing sets, 30 randomly pairs of runs were extracted as training sets, and another 30 pairs of runs extracted for testing sets. Table 4.2 shows the number of features in each run of the metabolomic dataset.

Metabolomic Run	**# features**	**Metabolomic Run**	**# features**
1	4999	7	6319
2	4986	8	4101
3	6836	9	5485
4	9752	10	5034
5	7076	11	5317
6	4146		

Table 4.2: No. of features in the full metabolomic dataset

Alignment ground truth was constructed from the putative identification of peaks in each of the 11 runs separately at 3 ppm using mzMatch's Identify module, taking as additional input a database of 104 compounds known to be present and a list of common adducts in positive ionisation mode (Table 4.3). This is followed by matching of features that share same annotations across runs to construct the alignment ground truth. Only peaks unambiguously identified with exactly one annotation are used for this purpose, as peaks with more than one annotations per run are discarded from the ground truth construction. The results from this process is an alignment ground truth for a smaller subset of peaks in the runs that can be reliably identified at high mass precision. Note that constructing alignment ground truth in this manner does not introduce bias to the ground truth as the identification information is not used during the alignment stage.

Adduct Types			
M+2H	M+H	M+ACN+H	M+H+NH4
M+NH4	M+ACN+Na	2M+ACN+H	M+ACN+2H
M+Na	M+2ACN+H	M+2ACN+2H	M+CH3OH+H
2M+H			

Table 4.3: List of common adduct types in positive ionisation mode for ESI.

4.6.3 Glycomic Dataset

[1] provides a glycomic dataset containing 23 runs, produced from untargeted LC-MS study for identifying N-glycan disease biomarkers in glyomics studies. LC-MS data were acquired from a Dionex 3000 Ultimate nano-LC system, coupled to an LTQ-Orbitrap Velos mass spectrometer on positive mode. Alignment ground truth is established in [1] based on a manual comparison of measured mass values with theoretical values (taking into account hydrogen adducts) and visual inspection of potentially incorrect assignments. We randomly extracted 30 pairs of runs for training and another 30 pairs of runs for testing performance evaluation from the full glyocomic dataset provided by [1], which comes in 23 runs in total. The following tables show the number of features in each run and the indices of the pairs of files randomly selected as training and testing sets in our Glycomic experiment.

Glycomic Run	# features	Glycomic Run	# features
1	856	13	911
2	1088	14	1144
3	922	15	932
4	808	16	1541
5	886	17	1022
6	850	18	1051
7	979	19	1119
8	1008	20	1047
9	904	21	1017
10	1043	22	990
11	1041	23	977
12	885		

Table 4.4: No. of features in the full glycomic dataset from [1]

4.6.4 Experimental setup

The alignment tools evaluated have in common user-defined mass-to-charge ratio (m/z) and RT window parameters. These parameters act as hard thresholds that determine the solution space to be explored in the m/z and RT dimensions when matching features. Performance of all alignment procedures is highly dependent on the assumptions and choice of parameter values that underpin them [18]. For example, warping methods must make assumptions regarding the mathematical form of the warping function and are dependent on a good choice of reference run. Direct matching approaches typically need to decide on the form of peak similarity function, and define some m/z and RT windows, outside of which, peaks cannot be matched. Whilst the m/z window and parameters can often be determined based on the mass accuracy of the measurement equipment, there is no obvious way to determine the

RT window and associated parameters. The optimal choice of such parameters could have a significant influence on the final results [18], and there is no reason to believe that these parameters should remain constant across different experiments.

Previous studies that use the proteomic dataset presented here [23, 26, 27] varied the window parameters and reported the best performance achieved. Whilst informative, this procedure is unrealistic due to the role of the ground truth in choosing the optimal parameter values. To provide a more realistic estimate of performance, we also present the performance on a separate testing set. In other words, we optimise the window parameters on one alignment task and report the performance when using these optimised parameters on a second task (distinct from the first task). This reflects the scenario where the parameters are set based on performance on a previous dataset or due to information supplied from the instrument manufacturer and tells us how critical setting these parameters is for each method.

In this chapter, *training set* refers to the data on which alignment parameters are optimised and *testing set* refers to the independent set on which alignment performance is evaluated. We believe that this represents a more realistic measure of alignment performance and provides us with some information as to how the different algorithms generalise to new datasets. We addressed the lack of comparative evaluation of alignment tools as discussed in [18] by independently reproducing key results from [23] and [27] for the Join and SIMA alignment methods. Our evaluation studies were performed on proteomic, metabolomic and glycomic datasets introduced before to validate the hypothesis that using related-peak information can improve alignment performance. Since most direct matching algorithms work in a pairwise fashion (pairs of runs are matched and the results combined), pairwise performance therefore limits overall performance, justifying the choice for our experiments. For the proteomic datasets, each fraction in P1 has two runs used for alignment, while each fraction in P2 has three runs (we use only the first two to establish pairwise alignments). Similarly for the metabolomic and glycomic datasets, we randomly extracted 30 pairs of runs for training and another 30 pairs of runs for testing performance evaluation.

Performance is evaluated in terms of precision, recall and F_1-score. Looking at pairwise matching, we can define the following positive and negative instances with respect to some pairwise alignment ground truth:

- True Positive ($\boldsymbol{TP}$): pairs of peaks that should be aligned and are aligned.
- False Positive ($\boldsymbol{FP}$): pairs of peaks that should not be aligned but are aligned.
- True Negative ($\boldsymbol{TN}$): pairs of peaks that should not be aligned and are not aligned.
- False Negative ($\boldsymbol{FN}$): pairs of peaks that should be aligned but are not aligned.

In the context of alignment performance, precision ($\frac{TP}{TP+FP}$) is the fraction of aligned pairs in the output that are correct with respect to the ground truth, while recall ($\frac{TP}{TP+FN}$) is the fraction of aligned pairs in the ground truth that are aligned in the output. A perfect alignment would have both precision and recall to be 1. In addition, we also computed the F_1 score (the harmonic mean of precision and recall) such that $F_1 = 2(precision \cdot recall)/(precision + recall)$. Only feature pairs present in the ground truth are considered for evaluation. The idea of using pairwise matching to define alignment performance evaluation is not new, and has also been done in [29]. Collectively for the purpose of performance evaluation, the set of Precision, Recall and F_1 values is referred to as a 'measurement'.

4.6.5 Other Alignment Tools For Comparison

Our proposed approach was benchmarked against MZmine2's Join Aligner [24] and SIMA [27]. Our own matching method (MW) also serves as a useful baseline to demonstrate any difference in performance with or without using clustering information. The two benchmark tools employ different approaches towards alignment. Join Aligner is a greedy direct-matching method, while SIMA is a combinatorial direct-matching method, with an optional warping step to correct RT shifts after an initial matching has been established. Users of the MZmine2's toolkit may have good reasons to prefer Join Aligner to the more recent RANSAC Aligner due to its simplicity and speed. Join Aligner produces a deterministic alignment output (so running it each time on the same input and parameters gives the same result), in contrast to the RANSAC aligner, which is non-deterministic. Join Aligner has relatively few parameters to configure, the most important ones being the *m/z tolerance* and *retention time tolerance* parameters. These parameters are used for thresholding and score calculations, and they were varied within reasonable ranges during our experiments. Similarly, the two most important parameters used in SIMA for thresholding and computing feature similarities are the $T_{(m/z)}$ and T_{rt} parameters (equivalent to our σ_m and σ_t). We let these two parameters vary in our experiments. SIMA also offers an optional step to correct for retention time distortion by constructing a smooth and monotonic warping function for the maximum likelihood alignment path after the initial matching has been done. The utility of this optional step is not obvious to end-users, since it requires additional parameters to configure and relies on having an initial correspondence established. Therefore, we chose to test only the core matching functionality in SIMA.

4.6.6 Parameter Optimisation

For every evaluated method in our experiments, we performed grid-search on the m/z and RT windows parameters using the training set. We then used those optimal parameters to

perform alignment on the testing set, giving us the respective performance measures (Precision, Recall, F_1) on the testing set. For testing set consisting of multiple fractions, we report the average performance measures on the testing fractions.

For training using the P1 and P2 datasets in the proteomic experiments, the m/z and RT tolerances were varied within: $\{1.0, 1.2, ..., 2.0\}$ for the m/z tolerance, and $\{5, 10, .., 300\}$ seconds for the RT tolerance. The parameter ranges were chosen based on reasonable estimates of the instrument's precision and prior RT tolerance values as reported by [23]. We kept all the default values for the remaining parameters in each evaluated tool, if any. For MWG, we also varied the ratio parameter α from $\{0.1, 0.2, ..., 1.0\}$ and the grouping parameter g_{tol} from $\{1, 2, ..., 10\}$ seconds and uses the combination that results in the best performance. For MWM, the ratio parameter α was varied from $\{0.1, 0.2, ..., 1.0\}$ but mixture model parameters were kept the same for clustering of all fractions in P1 and P2. When clustering all fractions in a dataset, a broad Gaussian prior was set for the component mean μ_j of each cluster j. The component precision s_j was set to 5 seconds, while the DP concentration parameter γ is set to 1. We drew 2000 posterior samples (with 1000 initial burn-in samples) for each run during the Gibbs sampling steps to construct the probability matrix of peak-vs-peak to be in the same cluster.

For the Metabolomic and Glygomic experiments, 30 pairs of run were randomly extracted from the M1 metabolomic dataset in [23] and from the glycomic dataset in [1]. These were assigned to be the training sets. Another 30 pairs of runs were extracted from each dataset to be the testing sets. Each pair of runs in the training set is assigned a partner pair of runs in the testing set. Parameters were optimised on pairwise runs in the training set and performance evaluated on the assigned partner runs in the testing set. For both datasets, the m/z tolerances used were $\{0.05, 0.1, 0.25\}$ and RT $\{5, 10, 15, ...100\}$ seconds. These ranges of parameters were selected in view of instrument accuracy and RT noise level of the LC-MS instruments that generate our metabolomic data and in [1]. The ratio parameter α was from $\{0.1, 0.2, ..., 1.0\}$ and the grouping parameter g_{tol} from $\{2, 4, ..., 10\}$ seconds for both datasets, and for the metabolomic dataset where chromatographic peak shapes information is available and used for greedy clustering in MWG, the threshold for the Pearson correlation coefficient between peak shape signals was varied from $c = \{0.70, 0.75, 0.80, 0.85, 0.90, 0.95\}$.

4.7 Results and Discussions

4.7.1 Proteomics Experiments

Single-fraction Experiment

Both P1 and P2 data consist of multiple fractions. In the first experiment, we investigate the best possible performance by using the same fraction as training and testing sets. As described in Section 4.6.6, on each training set (a fraction), we optimised the m/z and RT window parameters for alignments. The m/z parameters are in parts per million, normally notated 'ppm' and the range of m/z parameters used were $\{1.0, 1.1, ..., 2.0\}$ and RT $\{5, 10, ..., 300\}$ seconds. Parameters that control the grouping and influence of the cluster similarity score for our MWG and MWM methods were also optimised. The ratio parameter α was set to $\{0.1, 0.2, ..., 1\}$ for both MWG and MWM. The grouping tolerance g_{tol} was set to $\{1, 2, ..., 10\}$ seconds for greedy clustering, while the same hyperparameters were used for clustering of all fractions in case of mixture-model clustering (further details on parameter range selections are in Section 4.6.6).

The results are shown in Tables 4.5 and 4.6. We see that approximate maximum weighted matching (MW) alone performs competitively to other tools. On the P1 data (Table 4.5), incorporating grouping information (MWG, MWM) improves F_1 score performance over MW. MWG outperforms MWM, which may be due to the fact that the greedy approach is easier to optimise. For the P2 data (Table 4.6), which contains features with significantly higher RT drift across runs, again we find that MW is competitive and clustering information (MWG) improves performance for all fractions. The results here show the potential of our proposed approach: any peak grouping results expressed in a suitable matrix format can be incorporated into our method, and used as additional information during the matching stage. Figures 4.2 and 4.3 show how the benefit of incorporating clustering information is realised during matching: it allows the matching methods to explore regimes in the solution space having higher precision and recall values. On some training fractions, both methods that incorporate clustering information show significant increases in the best possible F_1 score. For dataset P1 fraction 000, this is an 11%-improvement for MWG and a 7.5%-improvement for MWM. For dataset P2 fraction 100, this is a 51%-improvement for MWG and 25%-improvement for MWM. Smaller improvements can be observed from other fractions in the Proteomic datasets too.

Multiple-fractions Experiment

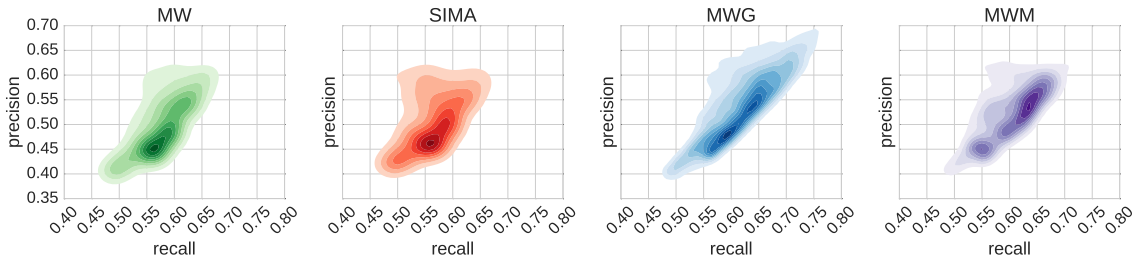
The single-fraction experiment does not represent a very realistic scenario as the optimal parameters were determined with respect to an alignment ground truth; practitioners might

Fraction	Join	SIMA	MW	MWG	MWM
000	0.63	0.64	0.64	**0.77**	0.71
020	0.88	0.88	0.88	**0.95**	0.90
040	0.82	0.83	0.85	**0.87**	0.86
060	0.76	0.78	0.78	**0.88**	0.83
080	0.90	0.89	0.88	**0.92**	0.90
100	0.89	0.89	0.89	**0.91**	0.91
Mean	0.81	0.82	0.82	**0.88**	0.85

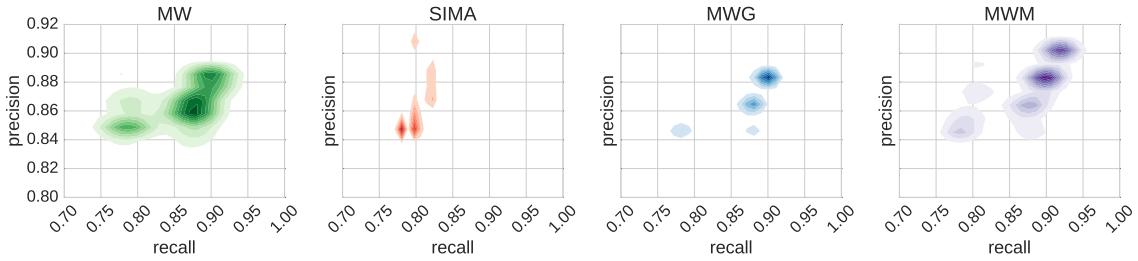
Table 4.5: F_1 scores for the single-fraction experiment results on the P1 dataset. The tool with the highest F_1 score for each fraction is highlighted in bold. The results for 'All' show the average F_1 scores of individual fractions.

Fraction	Join	SIMA	MW	MWG	MWM
000	0.45	0.45	0.45	**0.49**	0.45
020	0.77	0.78	0.79	**0.80**	0.79
040	0.77	0.78	0.77	**0.80**	0.77
080	0.66	0.68	0.67	0.67	**0.72**
100	0.55	0.58	0.56	**0.85**	0.70
Mean	0.64	0.65	0.65	**0.72**	0.69

Table 4.6: F_1 scores for the single-fraction experiment results on the P2 dataset. The tool with the highest F_1 score for each fraction is highlighted in bold. The results for 'All' show the average F_1 scores of individual fractions.

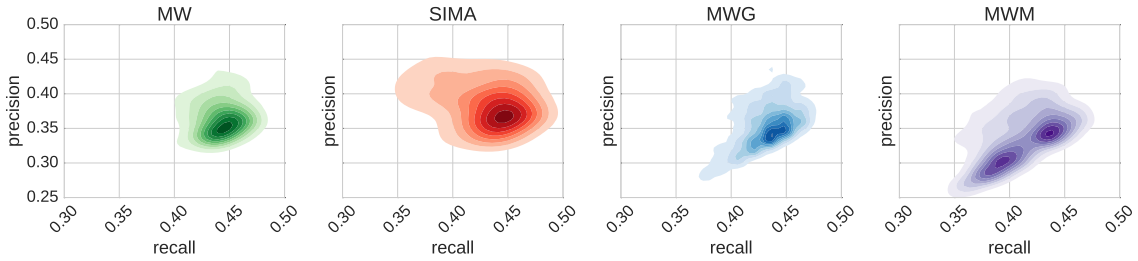


(a) P1 Fraction 000

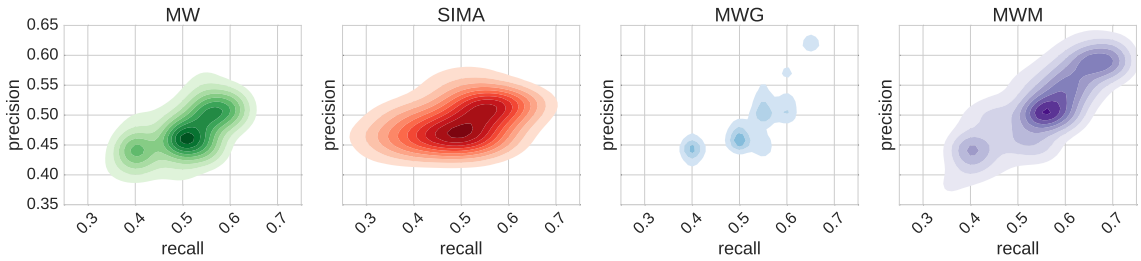


(b) P1 Fraction 100

Figure 4.2: Precision and recall training performance for all parameters (m/z, RT tolerance, α and g_{tol}) varied in the experiment for the fractions containing the most (a) and the least (b) number of features in the P1 dataset.



(a) P2 Fraction 000



(b) P2 Fraction 100

Figure 4.3: Precision and recall training performance for all parameters (m/z, RT tolerance, α and g_{tol}) varied in the experiment for the fractions containing the most (a) and the least (b) number of features in the P2 dataset.

not possess that information in real analytical situations. In this experiment, we improved upon the single-fraction experiments by using each fraction in each dataset as the training set and the remaining fractions as the testing set. Parameters were optimised on the training set and performance evaluations were performed on the testing set. This training-testing procedure produces 6 measurements for P1 and 5 measurements for P2, corresponding to the number of training fractions in each dataset. The overall F_1 score reported for each measurement is the average F_1 scores from individual testing fractions. The aim of this experiment is to investigate how well the different methods generalise to data that may have slightly different characteristics from that used to optimise the parameters – i.e. how critical the particular parameter values are.

Tables 4.7 and 4.8 show the F_1 score across measurements. On P1, the best overall performance is achieved by our methods that incorporate clustering information into alignment (MWG, MWM). On P2, the results are less homogeneous, with no method consistently performing best on all the different testing fractions. In the case of the noisiest data (dataset P2 fraction 000), our proposed approach incorporating greedy clustering (MWG) shows a decrease in overall testing performance instead. This is because the greedy clustering method used is sensitive to the choice of parameters and do not generalise well across the different fractions of P2. For instance, the best MWG's grouping tolerance parameter for fraction 000 is 5 seconds, while it is 1 second for fraction 080. The results suggest the dependence of our methods on the quality of groupings of related peaks in order to generalise well on

different runs. The heterogeneous testing performance in the multiple-fractions experiment of P2 shows that no method performs best and the choice of optimal parameters that work for certain runs do not generalise well to others on datasets with very high RT variability.

Training Frac.	Testing Performance				
	Join	SIMA	MW	MWG	MWM
000	0.82	0.85	0.82	**0.86**	**0.86**
020	0.78	0.76	0.78	**0.79**	0.75
040	0.78	0.76	0.77	0.79	**0.81**
060	0.78	0.78	0.77	**0.84**	0.83
080	0.71	0.73	0.72	0.77	**0.78**
100	0.75	0.77	0.74	0.76	**0.78**

Table 4.7: Multiple-fractions experiment results for the P1 dataset. For each training fraction, the reported testing performance is the average of individual F_1 scores from the testing fractions. The top-performing method (highest F_1 score) is highlighted in bold.

Training Frac.	Testing Performance				
	Join	SIMA	MW	MWG	MWM
000	0.62	**0.64**	0.61	0.48	0.61
020	**0.58**	0.56	0.55	0.43	0.55
040	0.52	**0.56**	**0.56**	0.41	**0.56**
080	0.56	0.50	0.50	0.50	**0.57**
100	**0.63**	0.57	0.56	0.44	0.57

Table 4.8: Multiple-fractions experiment results for the P2 dataset. For each training fraction, the reported testing performance is the average of individual F_1 scores from the testing fractions. The top-performing method (highest F_1 score) is highlighted in bold.

4.7.2 Metabolomic and Glycomic Datasets

We further explore the performance of our proposed methods on the metabolomic and glycomic datasets. From the full dataset, we randomly extracted 30 pairs of runs as the training sets and another 30 pairs of runs as the testing sets. Each training set is paired to a testing set. Parameters were optimised on the training set and the best attainable performance reported as the training performance. Generalisation performance is evaluated on testing sets using the optimal parameters from the training stage.

Figures 4.4 and 4.5 summarise the results from the experiments. We see that all methods perform better on the glycomic set than on the metabolomic set. This is explained by the fact that the metabolomic runs represent a generally more challenging alignment scenario with significantly more features to align. MW performs identically to SIMA on both datasets due to the similar form of Mahalanobis distance function used. This is despite the differences

in the actual matching method that establishes feature correspondences in SIMA and MW, emphasising the fact that the actual choice of matching function might be less important than other factors, such as the determination of similarity scores between peaks. On the glycomic dataset, adding clustering information improves the training performance, with an increase in the mean of the F_1 scores across 30 measurements from 0.89 (MW) to 0.93 (MWG) and 0.92 (MWM). This also translates into statistically significant improvements on the testing sets for both MWG (p=0.01, paired t-test) and MWM (p=0.002, paired t-test) over MW.

On the metabolomic dataset, where it is potentially harder to produce good clustering results due to the larger number of peaks and the more complex elution profile, we observe improvements in the mean of the F_1 scores from 0.83 (MW) to 0.90 (MWG) and 0.85 (MWM) on the training sets. These are also statistically significant improvements for both MWG ($p<0.001$, paired t-test) and MWM ($p<0.001$, paired t-test) over MW. The training results confirm our hypothesis that indeed incorporating clustering information (by modifying the similarity matrix used for matching in the proposed manner) can be used to help improve matching results over the case when such information is not used. However, this does not translate into any statistically significant improvements on the testing sets, suggesting that for the metabolomic dataset evaluated here, our proposed methods are also sensitive to parameter choices, and the choices of particular parameters (especially for the clustering step) that work on some runs may not generalise well to others. The results shown for running MWG on the metabolomic data in Figures 4.4 and 4.5 takes into account the Pearson correlations of the chromatographic shapes between peak features during the clustering process, since that information is available and straightforward to incorporate into the greedy clustering process. Results for MWG that consider only the RT values are presented and discussed in the following paragraph.

We also compared the results for MWG on both the training and testing sets on the standard metabolomic dataset when the greedy grouping is performed using only RT information (MWG (RT)) and when chromatographic peak shape correlations are also considered (MWG(RT+PS)) during the grouping process. Statistically significant differences can be observed on the training performance of Figure 4.6, with the mean of F_1 scores for MW 0.83, MWG(RT) 0.88 and MWG(RT+PS) 0.90. However, this does not translate to any improvements on the testing sets, with the mean of F_1 scores for MW 0.86, MWG(RT) 0.83 and MWG(RT+PS) 0.85. Introducing clustering information when only RT information is used during the clustering process (MWG(RT)) reduces testing performance. The training results suggest that where additional information such as chromatographic peak shapes are available, they should be used for the clustering step in the proposed methods. However, the lack of any statistically significant testing improvements between MW and MWG (RT+PS), suggest that the optimal parameters from training runs do not generalise well to different testing runs for the greedy clustering approach in general, especially for complex metabolomic runs,

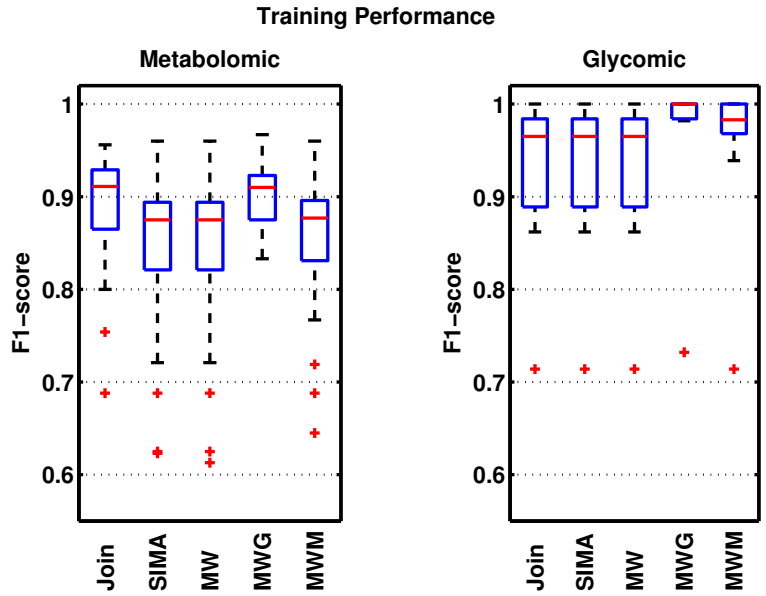


Figure 4.4: Training performance shows the best F_1 scores obtained by each method on 30 pairs of randomly-selected metabolomic and glycomic training sets.

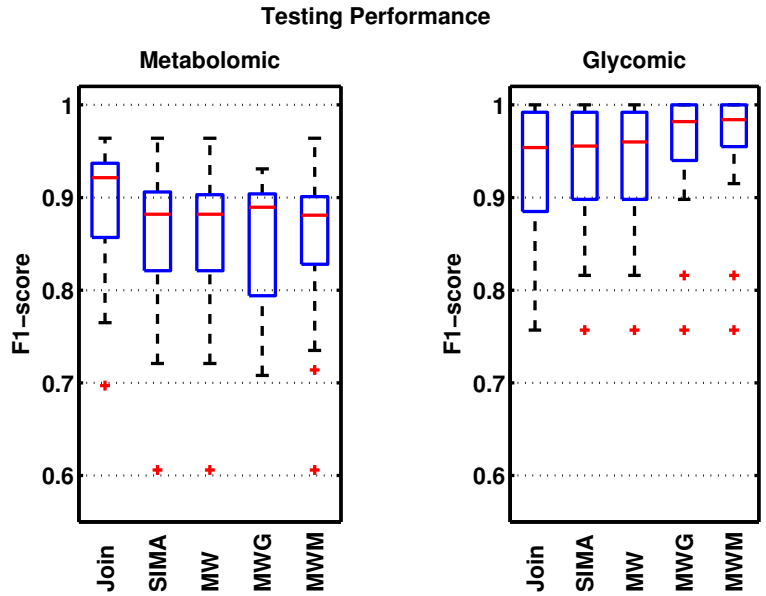


Figure 4.5: Testing performance shows how well each method generalise on the 30 different testing sets, each evaluated using the optimal training parameters from its corresponding training set.

with large number of features that tend to closely co-elute with each other.

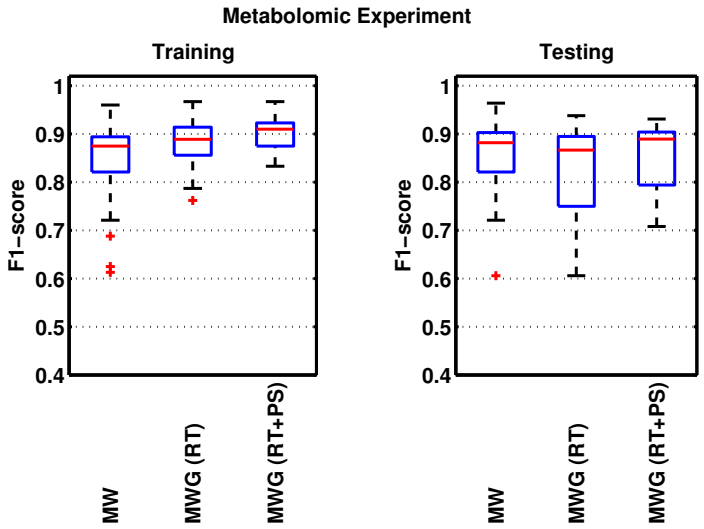


Figure 4.6: Comparisons in matching performance when greedy clustering with retention time (MWG(RT)) and peak shape correlations (MWG(RT+PS)) are used.

4.7.3 Running Time

Computational times of the proposed methods are primarily affected by the number of features in the runs being aligned and to some extent, the thresholding parameters used during similarity score computation and feature matching. Table 4.9 reports the measured running time for each proposed method using the parameters that give the best training performance). For each fraction being aligned, the running times were measured three times on a standard laptop with Intel Core i5 CPU running at 2.5 GHz, and the average value reported for matching only (MW), matching incorporating greedy clustering (MWG) and matching incorporating mixture model clustering (MWM). The time complexity of the mixture-model clustering step in MWM is $O(N)$ where N is the number of features in the run being clustered. We took 2000 posterior samples, discarding the first 1000 samples during the burn-in period. The number of samples were chosen to ensure convergence to the stationary distribution during inference.

4.8 Conclusion

In this chapter, we have proposed a novel peak matching method that incorporates related peak information to improve alignment performance. The method takes related peak information in the form of peak-by-peak binary or real-valued similarity matrices and as such is independent of the particular method used to compute these. The method fits into the

Fraction	Total Features	MW	MWG	MWM
000	10606	9	12	2700
020	2135	1	2	524
040	2188	2	2	540
060	3342	2	3	825
080	2086	2	2	505
100	1326	1	2	321

Table 4.9: Example measured execution time in seconds on fractions of the P1 dataset

category of *direct matching* approaches – those alignment approaches that do not perform an explicit time-warping phase. Our experimental results demonstrate the potential of this approach. From the training results, we see evidence of performance improvement across all evaluated datasets by incorporating grouping information into the matching process in the proposed manner. With the exception of the metabolomic dataset, both the greedy and model-based clustering approaches evaluated in our experiments rely only on the RT information for grouping related peaks. By looking at the testing performance, our results also explore the ability of the evaluated methods to generalise on different runs using less than optimal parameters. This is important because in the actual analytical situation of LC-MS data, neither the optimal parameters nor the alignment ground truth is known.

Note that our method relies on grouping of related peaks, and this introduces additional user-defined parameters. However, as our experiments have shown, in some settings, it may be much easier to produce good groupings of related peaks than accurately determine RT window parameters (the same grouping parameters were used for all evaluation datasets in the case of mixture-model clustering). Depending on the nature of the data, parameters relating to within-run characteristics (e.g. RT window for grouping related peaks) may be more likely to generalise across runs and experiments than parameters relating to between-run characteristics (particularly RT). For example, changes in the liquid chromatography (LC) column would likely result in related-peaks still co-eluting but could significantly change the absolute RT.

It would be interesting to investigate in greater detail any performance improvements that can be obtained from using other peak grouping methods, such as [55] that uses a mixture model of peak shape correlations or [40] that considers the dependencies between adduct and isotopic peaks when clustering. Exploring alternative approximate matching algorithms (such as the scaling algorithm in [52], which provides a $(1 - \epsilon)$ approximation of the maximum weighted matching in optimal linear time for any ϵ) and evaluating the benefits of incorporating different clustering approaches into our proposed alignment method are avenues for future work. Finally, the different alignment methods evaluated in this chapter also suffer from variable behaviours depending on the order of the runs being aligned [7]. This is particularly true in the case of alignment of multiple runs (typical in large-scale LC-MS

studies), where the final alignment results are often constructed through merging of intermediate alignments of pairwise runs. Different alignment methods may employ a different merging approach, for example, Join merges the intermediate results towards a reference run while SIMA allows the possibility of using a greedy hierarchical merging scheme. Systematic evaluation on how the chosen merging scheme may influence alignment performance is beyond the scope of this chapter and is an item for future work.

The related-peak based similarity score that underpins our approach could be applied to many other direct matching approaches, e.g. SIMA: [27] and similar ideas could also be incorporated into recently developed methods that take into account the presence of internal standards [1]. The evaluation datasets and pipeline developed over the course of our experiments in this chapter serves as the foundation for performance evaluations in subsequent chapters that follow.

Chapter 5

Precursor Clustering of Ionisation Product Peaks

5.1 Introduction

Chapter 4 explores the idea of using the grouping of related peaks to modify the similarity scores used for matching with the aim of improving alignment results. However, the grouping of related peaks used by the MWG and MWM methods in Chapter 4 is performed based on the retention time alone. Valuable information present in the mass domain and also in the chemical relationships of related peaks is not used in the grouping process. In this work, we extend upon the methods in the previous chapter and propose a novel Bayesian mixture model (PrecursorCluster) to perform the ionization product clustering of related peak features based on mass, retention time and a list of possible ionization transformations — bringing together peaks that share chemically meaningful relationships and can be related to a common precursor mass according to a set of transformation rules configurable by the user.

Building upon the results returned by PrecursorCluster, two alternative alignment methods (illustrated in Figure 5.1) are introduced for aligning IP clusters across runs: **(i)** ClusterMatch, a fast direct-matching method of IP clusters that uses the posterior precursor mass and RT values of IP clusters to compute the approximate maximum-weighted matching of the IP clusters and **(ii)** Cluster-Cluster, a second-stage clustering model that constructs alignment by means of grouping the IP clusters according to their likelihood of being assigned to the same same top-level cluster (in this manner, IP clusters assigned to the same top-level cluster are considered to be matched).

The aim of this chapter is to evaluate whether through the proposed methods, the matching of IP clusters can improve upon the matching of LC-MS peak features alone. For the

purpose of evaluations, two benchmark datasets of standard and beer mixtures, alongside their associated alignment ground truth and a list of 14 adduct transformations in positive ionization mode, were used. Using precision, recall and F_1-score as evaluation measures, the performance of the proposed method of matching IP clusters (Cluster-Match) were compared against the direct matching of peak features (MW) and its variant (MWG) that modifies the similarity matrix used during matching to bring together group of peaks related by RT closer during matching (described in Chapter 4). Additionally, the probabilistic matching results produced by Cluster-Cluster is also described, demonstrating that it is possible to use its output to extract aligned peaksets with varying degrees of confidence.

5.2 Related Work

According to [49], alignment objective function can be improved by operating on the groupings of related IP peaks rather than at individual peak level alone. In MetAssign [40], a Bayesian mixture model was introduced to perform the identification of a set of observed peaks based on how well they fit the theoretical mass spectrum of a metabolite computed from a given formula. While the groupings of related peaks extracted from PrecursorCluster naturally lend themselves to interpretation and can potentially be used in a similar manner as MetAssign, to perform a more robust annotation of metabolites present the sample, here we investigate its uses in improving the alignment step. Unlike MetAssign, PrecursorCluster does not require a prior library of possible metabolite formulas to be specified to perform ionization product clustering, relying only on prior chemical knowledge of which ionization transformations are expected to be present in the data. Unlike CAMERA [39], which approaches the problem of ion species annotation from a graph-theoretic point-of-view, PrecursorCluster is a fully probabilistic model, relying on Bayesian inference to update the probabilities of which LC-MS peak features can be explained by which transformations into IP clusters. This additional information can be used to provide an estimate to the uncertainty of IP annotations. The Bayesian model proposed in PrecursorCluster can also be easily extended to incorporate additional sources of information (e.g. chromatographic peak shapes) for clustering peaks in a different manner.

Since alignment is such an important part of the data preprocessing steps, it is useful to be able to robustly identify the uncertainty or confidence in the alignment results. In the absence of ground truth information (typically the case in untargeted metabolomics experiment), the user measures alignment quality through manual inspection or by comparing and visualising the summary statistics (e.g. median, standard deviation of retention time) across different replicates. Alignment methods that can produce matching confidence values is a big research gap that, to our knowledge, has not been addressed by any of existing direct-matching tools.

Tools such as MAVEN [56] assigns quality scores to individual peaks by training a predictive model on expert-annotated training data of peak quality metrics, but this does not extend to scoring groups of peaks. Other approach like [57] computes the Pearson correlations between intensity profiles of all peaks across replicates. Moving from these approaches towards a robust method that can provide confidence values for groups of aligned peaks across many label-free experiments is challenging research problem.

The subject of identifying and quantifying uncertainty has been extensively investigated in the problem of multiple sequence alignment (MSA) for genomics and transcriptomics. [58] attempt to quantify the alignment uncertainty of the popular MSA tool ClustalW [59], based on evaluations using synthetic data, and concludes that between half to all columns in their benchmark MSA results contain alignment errors. [60] construct a score that reflects the consensus between all possible pairwise alignments in T-COFFEE, while [61] propose GUIDANCE, a confidence measure obtained from pertubations of guide trees. Statistical approaches that provide a measure of confidence in alignment results have also been explored by [62] and [63], where the MSA results and phylogeny are constructed simultaneously, thus eliminating the need for a guide tree.

Despite the clear benefits of alignment uncertainty quantification in the sequence domain, the challenge of quantifying alignment results remains relatively unaddressed for the alignment of multiple runs in LC-MS-based-omics. Bayesian methods operating on profile data (e.g. [64, 65, 1]) and feature-based alignment methods (e.g. [66, 24, 27]) exist to correct RT drift, but in such methods, uncertainties are not propagated from the RT regression stage to the necessary peak matching stage that follows. Several recent feature-based alignment methods incorporate probabilistic modelling as part of their workflow, making it possible to extract some form of scores or probabilities on the alignment results. These methods are often limited to the alignment of two runs, which is not a realistic assumption in actual LC-MS experiments. For example, [67] propose an empirical Bayes model for pairwise peak matching. Matching confidence can be obtained from the model in form of posterior probability for any peak pair in two runs, however constructing multiple alignment results in [67] still requires a greedy search to find candidate features within m/z and RT-RT tolerances to a predetermined set of 'landmark' peaks. [68] describe PeakLink, a workflow for alignment that performs an initial warping using a fourth-degree polynomial. PeakLink poses the pairwise matching problem as a binary classification problem, where a Support Vector Machine (SVM) is trained based on an alignment ground truth derived from MS-MS information and used to differentiate matching and non-matching candidate pairs to produce the actual alignment results. While not explicitly included in the output of PeakLink, a matching score can be extracted from the SVM that represents how far each candidate pair is from the decision boundary. Note that these scores are not well-calibrated in the probabilistic sense, thus making comparisons of matching scores less straightforward. PeakLink is also not extended to

the problem of aligning multiple runs, although [68] state that it would be possible to do so with the choice of a suitable reference run.

5.3 Methods

The workflow is illustrated in Figure 5.1. A novel Bayesian model, **PrecursorCluster**, is introduced to group related peaks into IP clusters (Section 5.3.1). Each LC-MS run is processed separately through PrecursorCluster — the model takes as input the list of m/z, RT and intensity values of peak features and the list of user-defined transformations and produces as output the set of IP clusters per run. Alignment of IP clusters across runs are performed through **Cluster-Match** (Section 5.3.2) or **Cluster-Cluster** (Section 5.3.3). From Cluster-Match, a list of aligned peaksets (the set of peak features matched across runs) is obtained, while from Cluster-Cluster, the resulting aligned peaksets are produced alongside the probabilities of matching confidence.

5.3.1 PrecursorCluster: clustering of ionization product peaks

PrecursorCluster uses a mixture model to group the multiple ionization products that arise from each metabolite. We describe and evaluate the model for the positive ionization mode data, but the method could easily be adapted to negative mode data. In a run, the n-th peak feature is represented as the vector $\mathbf{d}_n = (d_n^m, d_n^t, d_n^p)$ with d_n^m the m/z value, d_n^t the RT value and d_n^p the intensity value of that peak. A list of T transformation functions of commonly-known IP types is also required (for e.g. see Table 5.1). A transformation function t_k takes as input the observed m/z value of a peak and produces as output the precursor mass into an IP cluster k under that transformation. This takes the form of $t_k(d_n^m) = \frac{d_n^m|c|+ce-\sum_i h_i G_i}{n}$, where c is the charge, e is the mass of an electron, n the multiplicity of the original molecule, and h_i and G_i are the count and atomic masses of the ith adduct part. For example, for $[M+H+NH4]^+$, c is 2, n is 1 while $\sum_i h_i G_i$ is the total atomic mass of $H + NH3$.

Although it is not strictly necessary, we found it useful to add some constraints to our mixture model. In particular, we make the assumption that an IP cluster must contain the $[M+H]^+$ ion peak and this must be the most intense peak. Although this will not always be the case, we found good performance under this assumption. These assumptions allow us to define the complete set of clusters — one for each peak, with the precursor mass of that cluster computed via assuming the peak is an $[M+H]^+$ ion. The k-th cluster is represented by the tuple (c_k^m, c_k^t, c_k^p), where the cluster's precursor mass c_k^m is the M+H transformed precursor mass of the respective peak's m/z value, and the cluster's RT (c_k^t) and intensity (c_k^p) values are the peak's RT and intensity values. Having created the set of clusters, an enumeration step is

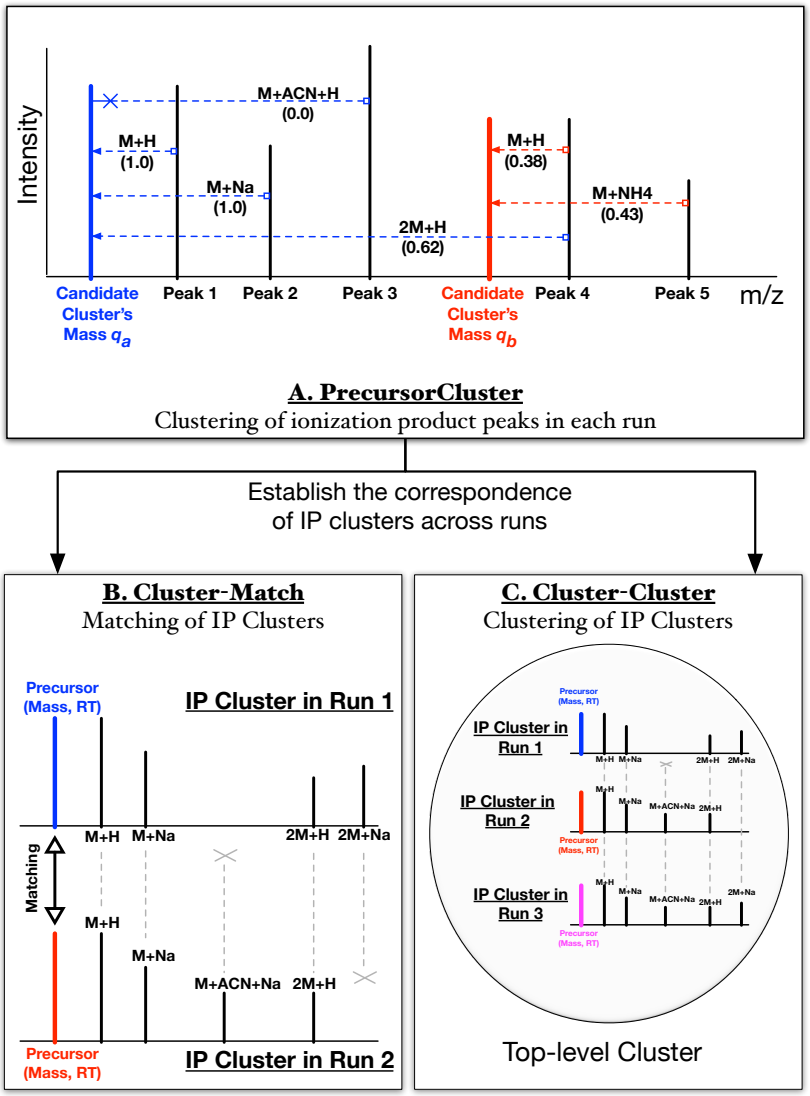


Figure 5.1: The proposed workflow. The input to PrecursorCluster is a list of m/z, RT and intensity values. During the enumeration stage, candidate IP clusters are generated from each peak through the M+H transformation. In this example, Peak 1 and Peak 4 generate candidate IP clusters with precursor masses q_a (blue) and q_b (red). In the inference stage, Peak 1 and Peak 2 are clustered to q_a through transformation M+H and M+Na with probabilities 1.0. Peak 3 has a valid transformation to q_a, but is not allowed to join that cluster since its intensity is $>$ than the intensity of the $[M + H]^+$ peak that generated the cluster (peak 1). Peak 4 can join the q_a cluster through the 2M+H transformation (with probability 0.62) or form its own candidate M+H cluster having the precursor mass q_b (with probability 0.38.) The latter allows for Peak 5 to join that cluster through the M+NH4 transformation (with probability 0.43). The final clustering is established by taking the *maximum a posteriori* assignment for each peak feature. Non-empty IP clusters can be aligned by matching their posterior precursor mass and RT values (Fig. 5.1B) or through a second-stage clustering process (Fig. 5.1C). The correspondence of peak features in matched IP clusters is constructed by grouping peak features having the same transformation types, shown as the gray dotted lines in Figures 5.1B & C.

performed to determine which possible clusters each peak can belong to. A peak $\mathbf{d}_n$ can be assigned to a possible cluster k if **(1)** the m/z value of that peak can be transformed (through any of the T transformations) into a precursor mass value that is within γ_m, the tolerance in parts-per-million (ppm), from c_k^m, **(2)** the RT value of that peak is within a certain tolerance (γ_t seconds) from c_k^t and **(3)** the intensity of that peak is less than the cluster's intensity threshold c_k^p. All observed peaks belong to at least one possible cluster (the one for which it is the $[M+H]^+$ peak).

Let $\boldsymbol{z}_{nk}$ denote the assignment of peak feature $\mathbf{d}_n$ to a possible cluster k, i.e. $\boldsymbol{z}_{nk}$ is 1 if peak n is assigned to cluster k and 0 otherwise. A peak can only be assigned to exactly one cluster ($\sum_{k=1}^{K} \boldsymbol{z}_{nk} = 1$). Following the standard mixture model construction, $\boldsymbol{z}_n$ is modelled as a multinomial distribution having the parameter vector $\boldsymbol{\theta}$, itself drawn from a prior Dirichlet distribution having the symmetric parameter α. The likelihood of a peak $\mathbf{d}_n$ being assigned to a cluster k depends on the likelihood of that peak's transformed precursor mass and RT values under the possible cluster's mass and RT values. Assuming independence between mass and RT terms, this is:

$$p(\boldsymbol{d}_n|\boldsymbol{z}_{nk}=1,...) = p(t_k(d_n^m)|\boldsymbol{z}_{nk}=1,...) \cdot p(d_n^t|\boldsymbol{z}_{nk}=1,...). \tag{5.1}$$

The likelihood of the transformed precursor mass $t_k(d_n^m)$ in the mass term $p(t(d_n^m)|\boldsymbol{z}_{nk} = 1,...)$ in eq. (5.1) is a product of two further terms. The first is the indicator function $I(n,t,k)$, set to 1 if no other peaks apart from $\mathbf{d}_n$ are currently assigned to cluster k through transformation t, and 0 otherwise. This allows each transformation type to appear only once in each cluster. We assume that the mass of cluster k, μ_k^m, has a Gaussian prior with mean c_k^m and fixed precision δ. The precision is set to reflect the mass tolerance in parts-per-million used during the enumeration of peaks to possible clusters, such that one standard deviation ($\sqrt{\delta^{-1}}$) is $\frac{\gamma_m * c_k^m / 1e6}{3}$. Within a cluster, we assume Gaussian noise in the mass, with the prior mass mean μ_0 set to the value of the cluster's precursor mass c_k^m used during enumeration and precision again equal to δ. The mass component of the likelihood is given by:

$$p(t_k(d_n^m)|\boldsymbol{z}_{nk}=1,...) = I(n,t,k) \cdot \mathcal{N}(t_k(d_n^m)|\mu_k^m, \delta^{-1}) \tag{5.2}$$

$$p(\mu_k^m|\mu_0,\delta) = \mathcal{N}(\mu_k^m|\mu_0,\delta^{-1}) \tag{5.3}$$

Similarly, Gaussian noise is assumed for the RT values. The k-th cluster has mean RT value given by μ_k^t and precision λ set to reflect the RT tolerance used during enumeration of possible assignments, i.e. γ_t is $3\sqrt{\lambda^{-1}}$. Within a cluster, the noise is assumed Gaussian, with

the prior RT mean ψ_0 set to the cluster's RT value c_k^t and precision λ:

$$p(d_n^m|\boldsymbol{z}_{nk}=1,\mu_k^t,\lambda)=\mathcal{N}(d_n^m|\mu_k^t,\lambda^{-1}) \tag{5.4}$$

$$p(\mu_k^t|\psi_0,\lambda)=\mathcal{N}(\mu_k^t|\psi_0,\lambda^{-1}) \tag{5.5}$$

A collapsed Gibbs sampling scheme is used to infer $\boldsymbol{z}_{nk}$, the assignments of peak n to cluster k (details in the next section). Averaging over the posterior samples, peaks are assigned to the most likely IP cluster based on their *maximum a-posteriori* (MAP) probabilities. The result from inference is the set of IP clusters, some of which may be empty and can be ignored, while others consist of related ionization products.

PrecursorCluster can be seen as a data-reduction procedure, taking as input the set of observed peak features per run and producing as output their MAP assignments into IP clusters. Non-empty IP clusters can now take the place of individual peak features as objects to be aligned. Each IP cluster in run j can be represented by $\boldsymbol{c}_{jk}=(\bar{q}_{jk},\bar{r}_{jk},\bar{\boldsymbol{u}}_{jk})$., with $\bar{q}_{jk}$ the IP cluster's posterior precursor mass value, $\bar{r}_{jk}$ the posterior RT value and $\bar{\boldsymbol{u}}_{jk}$ the adduct 'fingerprint' vector of length T for that IP cluster, created after the MAP assignments of observed peaks into the cluster. This stores the binary flags on which adduct transformations bring member peaks into that IP cluster (1 if that transformation brings a peak into the cluster and 0 otherwise). These posterior mass, RT and adduct fingerprint values are used during the latter alignment stage.

Gibbs Sampling for PrecursorCluster

For Gibbs sampling, the conditional distribution of a peak $\boldsymbol{d}_n$ currently being sampled to be placed in any of the K IP clusters is given by

$$P(z_{nk}=1|\boldsymbol{d}_n,\ldots)\propto(\alpha_k+n_k)\cdot p(\mathbf{d}_n|z_{nk}=1,...) \tag{5.6}$$

where n_k is the current number of members (peak features) in an IP cluster k, $\alpha_k=\frac{\alpha}{K}$ the symmetric prior on the Dirichlet distribution and $p(\mathbf{d}_n|z_{nk}=1,...)$ is the likelihood of peak $\mathbf{d}_n$ in a cluster k. Assuming independence between the mass and RT terms, the likelihood $p(\mathbf{d}_n|z_{nk}=1,...)$ can be factorised into its mass and RT terms (see eq. 5.1). However, the probability of a peak n to be placed in cluster k is 0 if the indicator function $I(n,t,k)$ in eq. (5.2) returns 0, i.e. another peak apart from n is already assigned to cluster k through transformation t. Otherwise, marginalising over all mixture components in eq. (5.2), the following posterior predictive distribution is obtained for the mass term:

$$p(t_k(d_n^m)|z_{nk}=1...) \;=\; \mathcal{N}(t_k(d_n^m)|\mu_k,\sigma_k^{-1}) \tag{5.7}$$

where $\sigma_k = (\delta(1+c_k)^{-1} + \delta^{-1})^{-1}$ and $\mu_k = \frac{1}{\sigma_k}[\delta(\mu_0 + \sum_n t(d^m_{n\in k}))]$. Here, $\sum_n t_k(d^m_{n\in k})$ denotes the sum of the transformed mass values of all the peaks (excluding the current peak being sampled) that have been assigned to cluster k, and c_k the count of such peaks. Similarly, the RT term in eq. 5.4 can be marginalized into

$$p(d^t_n|z_{nk}=1...) \quad = \quad \mathcal{N}(d^t_n|\mu_k, \sigma_k^{-1}) \tag{5.8}$$

where $\sigma_k = (\lambda(1+c_k)^{-1} + \lambda^{-1})^{-1}$ and $\mu_k = \frac{1}{\sigma_k}[\delta(\psi_0 + \sum_n d^t_{n\in k})]$, with $\sum_n d^t_{n\in k}$ denoting the sum of the RT values of all the peaks (excluding the current one) in cluster k.

5.3.2 Cluster-Match: direct matching of ionization product clusters

The ionization product clustering model described in Section 5.3.1 is essentially a data-reduction procedure, where within a single file j, the model takes as input the set of observed peaks in a single run and produces as output their groupings into IP clusters. Given the set of non-empty IP clusters and the peak features they contain, we can now treat IP clusters as a reduced set of features within a run and align (match) them across runs. We call this approach Cluster-Match. This contrasts to the conventional approach of matching all peak features directly to produce the alignment of peak features across runs.

As detailed in Chapter 4, in the direct matching alignment of two runs, the problem of establishing the matching between two runs can be viewed as finding the maximum weighted matching in a bipartite graph, where a node in the graph represents a peak feature, an edge represents a potential matching across two sides of the graph and the edge weight is the similarity between two potential matches. The MW method in Chapter 4 is an instance of a greedy algorithm that produces an approximation of at least 1/2 of the maximum weight in the matching of a bipartite graph [52]. Only peaks that are within mass and RT tolerances from each other across runs can possibly be matched (they have an edge linking them in the graph). While simple, the results in Chapter 4 shows that the MW method is generally competitive in performance to more sophisticated direct-matching methods, such as SIMA that relies on constructing stable-matching. We apply this direct matching methods to match IP clusters across runs, with IP clusters taking the place of individual peak features as nodes in the bipartite graph to be matched. The matching is therefore performed based on the precursor mass and RT values of IP clusters, rather than the observed peak's m/z and RT values. Once matching has been constructed, the alignment between the actual peak features in matched IP clusters can be established by grouping peaks that have the same transformation type across matched IP clusters (Figure 5.1B.)

To extend the above procedure to the alignment of multiple runs, two initial runs are first

aligned to construct an intermediate merged results. Consensus features are created by taking the average m/z and RT values of matched features, and the next run is then aligned to the merged results. This procedure is repeated until all runs have been exhausted. This match-merge scheme is commonly employed by other direct matching methods [27, 24] and requires selecting a reference run. In practice, the choice of reference run is arbitrary and its effect has not been fully investigated (in our implementation, the first run in alphabetical sorting is used as the reference run and the same ordering of runs is always used for all methods compared.)

5.3.3 Cluster-Cluster: across-run clustering of ionization product clusters

The proposed pairwise matching scheme used by Cluster-Match is frequently extended to the processing of multiple runs in a fairly ad-hoc manner (as seen in the match-merge approach at the end of Section 5.3.2, also commonly used by other direct matching tools). This approach suffers from the limitation of having to set a reference run for the matching and consequently, the fact that altering the ordering of runs to be processed might change the alignment results [18]. The alternative approach of generalizing from pairwise matching in a bipartite graph into finding the maximum weighted matching in a general graph is typically a computationally expensive operation. Producing a distance measure that works well for measuring similarities of peaks across runs is a non-trivial problem, and such matching procedures, whether through successive pairwise merging or operating on a general graph, generally do not take into account the uncertainties in the matching of peak features across runs.

Here, we propose using another clustering procedure (Cluster-Cluster) to further cluster the IP clusters produced from the first-stage IP clustering in Section 5.3.1. In this manner, IP clusters coming from different runs are further clustered into top-level clusters shared across runs (Figure 5.1C). The actual alignment of peak features can then be established by **(1)** looking at which IP clusters are put together into the same top-level cluster (essentially, their matching) and **(2)** in a top-level cluster, grouping peak features from different runs that have the same transformation type to establish their alignments. In this scheme, there is no need to set a reference run. Crucially, the posterior probabilities of certain IP clusters being assigned into the same top-level cluster provides us with an estimate of matching confidence of peak features.

Only peaks within a certain across-run mass tolerance should be matched, so a partitioning of IP clusters into top-level bins is performed. Across all runs $j = 1, ..., J$, IP clusters are sorted by their posterior mass values $\{\bar{q}_{jk}\}$. The smallest unprocessed mass value $min(\{\bar{q}_{jk}\})$

is used to initialize a top-level bin. Subsequent IP clusters (in ascending mass order) are grouped into the bin until an IP cluster with a posterior mass that differs by γ'_m ppm (a user-defined mass tolerance across runs) from $min(\{\bar{q}_{jk}\})$ is encountered, in which case, a new top-level bin is started using that cluster. The process is repeated until all IP clusters are processed.

If a top-level bin contains only one IP cluster, no possible matching can be constructed, otherwise IP clusters in the same bin can potentially be clustered (into top-level clusters) and therefore matched. To avoid specifying the number of top-level clusters *a priori*, we use an infinite Gaussian mixture model, described in Chapter 3, to model the data. Let $\bar{\boldsymbol{z}}_{jki} = 1$ denote the assignment of IP cluster k coming from file j into top-level cluster i.

Then:

$$\boldsymbol{\pi}|\alpha' \sim GEM(\alpha') \tag{5.9}$$

$$\bar{\boldsymbol{z}}_{jk}|\boldsymbol{\pi} \sim Multinomial(\boldsymbol{\pi}) \tag{5.10}$$

$$\boldsymbol{c}_{jk}|\bar{\boldsymbol{z}}_{jki} = 1, ... \sim p(\boldsymbol{c}_{jk}|\bar{\boldsymbol{z}}_{jki} = 1, ...) \tag{5.11}$$

where $\boldsymbol{\pi}$ are the mixing proportions, distributed according to the GEM (Griffiths, Engen and McCloskey) distribution (details in Chapter 3). The likelihood of $\boldsymbol{c}_{jk}$, the k-th IP cluster from run j, to be placed in a top-level cluster i is assumed to be factorized into independent factors of its mass, RT and adduct signature terms:

$$\begin{aligned} p(\boldsymbol{c}_{jk}|\bar{\boldsymbol{z}}_{jki} = 1, ...) = p(\bar{q}_{jk}|\bar{\boldsymbol{z}}_{jki} = 1, ...) \cdot p(\bar{r}_{jk}|\bar{\boldsymbol{z}}_{jki} = 1, ...) \cdot \\ p(\bar{\boldsymbol{u}}_{jk}|\bar{\boldsymbol{z}}_{jki} = 1, ...) \end{aligned} \tag{5.12}$$

In eq. (5.12), the mass term $p(\bar{q}_{jk}|\bar{\boldsymbol{z}}_{jki} = 1, ...)$ is defined analogously to the first-stage clustering step (Section 5.3.1). The indicator function $\bar{I}(k, j, i)$ in eq. (5.13) is set to 1 if there are no other IP clusters from run j, apart from the k-th IP cluster, that are assigned to the i-th top-level cluster, and 0 otherwise. This ensures that there is at most one IP cluster from each run assigned to a top-level cluster. The IP cluster posterior mass $\bar{q}_{jk}$ is distributed according to a Gaussian distribution with mean c_m and precision $\bar{\delta}$, where the across-run mass tolerance γ'_m is set to be equivalent to 3 standard deviations in ppm. The mass of top level cluster c_m is in turn drawn from a base Gaussian distribution having prior mass mean $\bar{\mu}_0$ and precision σ_m (eq. 5.14). The $\bar{\mu}_0$ parameter is set to the mean of the posterior m/z values of the IP clusters in the top-level bin, while σ_m is set to a broad value of 5E-3.

$$p(\bar{q}_{jk}|\bar{\boldsymbol{z}}_{jki} = 1, c_m, \bar{\delta}, ...) = \bar{I}(j, i) \cdot \mathcal{N}(\bar{q}_{jk}|c_m, \bar{\delta}^{-1}) \tag{5.13}$$

$$p(c_m|\bar{\mu}_0, \sigma_m) = \mathcal{N}(c_m|\bar{\mu}_0, \sigma_m^{-1}) \tag{5.14}$$

In the RT term $p(\bar{r}_{jk}|\bar{\boldsymbol{z}}_{jki} = 1, ...)$, $\bar{r}_{jk}$ is distributed according to a Gaussian distribution with mean c_t and precision $\bar{\lambda}$ (eq. 5.15). Again, the across-run RT tolerance γ'_t is set to be equivalent to 3 standard deviations in seconds. The same uninformative parameter values are set on the prior RT mean parameter $\bar{\psi}_0$ and precision σ_t (eq. 5.16).

$$p(\bar{r}_{jk}|\boldsymbol{z}_{jki} = 1, c_t, \bar{\lambda}) = \mathcal{N}(\bar{r}_{jk}|c_t, \bar{\lambda}^{-1}) \tag{5.15}$$

$$p(c_t|\bar{\psi}_0, \sigma_t) = \mathcal{N}(c_t|\bar{\psi}_0, \sigma_t^{-1}) \tag{5.16}$$

Finally, in the adduct fingerprint term $p(\bar{\boldsymbol{u}}_{jk}|\boldsymbol{z}_{jki} = 1, ...)$, the vector $\bar{\boldsymbol{u}}_{jk}$ is modelled using a multinomial distribution having a Dirichlet prior with symmetric hyper-parameter β. The entire likelihood function of eq. 5.12 ensures that IP clusters from different runs are placed in a single top-level cluster if: **(1)** they are from different runs, **(2)** they share similar posterior precursor mass and RT values, and **(3)** they have similar adduct fingerprint. Inference on model parameters is again performed via Gibbs sampling. Within each posterior sample, peak features in matched IP clusters sharing the same transformation type are grouped (Figure 5.1C), forming aligned peaksets. The occurrences of aligned peaksets are counted and averaged across samples to give matching confidences.

Gibbs Sampling for Cluster-Cluster

Analytical inference is not tractable here, so we use a collapsed Gibbs sampling scheme for inference of Cluster-Cluster. The conditional probability of $P(\bar{\boldsymbol{z}}_{jki} = 1|...)$ of IP cluster k in file j to be placed in an existing top-level cluster i (or i^* if a new top-level cluster is to be created), is given by:

$$P(\bar{\boldsymbol{z}}_{jki} = 1|\boldsymbol{c}_{jk}, \ldots) \propto \begin{cases} n_i \cdot p(\boldsymbol{c}_{jk}|\bar{\boldsymbol{z}}_{jki} = 1, ...) \\ \alpha' \cdot p(\boldsymbol{c}_{jk}|\bar{\boldsymbol{z}}_{jki^*} = 1, ...) \end{cases} \tag{5.17}$$

where n_i is the current number of members (IP clusters) in an existing top-level cluster i. $p(\mathbf{y}_n|z_{nk} = 1, ...)$ is the likelihood of peak $\mathbf{y}_n$ in an existing cluster k. The top part of eq. (5.17) is the conditional probability on existing mixture components of the model, and can be factorized into its independent mass, RT and adduct fingerprint terms. The bottom part of eq. (5.17) represent new components that are created as needed.

1. For the mass term $p(\bar{q}_{jk}|\bar{\boldsymbol{z}}_{jki} = 1, ...)$, we obtain the following predictive distribution after marginalizing over all mixture components:

$$p(\bar{q}_{jk}|\bar{\boldsymbol{z}}_{jki} = 1, ...) \quad = \quad \mathcal{N}(\bar{q}_{jk}|\mu_k, \gamma_k^{-1}) \tag{5.18}$$

where $\gamma_k = ((\sigma_m + \bar{\delta} n_i)^{-1} + \bar{\delta}^{-1})^{-1}$ and $\mu_k = \frac{1}{\gamma_k}\left[(\sigma_m \bar{\mu}_0) + (\bar{\delta} \sum_j \sum_k \bar{q}_{jk \in i})\right]$. Note

that in the summation terms of μ_k, $\sum_j \sum_k \bar{q}_{jk \in i}$ denotes the sum of posterior mass values of IP clusters currently assigned to top-level cluster i (excluding the current IP cluster being sampled), and n_i the count of such IP clusters.

2. Similarly, the RT term $p(\bar{r}_{jk}|\bar{\boldsymbol{z}}_{jki} = 1, ...)$ can be marginalized into a Gaussian with precision $\gamma_k = ((\sigma_t + \bar{\lambda} n_i)^{-1} + \bar{\lambda}^{-1})^{-1}$ and mean $\mu_k = \frac{1}{\gamma_k}\left[(\sigma_t \bar{\psi}_0) + (\bar{\lambda} \sum_j \sum_k \bar{r}_{jk \in i})\right]$, with $\sum_j \sum_k \bar{r}_{jk \in i}$ the sum of posterior RT values of member IP clusters in top-level cluster i, excluding the current IP cluster being sampled.

3. Lastly for the adduct fingerprint term $p(\bar{\boldsymbol{u}}_{jk}|\bar{\boldsymbol{z}}_{jki} = 1, ...)$, we marginalize over the mixture components and obtain $\frac{C\left(\bar{\boldsymbol{u}}_{jk} + \sum_j \sum_k \bar{\boldsymbol{u}}_{jk \in i} + \boldsymbol{\beta}\right)}{C\left(\sum_j \sum_k \bar{\boldsymbol{u}}_{jk \in i} + \boldsymbol{\beta}\right)}$ with $\sum_j \sum_k \bar{\boldsymbol{u}}_{jk \in i}$ the sum of all adduct fingerprint vectors currently assigned to top-level cluster i (excluding the current IP cluster being sampled), $C(\boldsymbol{X}) = \frac{\prod_{j=1}^{m} \Gamma(\boldsymbol{X}_j)}{\Gamma(\sum_{j=1}^{m} \boldsymbol{X}_j)}$ and Γ the gamma function.

For new components, marginalising over the base distributions for the mass term results in a Gaussian with mean $\bar{\mu_0}$ and precision $\bar{\delta}^{-1} + \sigma_m^{-1}$. Similarly, for the RT term, this results in a Gaussian with mean $\bar{\psi_0}$ and precision $\bar{\lambda}^{-1} + \sigma_t^{-1}$. For the adduct term, this results in $\frac{C\left(\bar{\boldsymbol{u}}_{jk} + \boldsymbol{\beta}\right)}{C(\boldsymbol{\beta})}$.

5.4 Evaluation Study

5.4.1 Evaluation Datasets

Two metabolomics datasets were used for performance evaluation. The Standard dataset was generated from a mixture of 104 standard metabolites used for chromatographic columns calibration and has been used for performance evaluation in in Chapter 4. This dataset contains eleven runs and represents a challenging alignment scenario with large RT variability (runs were separated by weeks and generated from different instruments). A Beer dataset of three runs from one batch that is representative of the typical biochemical diversity in a complex metabolomics study is introduced. All runs were processed through PrecursorCluster using the same parameters and the list of transformations in Table 5.1. This list includes the common adduct transformations in positive ionisation mode, but optionally isotopes can also be included.

Alignment ground truth for both datasets was constructed from the putative identification of each run at 3 ppm using the Identify module from mzMatch [54], taking as input a database of the 104 standard compounds known to be present and the transformations in Table 5.1. Peak features with the same unique identifications are matched across runs, resulting in an alignment ground truth for a subset of all peaks. Only peaks present in the ground truth are

considered for evaluation. The Standard ground truth accounts for 304 aligned peaksets (the set of peak features matched across runs) spanning 1936 peak features across all Standard runs, while the Beer ground truth consists of 108 aligned peaksets of 300 peak features across all Beer runs.

Table 5.1: List of common adduct transformations in positive mode used for the precursor clustering of the Standard and Beer runs.

M+2H	M+H	M+ACN+H	2M+Na	M+H+NH4
M+NH4	M+ACN+Na	2M+ACN+H	M+ACN+2H	M+Na
M+2ACN+H	M+2ACN+2H	M+CH3OH+H	2M+H	

5.4.2 Performance Measures

Precision and recall are widely used to evaluate alignment performance [23, 24, 26, 27, 69], also in Chapter 4. To evaluate alignment performance on multiple runs, we propose a generalized definition of precision and recall that extends from the pairwise definition in Chapter 4. From an alignment method or the ground truth, a list of aligned peaksets is obtained. For example, an alignment method returns a list of two aligned peaksets $\{a, b, c, d, \}, \{e, f, g\}$ as output. When $l = 2$, this output can be enumerated into a list of 9 'alignment items' of all the pairwise combinations of features: $\{a, b\}, \{a, c\}, \{a, d\}, \{b, c\}, \{b, d\}, \{c, d\}, \{e, f\}$, $\{e, g\}, \{f, g\}$. Let M and G be the results from such enumeration from a method's output and the ground truth respectively. Each distinct combination of features in M and G can be considered as an item during performance evaluation. Intuitively, the choice of l reflects the strictness of what is considered to be a true positive item, with larger values of l demanding an alignment method that produces results spanning more runs correctly. In this manner, l goes from 2 to as many runs being aligned.

For a given l, the following positive and negative instances of alignment item can now be defined for the purpose of performance evaluation:

- True Positive ($\boldsymbol{TP}$): items that should be aligned (present in G) and are aligned (present in M).
- False Positive ($\boldsymbol{FP}$): items that should not be aligned (absent from G) but are aligned (present in M).
- True Negative ($\boldsymbol{TN}$): items that should not be aligned (absent from G) and are not aligned (absent from M).
- False Negative ($\boldsymbol{FN}$): items that should be aligned (present in G) but are not aligned (absent from M).

In the context of alignment performance, precision ($\frac{TP}{TP+FP}$) is therefore the fraction of alignment items in M that are correct with respect to some alignment ground truth G, while recall ($\frac{TP}{TP+FN}$) is the fraction of alignment items specified in G that are actually aligned in the alignment results M. By definition, a perfect alignment method would have precision and recall scores of 1. In practice, there is a trade-off between precision and recall, where increasing recall often results in lower precision and vice versa. To summarize these two numbers, we also report the F_1 score, which is the harmonic mean of precision and recall, defined as $F_1 = 2(precision \cdot recall)/(precision + recall)$. Since our alignment ground truth is usually smaller than the set of all pairs of peaks returned by a method, only those peaks present in the ground truth are considered for evaluation.

5.4.3 Evaluation Procedure

As the baselines for evaluation, we compare the performance of our proposed methods against the method of direct matching of peak features (MW) and its variant (MWG) that modifies the similarity matrix used during matching to bring together group of peaks related by RT closer during matching – described in Chapter 4.

To evaluate Cluster-Match, the procedure in Chapter 4 is followed. 30 random pairs of Standard runs were selected as the training set, and another 30 as the testing set. Matching tolerance parameters were varied within reasonable ranges (details in Section 5.4.4) on one pair in the training set, and parameters resulting in the best training performance (highest F_1-score) of one pair were used to align the associated pair in the testing set. The three Beer runs are too few to allow separation into training and testing sets, so each method is trained and evaluated on all three Beer runs. The direct matching of peak features (MW) and its variant (MWG) from in Chapter 4 that incorporates grouping information (based on RT and not mass) into the similarity matrix used for matching are used as a baseline.

To evaluate Cluster-Cluster, five sets of 2, 3, and 4 Standard runs were selected randomly as well as all 3 Beer runs. For each data set, parameters for Cluster-Match were varied to obtain the best attainable alignment performance. These are plotted alongside the results from Cluster-Cluster on the same data using a fixed (and potentially non-optimal) set of parameters. Cluster-Cluster was also run with and without the adduct fingerprint term to evaluate its importance. More details on parameter optimization can be found in Section 5.4.4.

5.4.4 Parameter Optimization

Following the parameter optimization procedure in Section 4.6.6, the same grid search on the m/z and Rt window tolerance parameters is used. The m/z and RT window tolerance

parameters define the maximum deviation acceptable for a candidate matching is allowed in the bipartite graph. The choice of m/z parameter is often determined by the accuracy of the mass spectrometry instrument and can be reasonably determined in advance. Due to RT drift, selecting the RT window is less straightforward.

For the evaluation of feature matching (MW, MWG) vs. cluster matching (Cluster-Match) on the Standard dataset, we performed grid-search on the m/z and RT windows parameters using the training set. The optimal training parameters are used to perform alignment on the testing set, giving the respective performance measures (testing Precision, Recall, F_1). On the Standard datasets, we varied the mass tolerance window of the methods tested within the range $\{2, 4, 6, 8, 10\}$ m/z and the RT tolerance window within $\{5, 10, 15, ..., 100\}$ seconds during the training stage. Parameter combinations that result in the best F1-score were then used for performance evaluation in the testing stage. For MWG, additional parameters are also required for the threshold t_g on greedy clustering of related peaks and α_g, the contribution on the different parts to the similarity score. We let t_g vary within $\{2, 4, 6, 8, 10\}$ seconds and α_g within $\{0, 0.2, 0.4, 0.6, 1.0\}$ in the training stage and use the best combinations of parameter values for the testing stage. The three Beer runs are too few to allow separation into training and testing sets, so each method is trained and evaluated on all three Beer runs using the previously-described parameters same as the Standards.

The following parameters were used for the first-stage clustering of the PrecursorCluster model for all the Standard runs being processed: within-run mass tolerance $\gamma_m = 5$ ppm, within-run RT tolerance $\gamma_t = 30$ seconds. For the Beer runs, we used the within-run mass tolerance $\gamma_m = 3$ ppm and the within-run RT tolerance $\gamma_t = 10$ seconds. The prior on the Dirichlet distribution α is set to 1.0 and Table 5.1 shows the list of common adduct transformations in positive ionization mode used for precursor clustering. 5000 posterior samples were obtained from Gibbs sampling.

For the second-stage clustering in Cluster-Cluster, the following parameters were used for all input Standard and Beer runs: across-run mass tolerance $\gamma'_m = 10$ ppm, across-run RT tolerance $\gamma'_t = 60$ seconds, α' the Dirichlet Process concentration parameter is set to 1000.0. As relatively few number of runs are being aligned in our experiments, the large value of $alpha'$ encourages more top-level clusters, each having fewer member IP clusters inside. β, the symmetric prior on the Dirichlet prior distribution for adduct signature vector is set to 0.1. Inference is performed on each top-level bin that has more than 1 IP clusters inside, with 500 posterior samples drawn for each top-level bin.

5.5 Results and Discussions

With PrecursorCluster, the large number of peaks present within a single LC-MS run can now be reduced to a smaller number of IP clusters, making alignment easier as fewer objects have to be matched across runs. Section 5.5.1 presents the results of running the ionization product clustering on the Standard and Beer datasets.

While the resulting IP clusters potentially have many uses (e.g. to the problem of annotation of related peaks and the identification of metabolites), peaks assigned to any IP cluster have now been annotated with the transformation type that brings them into the clusters. IP clusters can therefore be aligned across runs (through direct-matching or a second-stage clustering process) and their member peak features (sharing the same transformation type) matched to produce alignment. Section 5.5.2 demonstrated from our experiments how the proposed approach of direct-matching IP clusters can improve upon the matching of LC-MS peak features alone, while Section 5.5.3 describes how the resulting probabilities from Cluster-Cluster can be used to robustly quantify the matching uncertainties.

Being a direct matching method, Cluster-Match performs nearly as fast as alignment by matching of peak features alone while offering better performance. As Cluster-Cluster performs Bayesian inference on which IP clusters should be put together into the same top-level clusters, the alignment of LC-MS features can now be established without the need for a reference run. While this requires more computational time than the direct-matching alternative (Section 6.5.3), Cluster-Cluster is able to produce confidence scores on the matching quality of aligned peaksets from the posterior summaries computed during inference. This has a potential use in assisting the selection of high-confident aligned peaksets for subsequent analysis in the latter stage of the LC-MS pipeline

5.5.1 Ionization Product Clustering from PrecursorCluster

Within each run, PrecursorCluster produces the *maximum a posteriori* (MAP) assignments of peaks to IP clusters. An example of four IP clusters found in the Standard runs, identified as Cysteic acid, is shown in Figure 5.2. According to the ground truth, all member peaks across these four clusters should be aligned. Table 5.2 shows that within each run, a large number of peak features cannot be clustered to other peaks within the same run and can therefore only form an IP cluster with itself as the only member through the M+H transformation (we call these clusters of only one member peak the singleton IP clusters). In both the Standard and Beer runs, non-singleton IP clusters (containing more than one member peaks) comprise approximately 6% to 10% of the total IP clusters of that run. The distributions of the cluster sizes of these non-singleton clusters when only adduct transformations are used are given in Figure 5.3 for the Standard and Beer runs. We also note that for any given cluster size,

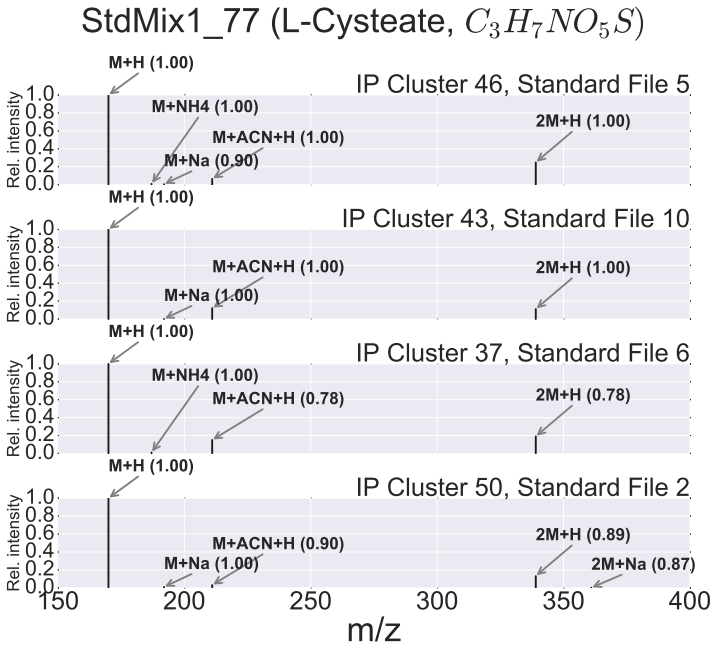


Figure 5.2: Different IP clusters (46, 32, 37, 50) in four different Standard runs, identified as Cysteic acid. The MAP transformation type of a peak and its probability are annotated as a labelled arrow and the bracketed number beside. According to the ground truth, all member peaks with the same transformation type should be aligned.

the counts of IP clusters of that size tend to differ significantly across the Standard runs, due to the varying number of LC-MS peak features present in each Standard run. This is the consequence of the Standard runs being produced in several batches separated over a period of time. The distributions of cluster sizes in Figure 5.3 across the three Beer runs are more consistently reproduced, reflecting the fact that the runs were generated within the same batch. As shown in Figure 5.3, the largest IP clusters of the Beer and Standard runs have 6 and 7 member peaks respectively.

Consistent with the number of singleton clusters, the M+H transformation dominate in the data. Non M+H transformations comprise 8% of the total MAP transformations for the Standard dataset and 10% for the Beer dataset. In both datasets, the M+ACN+H and M+Na transformations are highly prevalent (Figure 5.4). The presence of the M+ACN+H and M+NH4 transformations in the Beer dataset is expected, given the use of acetonitrile and ammonium carbonate buffers during chromatography. Similarly, the M+CH3OH+H adducts in the Beer data can also be explained by the use of methanol during the sample preparation process. The consistency of the example clusters in Figure 5.2 and the explainable transformations in Figure 5.4 suggest a valid result from PrecursorCluster, providing confidence that it can be used for alignment.

Table 5.2: The number of peak features and the counts of singleton and non-singleton IP clusters in each run of the Standard and Beer datasets. A singleton cluster is defined to be an IP cluster having only one member peak after MAP assignments, while a non-singleton IP cluster has more than one member peaks. The last column in the Table shows the counts of non-singleton IP clusters and also the percentage of non-singleton IP clusters from the total IP clusters in that run.

Data	# Peak Features	# Singleton IP Cluster	# Non- singleton IP Cluster
Std 01	4999	4327	301 (6.5%)
Std 02	4986	4341	288 (6.2%)
Std 03	6836	5755	481 (7.7%)
Std 04	9752	8011	775 (8.8%)
Std 05	7076	5801	551 (8.7%)
Std 06	4146	3655	216 (5.6%)
Std 07	6319	5272	469 (8.2%)
Std 08	4101	3579	232 (6.1%)
Std 09	5485	4789	312 (6.1%)
Std 10	5034	4304	310 (6.7%)
Std 11	5317	4574	337 (6.8%)
Beer 01	7553	6179	633 (9.3%)
Beer 02	7579	6203	631 (9.2%)
Beer 03	7240	5983	574 (8.6%)

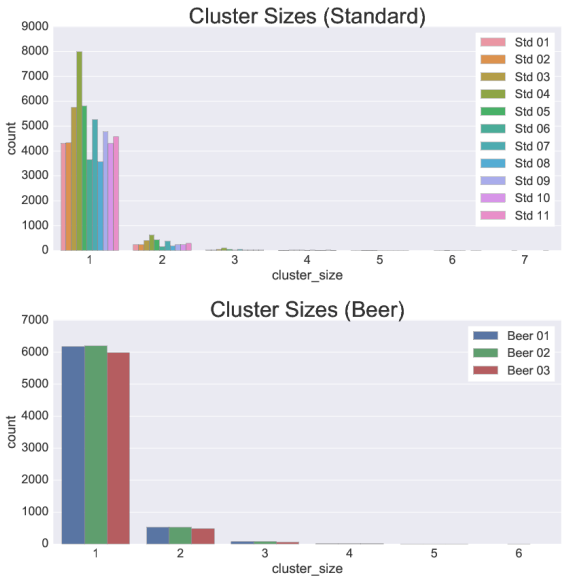


Figure 5.3: Ionization product cluster sizes for all runs in the Standard and Beer datasets. For any given size, the number of clusters are generally more consistent in the Beer runs compared to the Standard runs, which shows greater variability due to the differences in the number of peak features per run.

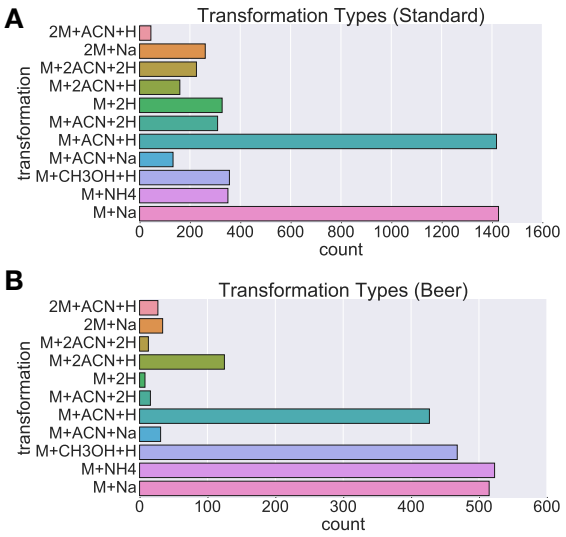


Figure 5.4: Barcharts showing the counts of transformation types in all Standard and Beer runs, excluding the M+H transformation.

5.5.2 Improved Alignment Performance from Cluster-Match

Precision and recall values produced by the different methods (across all parameter ranges) for the entire 30 Standard training sets and the 3 Beer runs can be found in Figure 5.5. Here, l (the size of peakset combinations to be considered during performance evaluation) to 2 to consider only pairwise features for performance evaluation as pairwise performance limits overall performance in direct matching methods that employ the merge-match scheme to construct an overall result. The results in Figure 5.5 (top row) shows that across all the m/z and RT window tolerances varied, Cluster-Match can produce higher precision while retaining similar recall values to feature matching (MW) or modified feature matching (MWG). This increase in precision comes from the increase of true positives and the decrease in false positives by taking into account the ionization product relationships between peak features when constructing the matching. The results here suggest that, regardless of the parameters selected for the m/z and RT tolerance windows, the proposed methods of matching by IP clusters can return a better alignment result compared to matching by peak features only.

Similar results can also be observed for the Beer dataset (Figure 5.5, bottom row). The complex Beer runs being aligned have minimal RT deviations when compared to the Standard runs, so all evaluated methods perform well, demonstrating smaller deviations in performance values despite varying the tolerances parameters. Again a general increase in precision of the results from Cluster-Match is observed over the other two baseline methods. MWG, which relies on the grouping of related peaks using their retention time values only, does not appear to produce any improvements over MW. The results here suggest that on complex LC-MS data such as the Beer data, the richer information present in the m/z and RT

values of related peak features, alongside their possible IP transformations and relationships to the precursor peak, is essential and has to be taken into account.

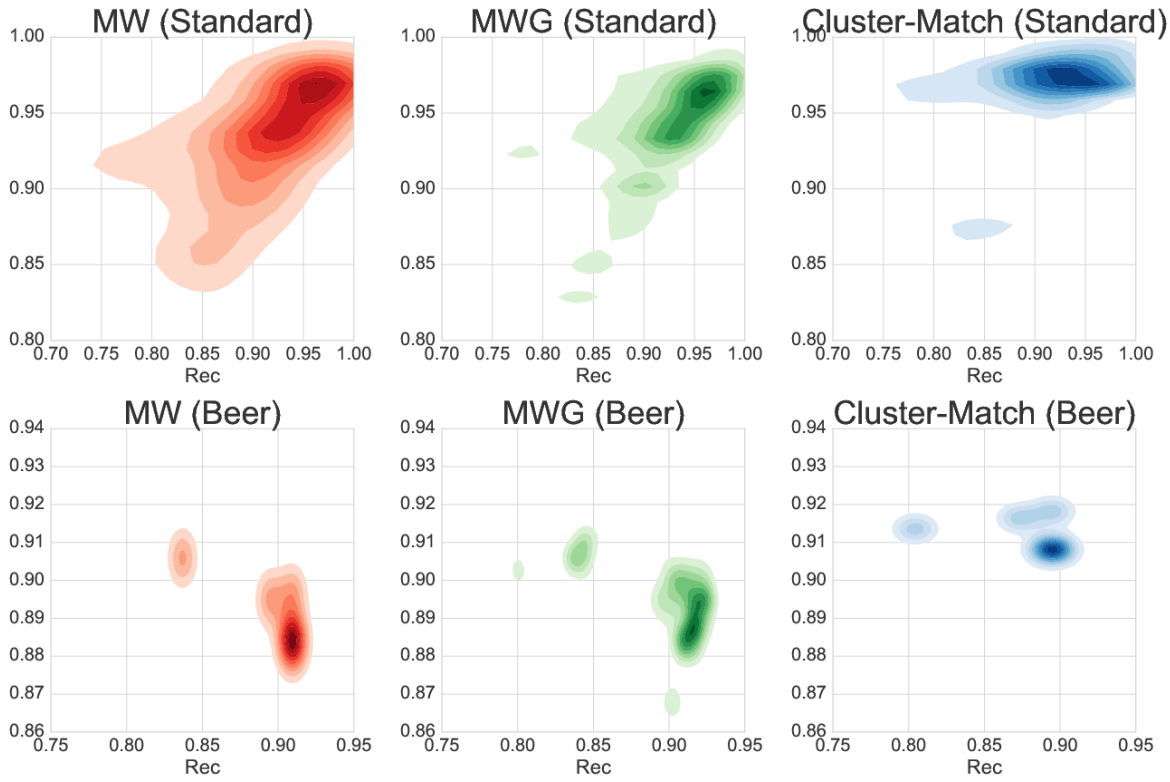


Figure 5.5: All the training results obtained by varying the m/z and RT window parameters from the alignment of the entire 30 sets of pairwise Standard runs (top row) and the 3 Beer runs (bottom row). For MWG, the grouping parameter t and score contribution α were also varied, while for Cluster-Match, the same set parameters of first-stage clustering was used for all input files.

Optimizing parameters on the training set and evaluating performance on the testing set measures how well a method generalizes to new and unseen data. The best Standard training and testing F_1-scores from each method are reported in Figure 5.6. Using a one-sided paired t-test, Cluster-Match is found to be statistically greater than that of MW in both the training (p-value=0.002) and the testing cases (p-value=0.026), suggesting that Cluster-Match generalizes better to new and unseen datasets. MWG produces even higher training F_1-scores compared to the other two methods. This difference is found to be statistically significant using a one-sided paired t-test (p-value=0.01). The higher training performance of MWG can be explained by the fact that the RT grouping tolerance parameter and matching ratio for MWG were optimized during the training phase, while the same (potentially non-optimal) clustering parameters were used for the ionization product clustering of all Standard runs. On the testing results, no statistically significant differences were found on the testing F_1-scores of MWG and Cluster-Match, suggesting that both methods generalize well.

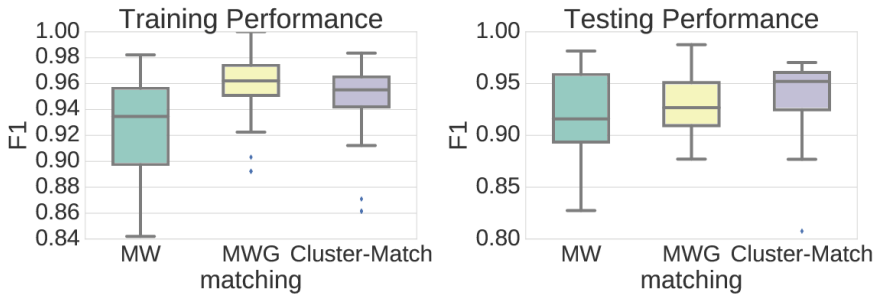


Figure 5.6: The best training and testing F_1-scores obtained from the alignment of 30 sets of pairwise Standard runs.

5.5.3 Probabilistic Matching Results from Cluster-Cluster

Direct-matching methods such as MW and Cluster-Match can only return a definite matching solution to the alignment problem (a peak from one run is either aligned to a peak in the other run, or not). In contrast, the second-stage clustering process of the IP clusters employed in the Cluster-Cluster method allows us to produce an estimate in the uncertainties of matching of peak features, producing as the alignment result a list of aligned peaksets that have been matched at varying levels of confidence. Figure 5.7 shows how a Precision-Recall (PR) curve, which shows how precision and recall change together, can be computed from the output of Cluster-Cluster on one of the sets of 4 randomly selected Standard runs and the set of 3 Beer runs. In Figure 5.7, the PR curves are plotted alongside the results from Cluster-Match at varying m/z and RT tolerance parameters (note that for Cluster-Cluster, we used only one set of potentially sub-optimal parameters for the second-stage clustering). Along both the PR curves on Figure 5.7, we see that generally, a decrease in the recall values is accompanied by an increase in the precision values. This applies to both the Standard and the Beer datasets, suggesting that by setting an appropriate threshold on the probabilities of aligned peaksets returned by Cluster-Cluster, we can obtain fewer aligned peaksets (lower recall) but at a higher confidence level of being correctly aligned (higher precision). In the face of further uncertainties with regard of user-defined parameters from the previous parts of the pipeline, the probabilistic alignment results returned by Cluster-Cluster allows the user to focus on peaksets of high matching confidence for subsequent analysis. This introduces the possibility of returning a smaller subset from the overall aligned peaksets that have a higher confidence score of being correctly aligned — an ability that few other matching methods can provide.

The results in Table 5.3 from running Cluster-Cluster, averaging over the sets of 2, 3, and 4 Standard runs, and on the entire 3 Beer runs, demonstrate that by setting some threshold values $\{0.30, 0.60, 0.90\}$ on the aligned peakset probabilities, various precision and recall values can be extracted. Upon aligning the sets of 4 Standard runs at threshold=0.30, Cluster-Cluster has a lower average precision of 0.81 than the best average performance

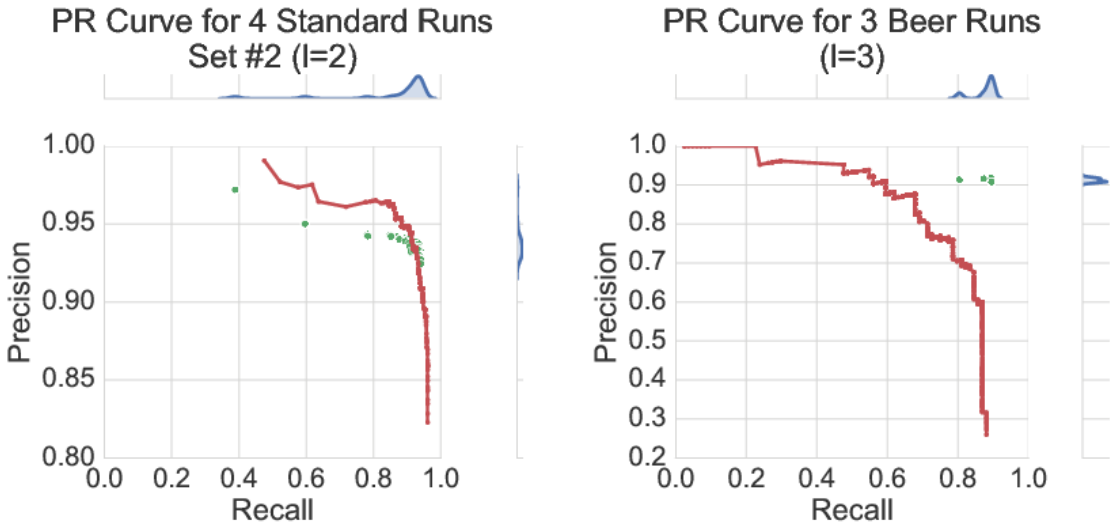


Figure 5.7: PR curves obtained from running Cluster-Cluster on one of the sets of 4 randomly selected Standard runs (left) and the 3 Beer runs (right). Green dots are performance points obtained from running Cluster-Match at varying m/z and RT tolerance parameters on the same datasets, with their distributions of the points plotted along the marginals. The same first-stage clustering results were used as input to both Cluster-Match and Cluster-Cluster.

from Cluster-Match at precision=0.87. Raising the threshold to 0.90 (consequently, decreasing recall as fewer aligned peaksets are returned) produces an average precision=0.90 for Cluster-Cluster, higher than 0.87 for Cluster-Match. On the complex Beer data, Cluster-Cluster produces precision=0.76 at threshold 0.30. Increasing the threshold to 0.90 produces a precision=0.94, which is again higher than the best attainable precision=0.91 from Cluster-Match. This demonstrates how recall can be traded for precision in Cluster-Cluster; a potentially useful ability in untargeted metabolomics experiments when the alignment ground truth is not available. In this situation, analysis effort can be focused on aligned peaksets with high confidence.

The importance of the adduct fingerprint term is shown in Table 5.3. Without the adduct fingerprint, a significantly lower F_1-score is produced by Cluster-Cluster at the probability threshold 0.90. This can be explained by the fact that excluding the adduct fingerprint term allows IP clusters having highly similar precursor mass and RT values to be placed in the same top-level cluster, despite each having an entirely different set of member adduct ions (and potentially corresponding to different metabolites). More false positive alignment items are produced, resulting in lower recall and F_1 scores. Using Figure 5.2 as an example, the aligned peakset consisting of the four $[M+H]^+$ peaks in Figure 5.2 has a high matching probability (0.97) when the adduct fingerprint term is used and almost never be placed together (near 0 probability) without. Similar observations can be concluded for the ions for other transformation types (e.g. the $[M+Na]^+$, $[M+ACN+H]^+$ adduct ions, etc.) shared

by the clusters in Figure 5.2. The inclusion of the adduct fingerprint term in Cluster-Cluster is necessary to ensure that well-calibrated probabilities on the alignment results are obtained, especially on aligned peaksets with higher matching confidence.

Table 5.3: Precision, recall and F_1 values from Cluster-Cluster for randomly selected sets of 2, 3 and 4 Standard runs (averaged) and the Beer runs for various l and thresholding levels $th = \{0.30, 0.60, 0.90\}$. Best results from Cluster-Match and the result of running Cluster-Cluster without the adduct fingerprint term are shown for comparison. Note that for Cluster-Cluster, the results come from using one set of potentially sub-optimal parameters for the second-stage clustering.

Dataset	l	Best Cluster-Match			Cluster-Cluster (CC)				CC (without adduct term)
		Avg. Prec.	Avg. Rec.	Avg. F_1	Threshold	Avg. Prec.	Avg. Rec	Avg. F_1	Avg. F_1
Standard	2	0.93	0.92	**0.93**	0.30	0.96	0.95	0.95	**0.95**
					0.60	0.98	0.93	**0.96**	0.93
					0.90	1.00	0.90	0.94	0.80
Standard	3	0.89	0.90	**0.89**	0.30	0.82	0.91	0.84	**0.86**
					0.60	0.86	0.88	**0.86**	0.85
					0.90	0.89	0.81	0.84	0.62
Standard	4	0.87	0.92	**0.89**	0.30	0.81	0.92	0.85	**0.89**
					0.60	0.84	0.89	0.85	0.86
					0.90	0.90	0.83	**0.86**	0.65
Beer 3 runs	3	0.92	0.89	**0.91**	0.30	0.76	0.77	**0.77**	**0.79**
					0.60	0.88	0.67	0.76	0.68
					0.90	0.94	0.54	0.68	0.63

5.5.4 Running time

Efficient inference is possible in PrecursorCluster as many peaks can only be placed in one cluster and need not be reassigned during Gibbs sampling. For the Standard runs with up to 5000 peak features, less than a quarter of peak features have to be reassigned. Taking 10000 posterior samples, Gibbs sampling for PrecursorCluster requires 20 minutes to process one Standard run on an Intel Core i5, 3.3GHz PC. Runs are processed independently and can be parallelized. In Cluster-Match, the matching of IP clusters via MW has a time complexity of $O(m \log n)$ time, where n and m are the number of vertices and edges in the bipartite graph to be solved, translating to a wall clock of less than a minute for each run. Cluster-Cluster requires longer computational time. With 1000 posterior samples per top-level bin, the processing of 2 Standard runs requires approximately half an hour. Each top-level bin can also be processed in parallel.

5.6 Conclusions

We have proposed an integrative workflow that performs the precursor clustering of ionization product peaks and uses that to improve alignment. The PrecursorCluster model intro-

duced is a data reduction process that can reduce the number of objects to be aligned based on a list of possible ionization transformation types. The clustering information extracted from PrecursorCluster can be used to improve other steps in the pipeline too. For instance, metabolite identification, currently the main bottlenecks in high-throughput metabolomics, might be improved through analyzing IP clusters as the objects of interest rather than individual peak features.

Through Cluster-Match, we have also demonstrated that IP clustering can be used to improve alignment, producing results that outperform the direct-matching of peak features alone. The proposed pairwise matching scheme used by Cluster-Match is frequently extended to the processing of multiple runs in a fairly ad-hoc manner and suffers from having to set a reference run (which can be considered another parameter to set). Producing a distance measure that works well for measuring similarities of peaks across multiple runs is non-trivial, and most methods do not take into account the uncertainties inherent in the matching of peak features across runs. Cluster-Cluster addresses these issues by not requiring a reference run and being able to return aligned peaksets at varying probabilities. As future work, PrecursorCluster can be improved by considering chromatographic shapes rather than RT values. An interactive visualisation module can be developed to let user visualize ionization product clustering and aligned peaksets (with their probabilities) from a single graphical interface. Such module can be incorporated as part of a larger metabolomics pipeline.

A weakness of the alignment methods described in this chapter is the fact that as a second-stage clustering step, both Cluster-Match and Cluster-Cluster requires the MAP assignment of peak features into their IP clusters. This means uncertainties in the ionization product clustering stage are not propagated to the matching stage. The next chapter addresses this problem by introducing a fully-hierarchical model that performs the clustering of peak features within run and across runs at once.

Chapter 6

Hierarchical Clustering of LC-MS Peaks

6.1 Introduction

The Cluster-Cluster method introduced in Chapter 5 performs the direct-matching of peak features in ionisation product (IP) clusters that themselves have been clustered together. However, peak features are assigned into IP clusters based on their maximum *a-posteriori* probabilities. In this chapter, we expand upon the idea of viewing direct matching as a fully hierarchical clustering problem by proposing **HDP-Align**, a Bayesian non-parametric model that groups related-peaks within runs and assigns them to global clusters shared across runs. Within each global cluster, peaks are further grouped by their m/z values into mass clusters, which represent the various ionisation products derived from the global compound. The proposed HDP-Align model allows us to infer the matching of peaks across all runs at once, without the need for any intermediate merging of pairwise runs, and the resulting posterior summaries provide us with a confidence score in the matching quality of aligned peaksets.

Similar to the Cluster-Cluster model introduced in Section 5.3.2, the proposed model of HDP-Align introduces the possibility of allowing the user to trade recall for precision from the alignment results by returning a smaller subset of the results having a higher confidence score of being correctly aligned. Figure 6.1 shows an illustration of the clustering process in HDP-Align. Additionally, the latent variables of clustering structure inferred in the model can potentially have physically meaningful identities that can be used for further analysis, and using a metabolomic dataset, we demonstrate the usefulness of such clustering objects by using the mass clusters derived from the model to perform putative annotations of features based on their potential adduct types and metabolite identities.

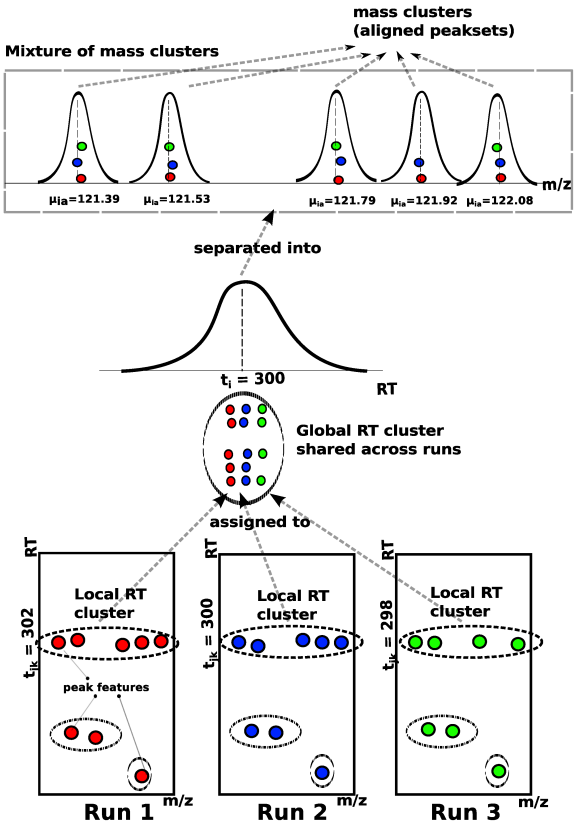


Figure 6.1: An illustrative example of how the proposed model in HDP-Align simultaneously **(1)** performs the clustering of related peak features into within-run local clusters by their RT values, **(2)** assigns the peak features to global RT clusters shared across runs, and **(3)** separates peak features into mass clusters, which correspond to aligned peaksets.

6.2 Related Work

The goal of establishing the matching of peaks across multiple runs at once can be viewed as a clustering problem, where a set of peaks can be grouped (by their m/z, RT and other suitable features) into local clusters within each run (representing all of the peaks from an individual compound), which are further grouped into global clusters shared across runs. A preliminary form of this idea has been explored in [70], where hierarchical clustering is performed on the total ion chromatogram data to group peaks into within-run local clusters, which are further grouped into across-run super clusters. The highly accurate mass information available from modern LC-MS platforms is not used in [70], although it is highlighted as a possible future work. The choice of using a hierarchical clustering method in [70] also requires choosing various user-defined parameters, such as determining a suitable cut-off for the dendogram produced, deciding on a suitable linkage method and defining an appropriate distance measure between groups of peaks.

6.3 Hierarchical Dirichlet Process Mixture Model for Alignment

6.3.1 Model Description

The proposed model for HDP-Align is framed as a Hierarchical Dirichlet Process (HDP) mixture model [71], described further in the background in Section 3.5. Essential modifications to the basic HDP model were performed to suit the nature of the multiple peak alignment problem. Our input consists of J input files, indexed by $j = 1, ..., J$, corresponding to the J LC-MS runs to be aligned. Each j-th input file contains N_j peak features in total, which can be separated into K_j local clusters of related-peak features. In a j-th file, peak features are indexed by $n = 1, ..., N_j$ and local clusters are indexed by $k = 1, ..., K_j$. Across all files, we assign each local cluster k in file j to a global cluster $i = 1, ..., I$, where I is the total number of global clusters, using the indicator variable v, as described in the following paragraph. A global cluster corresponds to the compound of interest during LC-MS analysis, e.g. metabolite or peptide fragment, that is present across runs, while local clusters are realisations of the global clusters in a specific run. Finally, within each global cluster i, we can further group peak features by their m/z values into A mass clusters (indexed by $a = 1, ..., A$) corresponding to the ions produced by different adduct-isotope combinations of the global compound during the MS process. To be specific, we call these peaks the ionisation products (IPs) from here on, rather than using the term 'related peaks' as in the previous chapter. Related/IP peaks have previously been elaborated further in Section 2.4.1.

We use the indicator variable $\boldsymbol{z}_{jnk} = 1$ to denote the assignment of peak n in file j to local cluster k in that file. Similarly, $\boldsymbol{v}_{jni} = 1$ if peak n in file j is assigned to global cluster i, and $\boldsymbol{v}_{jnia} = 1$ if peak n in file j is assigned to mass cluster a linked to metabolite i. Let d_j be the list of observed data of peak features in file j, $d_j = (\mathbf{d}_{j1}, \mathbf{d}_{j2}, ..., \mathbf{d}_{jn})$ where $\mathbf{d}_{jn} = (x_{jn}, y_{jn})$ with x_{jn} the RT value and y_{jn} the log m/z value of the peak feature. $\boldsymbol{\theta}$ denotes the global mixing proportions and $\boldsymbol{\pi}_j$ the local mixing proportions for file j. The global mixing proportions $\boldsymbol{\theta}$ are distributed according to the Griffiths, Engen and McCloskey (GEM) distribution:

$$\boldsymbol{\theta}|\alpha' \sim GEM(\alpha') \tag{6.1}$$

where the GEM distribution over $\boldsymbol{\theta}$ is described through the stick-breaking construction:

$$\beta_i \sim Beta(1, \alpha') \tag{6.2}$$

$$\theta_i = \beta_i \prod_{l=1}^{i-1}(1 - \beta_l) \tag{6.3}$$

The local mixing proportions $\boldsymbol{\pi}_j$ are distributed according to a Dirichlet Process (DP) prior with the base measure $\boldsymbol{\theta}$ and concentration parameter α_t.

$$\boldsymbol{\pi}_j|\alpha_t, \boldsymbol{\theta} \sim DP(\alpha_t, \boldsymbol{\theta}) \tag{6.4}$$

Within each file j, the indicator variable $\boldsymbol{z}_{jnk} = 1$ denotes the assignment of peak n in file j to local RT cluster k in that file. This follows the local mixing proportions for that file.

$$\boldsymbol{z}_{jnk} = 1|\boldsymbol{\pi}_j \sim \boldsymbol{\pi}_j \tag{6.5}$$

The RT value t_i of a global mixture component is drawn from a base Gaussian distribution with mean μ_0 and precision (inverse variance) σ_0.

$$t_i|\mu_0, \sigma_0 \sim \mathcal{N}(\mu_0, \sigma_0^{-1}) \tag{6.6}$$

The RT value t_{ij} of a local mixture component in file j is normally distributed with mean t_i and precision δ. The precision controls how much RT values of related-peak groups across runs are allowed to deviate from the parent global compound's RT.

$$t_{jk}|t_i, \delta \sim \mathcal{N}(t_i, \delta^{-1}) \tag{6.7}$$

Finally, the observed peak RT value is normally distributed with mean t_{jk} and precision γ. The precision controls how much RT values of peaks can deviate from their related-peak group.

$$x_{jn}|\boldsymbol{z}_{jnk} = 1, t_{jk}, \gamma \sim \mathcal{N}(t_{jk}, \gamma^{-1}) \tag{6.8}$$

The m/z value produced through high-precision MS instrument is highly accurate, and its correspondence is often preserved across runs. Once peaks have been assigned to their respective global clusters, we need to further separate peaks within each global cluster into mass clusters to obtain the actual alignment. These mass cluster corresponds to ionisation products. We do this by incorporating an internal DP mixture model on the m/z values (y_{jn}) within each global cluster i. Let the indicator $\boldsymbol{v}_{jnia} = 1$ denotes the assignment of peak n in file j to mass cluster a in the i-th global cluster. Then:

$$\begin{aligned}
\boldsymbol{\lambda}_i|\alpha_m &\sim GEM(\alpha_m) && (6.9)\\
\boldsymbol{v}_{jnia} = 1|\boldsymbol{\lambda}_i &\sim \boldsymbol{\lambda}_i && (6.10)\\
\mu_{ia}|\psi_0, \rho_0 &\sim \mathcal{N}(\mu_{ia}|\psi_0, \rho_0^{-1}) && (6.11)\\
y_{jn}|\boldsymbol{v}_{jnia} = 1, \mu_{ia} &\sim \mathcal{N}(\mu_{ia}, \rho^{-1}) \cdot I(\mathbf{d}_{jn}) && (6.12)
\end{aligned}$$

where the index ia refers to the a-th mass cluster of the i-th global cluster. $\boldsymbol{\lambda}_i$ is the mixing proportions of the i-th internal DP mixture for the masses, with α_m the concentration parameter. μ_{ia} is the mass cluster mean, drawn from the Gaussian base distribution with mean ψ_0 and precision ρ_0. The observed mass value is drawn from a Gaussian distribution with the component mean μ_{ia} and precision ρ, for which the value is set based on the MS instrument's resolution. Additionally, we add an additional constraint on the likelihood of y_{jn} using the indicator function $I(\cdot)$ such that $I(\mathbf{d}_{jn}) = 1$ if there are no other peaks inside the mass cluster that come from the same file as the current $\mathbf{d}_{jn}$ peak, and 0 otherwise. This constraint captures the restriction that a peak feature can only be matched to other peaks from different files, reflecting the assumption that within each LC-MS run, compounds produce ionisation products with distinct mass-to-charge fingerprints that can be used for matching to other runs.

6.3.2 Inference

Inference within the model is performed via a Gibbs sampling scheme, allowing us to compute posterior probabilities over the alignment of any set of peaks across the J files via the proportion of posterior samples in which they are assigned to the same mass component (a) in the same top-level cluster. In each iteration of the sampling procedure, we instantiate the mixture component parameters for the local RT cluster (t_{jk}) and global RT cluster (t_i) in the mixture model. In the internal DP mixture linked to each global cluster i, we marginalise out the mass cluster parameters (μ_{ia}). The initialisation step of the sampler is performed by assigning all peaks in each run into a single local RT cluster. Across runs, these local clusters are assigned under a global cluster shared across runs. Within a global cluster, peak features coming from different runs are assigned to a single mass cluster. The sampler than

iterates through each peak feature, removing it from the model, updating the assignment of peak features to clusters and performing the necessary book-keeping on any instantiated mixture components. Further details on the specific Gibbs update statements can be found in following sections.

Updating peak assignments

We use the following variables to denote the count of items in any clustering object: c_{jk} is the number of peaks in a local cluster k of file j. c_i is the number of local clusters in a global cluster i, and c_{ia} is the number of peaks in a mass cluster a inside a global RT cluster i. To update the assignment of a peak $\mathbf{d}_{jn}$ to local RT cluster k during Gibbs sampling, we need the conditional probability of $P(\boldsymbol{z}_{jnk} = 1)$ given every other parameters, denoted as $P(\boldsymbol{z}_{jnk} = 1|\mathbf{d}_{jn}, \ldots)$.

$$P(\boldsymbol{z}_{jnk} = 1|\mathbf{d}_{jn}, \ldots) \propto \begin{cases} c_{jk} \cdot p(\mathbf{d}_{jn}|\boldsymbol{z}_{jnk} = 1, ...) \\ \alpha_t \cdot p(\mathbf{d}_{jn}|\boldsymbol{z}_{jnk^*} = 1, ...) \end{cases} \tag{6.13}$$

We will consider the top and bottom terms of eq. 6.13 separately in the following.

1. The likelihood of the peak $\mathbf{d}_{jn}$ to be in an existing local RT cluster k, $p(\mathbf{d}_{jn}|\boldsymbol{z}_{jnk} = 1, ...)$ is proportional to c_{jk}. This is assumed to factorise across the RT (x_{jn}) and mass (y_{jn}) terms

$$p(\mathbf{d}_{jn}|\boldsymbol{z}_{jnk} = 1, ...) = p(x_{jn}|\boldsymbol{z}_{jnk} = 1, ...) \cdot p(y_{jn}|\boldsymbol{z}_{jnk} = 1, ...) \tag{6.14}$$

The RT term $p(x_{jn}|\boldsymbol{z}_{jnk} = 1, ...)$ in eq. 6.14 is normally distributed with mean t_{jk} and precision γ, while the mass term $p(y_{jn}|\boldsymbol{z}_{jnk} = 1, ...)$ is an internal DP mixture of mass components linked to the parent global cluster i of an existing local cluster k. We then marginalise over all mass clusters in i to get $p(y_{jn}|\boldsymbol{z}_{jnk} = 1, \boldsymbol{v}_{jni} = 1...)$

$$\begin{aligned} p(y_{jn}|\boldsymbol{z}_{jnk} = 1, \boldsymbol{v}_{jni} = 1...) = &\sum_a \frac{c_{ia}}{\alpha_m + \sum_a c_{ia}} p(y_{jn}|\boldsymbol{v}_{jnia} = 1, ...) \\ &+ \frac{\alpha_m}{\alpha_m + \sum_a c_{ia}} p(y_{jn}|\boldsymbol{v}_{jnia^*} = 1, ...) \end{aligned} \tag{6.15}$$

To compute the terms in eq. 6.15, first we consider the case for an existing mass cluster a in the global RT cluster i. Then,

$$p(y_{jn}|\boldsymbol{v}_{jnia} = 1, ...) = \mathcal{N}(\mu_{ia}, \rho^{-1}) \tag{6.16}$$

For a new mass cluster a^* in the global RT cluster i, we marginalise out μ_{ia} to obtain

$$p(y_{jn}|\boldsymbol{v}_{jnia^*} = 1, ...) = \mathcal{N}(\psi_0, \rho^{-1} + \rho_0^{-1}) \tag{6.17}$$

2. The likelihood of the peak $\mathbf{d}_{jn}$ to be in a new local cluster k^* is proportional to α_t. Marginalising over all global clusters, we get

$$p(\mathbf{d}_{jn}|\boldsymbol{z}_{jnk^*}=1,...) = \sum_i \left[\frac{c_i}{\alpha' + \sum_i c_i} p(\mathbf{d}_{jn}|\boldsymbol{v}_{jni}=1,...)\right] + \frac{\alpha'}{\alpha' + \sum_i c_i} p(\mathbf{d}_{jn}|\boldsymbol{v}_{jni^*}=1,...) \tag{6.18}$$

There are two terms to compute in eq. 6.18: whether peak $\mathbf{d}_{jn}$ is in an existing global cluster i or a new global cluster i^*. For an existing global RT cluster i in eq. 6.18, $p(\mathbf{d}_{jn}|\boldsymbol{v}_{jni}=1,...)$ is assumed to factorise into its RT and mass terms, so $p(\mathbf{d}_{jn}|\boldsymbol{v}_{jni}=1,...) = p(x_{jn}|\boldsymbol{v}_{jni}=1,...)\cdot p(y_{jn}|\boldsymbol{v}_{jni}=1,...)$. We marginalise over all local RT clusters to obtain

$$p(x_{jn}|\boldsymbol{v}_{jni}=1,...) = \mathcal{N}(x_{jn}|t_i, \gamma^{-1}+\delta^{-1}) \tag{6.19}$$

and marginalise over all possible mass clusters in the internal DP linked to global cluster i to obtain $p(y_{jn}|\boldsymbol{v}_{jni}=1,...)$. This is defined in eq. 6.15). Finally, for a new global RT cluster i^* in eq. 6.18, $p(\mathbf{d}_{jn}|\boldsymbol{v}_{jni^*}=1,...)$ is also assumed to factorise into its RT and mass terms. Then, we marginalise over t_{jk} and t_i to obtain

$$p(x_{jn}|\boldsymbol{v}_{jni^*}=1,...) = \mathcal{N}(x_{jn}|\mu_0, \sigma_0^{-1}+\gamma^{-1}+\delta^{-1}) \tag{6.20}$$

and marginalise over μ_{ia} to get

$$p(y_{jn}|\boldsymbol{v}_{jni^*}=1,...) = \mathcal{N}(y_{jn}|\psi_0, \rho^{-1}+\rho_0^{-1}) \tag{6.21}$$

Updating instantiated variables

The following expressions are used to update the instantiated mixture component parameters in the model during Gibbs sampling.

1. Updating global cluster's RT t_i: here, $t_{jk\in i}$ refers only to local RT clusters currently assigned to the global cluster i, and c_i is the count of such peaks. Then

$$p(t_i|\ldots) \propto p(t_i|\mu_0, \sigma_0^{-1}) \prod_j^J \prod_k^K p(t_{jk\in i}|t_i, \delta) = \mathcal{N}(\mu_i, \gamma_i^{-1}) \tag{6.22}$$

where $\mu_i = \frac{1}{\gamma_i}\left[\mu_0\sigma_0 + \delta\sum_j\sum_k t_{jk\in i}\right]$ and $\gamma_i = \sigma_0 + \delta c_i$.

2. Updating local cluster's RT t_{jk}: here, $x_{jn\in k}$ refers only to peaks currently assigned to the local RT cluster k, and c_{jk} is the count of such peaks.

$$p(t_{jk}|\ldots) \propto p(t_{jk}|t_i, \delta^{-1}) \prod_j^J \prod_n^N p(x_{jn\in k}|t_{jk}, \gamma) = \mathcal{N}(\mu_k, \gamma_k^{-1}) \quad (6.23)$$

where $\mu_k = \frac{1}{\gamma_k}\left[t_i\delta + \gamma \sum_j \sum_n x_{jn\in k}\right]$ and $\gamma_k = \delta + \gamma c_{jk}$.

6.3.3 Using the Inference Results

Feature Matching

The Gibbs sampling procedure produces a collection of samples from the posterior distribution over all parameters of the HDP-Align model. We can use these samples to compute various posterior summaries and more interestingly, extract the alignment of peaks from the inference results (since features assigned into the same mass cluster with the same global RT cluster are considered to be aligned). For each sample from the posterior distribution, we record the aligned peaksets of peak features put into the same mass cluster. Averaging over all samples provides a distribution over these aligned peaksets.

Note that across the returned aligned peaksets, it is possible for the same peak to be matched to different partners with varying probabilities, depending on how often they co-occur together in the same mass cluster. To allow the possibility of controlling precision and recall from the results, we provide another user-defined threshold t, where peak feature combinations are included in the output from the model only when they occur with matching probability $>t$. Varying this threshold allows user to trade precision for recall: a low value for t gives a larger set of results that are potentially less precise, while conversely a high t provides a smaller, more precise set of aligned peaksets. This is an output not available from other alignment methods and can potentially be useful in problem domains where high precision is required from the alignment results.

Isotopic Product and Metabolite Identity Annotations

As described in Section 2.4.1, in metabolomics studies using electrospray ionisation, a single metabolite can generate multiple ionisation products peaks, (such as isotopic variants, adducts, fragment peaks), alongside other peaks resulting from noise and artifacts introduced during mass spectrometry [2]. Determining and annotating these IP peaks are desirable to remove extraneous peaks and reduce the burden of subsequent downstream analysis. Additionally, deducing the precursor molecular masses that generate the IPs is often essential in order to query compound library databases before assigning putative metabolite identities.

The resulting clustering objects inferred from HDP-Align lend themselves to further analysis in a natural fashion, as global RT clusters in HDP-Align may correspond to metabolites, while local RT clusters may correspond to the noisy realisations of these metabolites within each run. Mass clusters in the internal mixture of each global cluster could correspond to the IPs. To demonstrate the possibility of obtaining additional information beyond alignment from the output of HDP-Align, we follow the workflow in [2] that performs IPs and metabolite annotations of peak features. This workflow is composed of multiple key steps: peak matching, ionisation product clustering and metabolite mass matching. A key difference of HDP-Align to the workflow in [2] lies in the fact that HDP-Align is able to perform the two separate steps of peak alignment and potential IP clustering simultaneously, as shown in Figure 6.2.

Given the set of potential IP clusters, we can perform IP annotation on the peaks. To do this using the metabolomic dataset, first we take the set of clustering objects produced in a single posterior sample. For each mass cluster, we assign its m/z value to be the average m/z values of features assigned to it, denoted by m. A list of common adducts (Table 4.3) in positive ionisation mode is used to compute the inverse transformation $t^{-1}(m, d, e, u) = ((e * m) - d)/u$ for a precursor mass c that generates m. Here, d is the adduct mass, e is the charge and u the number of metabolite molecules in the IP type. Following [2], any two mass clusters sharing the same precursor mass c (within tolerance) provide a vote on the presence of that consensus precursor mass. The respective pair of mass clusters and features within can then be annotated with the adduct type that produces the transformation t^{-1} to the shared precursor mass c. The set of precursor masses deduced in this manner can also be used to query KEGG (a database of metabolite compounds) in order to assign putative identities to global compounds.

6.4 Evaluation Study

6.4.1 Evaluation Datasets

Performance of the proposed methods and other benchmark methods is evaluated on the LC-MS datasets of proteomic, glycomic and metabolomic experiments first introduced in Section 4.6. As before, all 6 fractions from the P1 Proteomic dataset in [23] are used. Each fraction contains 2 runs of features having high RT variations across runs are used in our experiments. Unlike Section **??** where only pairs of runs used, here we use the first 10 runs of the Glycomic dataset provided by [1] for our multiple-runs experiment. Additionally, the Standard metabolomic dataset, first introduced in Section **??**, is also used. Here, we selected 6 runs for our experiment. Table 6.1 summarises the different evaluation datasets and the

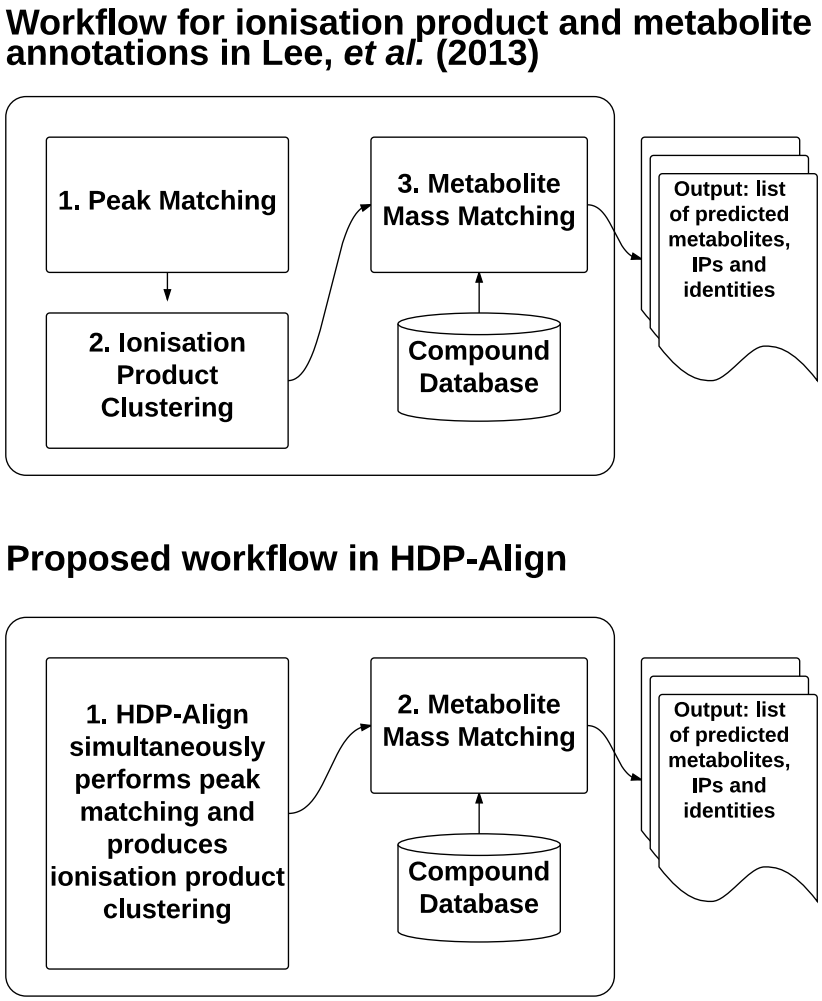


Figure 6.2: Comparisons on the workflow to assign putative annotations on isotopic products and metabolites described in [2] and in HDP-Align.

number of features each has.

Dataset	No. runs	Total Features
P1 Frac 000	2	10606
P1 Frac 020	2	2135
P1 Frac 040	2	2188
P1 Frac 060	2	3342
P1 Frac 080	2	2086
P1 Frac 100	2	1326
Glycomic	10	9344
Metabolomic	6	7477

Table 6.1: Total number of runs and features of the selected evaluation datasets.

6.4.2 Performance Measures

While a definition of precision and recall in the context of alignment performance has been proposed and used in Chapter 4, the performance measures defined there applies only to pairwise alignment, i.e. an aligned peakset can only consist of two matched peak features, at most. Here, we propose a generalisation of the performance measures defined in Section 4.6.4 to apply to the alignment of multiple runs.

To provide a definition of 'precision' and 'recall' suitable for evaluating alignment performance of multiple runs, we first enumerate all the possible q-size combinations for every aligned peakset in both the method's output and the ground truth list. For example, an alignment method returns a list of two aligned peaksets $\{a, b, c, d, \}, \{e, f, g\}$ as output. When $q = 2$, this output can be enumerated into a list of 9 'alignment items' of all the pairwise combinations of features: $\{a, b\}, \{a, c\}, \{a, d\}, \{b, c\}, \{b, d\}, \{c, d\}, \{e, f\}, \{e, g\}$, $\{f, g\}$. Let M and G be the results from such enumeration from a method's output and the ground truth respectively. Each distinct combination of features in M and G can be considered as an item during performance evaluation. Intuitively, the choice of q reflects the strictness of what is considered to be a true positive item, with larger values of q demanding an alignment method that produces results spanning more runs correctly.

For a given q, the following positive and negative instances of alignment item can now be defined for the purpose of performance evaluation:

- True Positive ($\boldsymbol{TP}$): items that should be aligned (present in G) and are aligned (present in M).
- False Positive ($\boldsymbol{FP}$): items that should not be aligned (absent from G) but are aligned (present in M).

- True Negative ($\boldsymbol{TN}$): items that should not be aligned (absent from G) and are not aligned (absent from M).
- False Negative ($\boldsymbol{FN}$): items that should be aligned (present in G) but are not aligned (absent from M).

In the context of alignment performance, precision ($\frac{\boldsymbol{TP}}{\boldsymbol{TP+FP}}$) is therefore the fraction of items in M that are correct with respect to G, while recall ($\frac{\boldsymbol{TP}}{\boldsymbol{TP+FN}}$) is the fraction of items in G that are aligned in M. A method with a perfect alignment output would have both precision and recall values of 1.0.

6.4.3 Benchmarking Method

Following Chapter 4, we benchmark HDP-Align against two established alignment methods: SIMA [27] and MZmine2's Join Aligner [24]. The selection of SIMA and Join as the benchmark methods is motivated by the fact that both methods are direct matching methods (thus easily comparable to HDP-Align) but still differ sufficiently in how they establish the final alignment results, in particular when it comes to the alignment of multiple runs. This is primarily due to the differences between both methods in the form of the distance/similarity function between peak features, the actual matching algorithm itself and the merging order of pairwise results to construct the full alignment results.

The two most important parameters to configure in both methods are the mass and RT tolerance parameters, used for thresholding and computing feature similarities during matching. We label these common parameters as the $T_{(m/z)}$ and T_{rt} parameters. Note that despite the common label, each method may use the parameter values differently during the alignment process. In our experiments, we let $T_{(m/z)}$ and T_{rt} vary within reasonable ranges (details in Section 6.4.4) and report all performance values generated by each combination of the two parameters.

6.4.4 Parameter Optimisations

Tables 6.2 and 6.3 describe the parameter ranges of each method during performance evaluation. For HDP-Align (Table 6.2), we perform the experiments based on our initial choices on the appropriate parameter values. These are almost certainly less than optimal and can be optimised further. The mass cluster standard deviation $\sqrt{\rho^{-1}}$ for HDP-ALign is set to the equivalent value in parts-per-million (ppm). These are 500 ppm for the Proteomic dataset and 3 ppm for the Glycomic and Metabolomic datasets. The local (within-run) cluster RT standard deviation $\sqrt{\gamma^{-1}}$ is assumed to be fairly constant and set to 2 seconds for all datasets,

while the global cluster standard deviation $\sqrt{\delta^{-1}}$ is set in the following dataset-specific manner: 50 seconds for the Proteomic dataset and 20 seconds for the remaining datasets. The larger standard deviation value is required for the Proteomic dataset to accomodate for greater RT drifts across runs. Other hyperparameters in HDP-Align are fixed to the following values: $\alpha' = 10$, $\alpha_t = 10$, $\alpha_m = 100$. The values of the precision hyperparameters for global cluster RT (σ_0) and mass cluster (ρ_0) are set to a broad value of 1/5E6. No significant changes were found to the results when these hyperparameters for the DP concentrations and cluster precisions were varied. The mean hyperparameters μ_0 and ψ_0 are set to the means of the RT and m/z values of the input data respectively. During inference for the Glycomic and Metabolomic datasets, 500 posterior samples were collected for the Gibbs sampling procedure, discarding the first 500 during the burn-in period. For the Proteomic dataset with larger RT deviations, 5000 posterior samples were obtained after discarding the first 5000 samples during burn-in. The number of samples is selected to ensure convergence during inference.

Dataset	HDP
P1 Frac 000	$\sqrt{\rho^{-1}} = 500$ ppm, $\sqrt{\gamma^{-1}} = 2$ s, $\sqrt{\delta^{-1}} = 50$ s
P1 Frac 020	
P1 Frac 040	
P1 Frac 060	
P1 Frac 080	
P1 Frac 100	
Glycomic	$\sqrt{\rho^{-1}} = 3$ ppm, $\sqrt{\gamma^{-1}} = 2$ s, $\sqrt{\delta^{-1}} = 20$ s
Metabolomic	$\sqrt{\rho^{-1}} = 3$ ppm, $\sqrt{\gamma^{-1}} = 2$ s, $\sqrt{\delta^{-1}} = 20$ s

Table 6.2: Parameters used for HDP-Align

For SIMA and Join, we report the results from all combinations of the mass and RT tolerance parameters within reasonable ranges listed in Table 6.3. This follows from the range of parameters selected for evaluation experiments in the previous Chapter 4. The ranges of $T_{(m/z)}$ and T_{rt} parameters used are based values reported on [23] for the Proteomic dataset and [1] for the Glycomic dataset. For the Metabolomic dataset, they were chosen in light of the mass accuracy and RT deviations of the data.

6.5 Results

Precision and recall values for the evaluated methods methods on the different datasets are shown in Sections 6.5.1 and 6.5.2. Additionally, an example of the further annotations for the putative adduct type and metabolite identity that can be produced by HDP-Align is also shown in Section 6.5.2. Running time of the evaluated methods are reported in Section 6.5.3.

Dataset	Benchmark (SIMA, Join)
P1 Frac 000	$T_{(m/z)} = \{1.0, 1.1, ..., 2.0\}$, $T_{rt} = \{10, 20, ..., 180\}$ s
P1 Frac 020	
P1 Frac 040	
P1 Frac 060	
P1 Frac 080	
P1 Frac 100	
Glycomic	$T_{(m/z)} = \{0.05, 0.1, 0.25\}$,$T_{rt} = \{5, 10, ..., 120\}$ s
Metabolomic	$T_{(m/z)} = \{0.001, 0.01, 0.1\}$, $T_{rt} = \{5, 10, ..., 120\}$ s

Table 6.3: Parameters used for the benchmark methods (SIMA, Join).

6.5.1 Proteomic (P1) Results

Figure 6.3 shows the results from performance evaluation on the Proteomic (P1) dataset. We see that both benchmark methods (SIMA and Join) produce a wide range of performance depending on the parameter values for $(T_{(m/z)}, T_{rt})$ chosen. Sensitivity to parameter values is expected on this dataset due to the low mass accuracy in the MS instrument that produces the data and the high RT drifts present across runs (further details in [23]). HDP-Align performs well on several fractions (particularly fractions 040, 060, 080, 100) with precision-recall performance close to the optimal performance attainable by the benchmark methods. On all fractions, HDP-Align is also able to produce higher-precision results compared to the benchmark methods by reducing recall through setting the appropriate values for the threshold t. The primary benefits of quantifying alignment uncertainties is realised here as the well-calibrated probability scores on the matching confidence of aligned peak features produced HDP-Align allows the user to choose which point along the PR curve to operate on. It is less obvious how this can be accomplished in the benchmark methods by varying the RT (T_{rt}) and m/z ($T_{m/z}$) thresholding parameters, if at all possible.

6.5.2 Glycomic and Metabolomic Results

Figures 6.4 and 6.5 show the results from experiments on the Glycomic and Metabolomic datasets. Similar to the Proteomic dataset, a wide range of precision-recall values can be observed in the results for the benchmark methods on the two datasets. The performance of HDP-Align, using the same set of parameters on both datasets, come close to the optimal results from the benchmark methods, while still allowing the user to control the desired point along the precision-recall curve to operate on.

The results for the Glycomic dataset (Figure 6.4) also show some additional results on how the measured precision-recall values might change depending on the strictness of what constitutes an 'item' during performance evaluation. This is accomplished by gradually in-

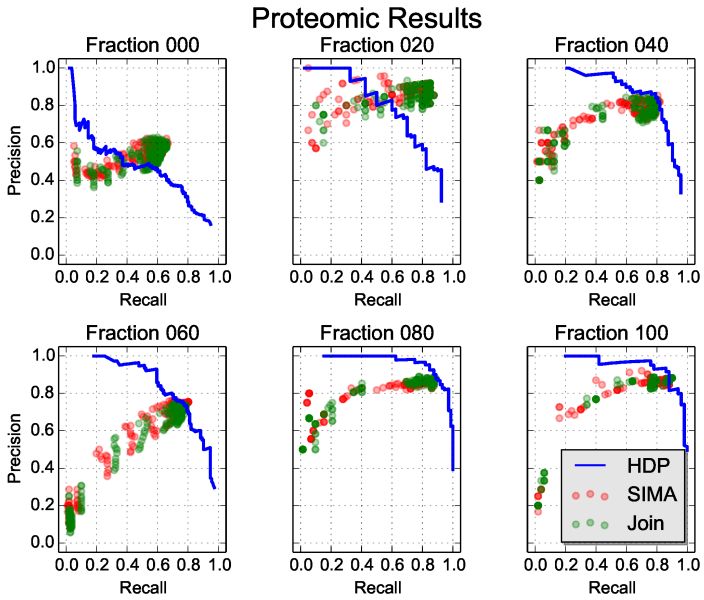


Figure 6.3: Precision-recall values on the different fractions of the Proteomic (P1) dataset.

creasing the value for q (described in detail in Section 6.4.2) that determines the size of the feature combinations enumerated from a method's output. For example, q=2 considers all pairwise combinations of features from the method's output during performance evaluation, while $q = 4$ considers all combinations of size 4, and so on. Figure 6.4 shows that as q is increased, parameter sensitivity seems to become more of an issue for the benchmark methods, with more parameter sets having lower precisions in the results. Across all qs evaluated, parameter pairs that produce the best alignment performance (points with high precision and recall values) are generally small $T_{(m/z)}$ and large T_{rt} values. Examples of parameter pairs that produce the best and worse performance for SIMA are shown in Figure 6.5. The results here appear to suggest the importance of having high mass precision during matching. Importantly, we see from Figure 6.4 that the performance of HDP-Align remains fairly consistent as q is increased.

The Metabolomic dataset also provides us with additional results in form of annotations of putative adduct type and metabolite identities. A thorough evaluation on the quality of such annotations, in comparison to e.g. the workflow proposed in [2], is beyond the scope of this chapter and would likely necessitate using a different and more appropriate evaluation dataset. Instead, we present an example of the further analysis performed by HDP-Align (as proposed in Section 6.3.3) on the resulting clustering objects after inference. Figure 6.6 shows a global RT cluster where peak features across runs have been grouped by their RT and m/z values. Within this global cluster, peak features are further separated into 6 mass clusters – corresponding to ionisation products produced by the global cluster during mass spectometry. In Figure 6.6, mass cluster A and B contain features aligned from several runs but they do not have any other mass cluster sharing a possible precursor mass. Mass

cluster C and D share a common precursor mass (292.12696) and can thus be annotated by the adduct type that produce the transformation. Similarly, mass cluster E and F share a common precursor mass at 383.14278. Queries to a local KEGG database are issued based on the precursor mass values, producing several compound identities that can be putatively assigned to the global RT cluster. It is a great strength of our approach that this putative identification step appears very naturally from the alignment results.

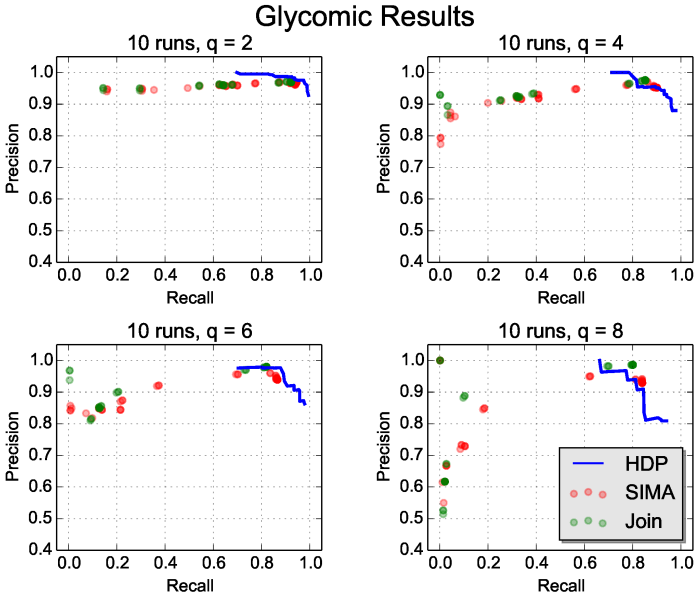


Figure 6.4: Precision-recall values on the alignment of 10 runs from the Glycomic dataset when q (the strictness of performance evaluation as described in Section 6.4.2) is gradually increased.

6.5.3 Running Time

The main factor affecting the running time of HDP-Align is the total number of peaks across all runs to be processed and the number of samples produced during Gibbs sampling. In each iteration of Gibbs sampling, HDP-Align removes a peak from the model, updates parameters of the model conditioned on every other parameters, and reassigns a peak into RT and mass clusters. The time complexity of this operation is $O(N)$, where N is the total number of peaks across all runs. In practice, additional time will also be spent on various necessary book-keeping operations, such as deleting empty clusters that are no longer required, updating internal data structures, etc. A representative running time is given as $N = 9344$ for the Glycomic dataset. HDP-Align requires approximately 5 hours to collect 1000 samples. In comparison, both SIMA and Join perform alignment within 5 to 10 seconds. Similarly, for $N = 7477$ for the Metabolomic dataset, HDP-Align produces the results in approximately 4 hours after collecting 1000 samples, while SIMA and Join complete

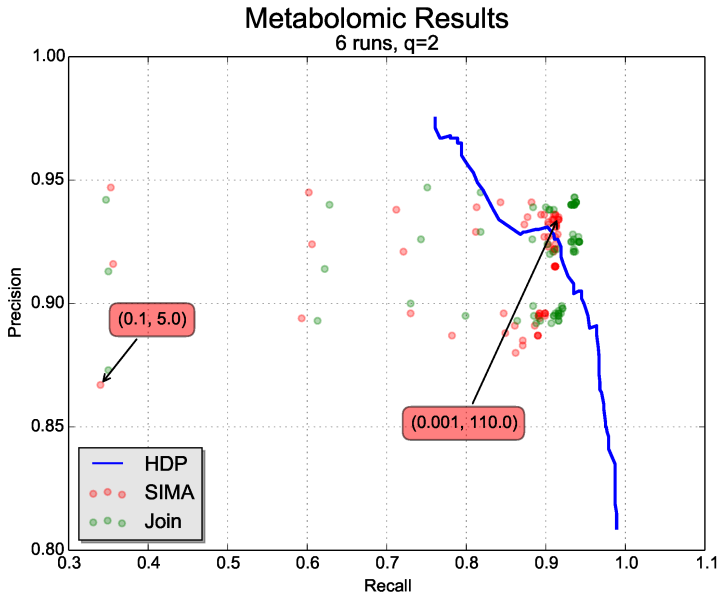


Figure 6.5: Precision-recall values on the alignment of 6 runs from the Metabolomic dataset. The parameter values $(T_{m/z}, T_{rt})$ that produce the best and worst performance in SIMA are also annotated in the Figure (red boxes).

within seconds. The running time of HDP-Align, while being significantly longer than these two benchmark methods, is comparable to other computationally-intensive steps (e.g. peak detection) in a typical LC-MS pipeline.

6.6 Discussion and Conclusion

We have presented a hierarchical non-parametric Bayesian model that performs direct matching of peak features, a problem of significant importance in the data pre-processing pipeline of large untargeted LC-MS datasets. Unlike other direct matching methods, the novelty of our proposed approach lies in its ability of to produce well-calibrated probability scores on the matching confidence of aligned peak features (evidenced by the increasing precision and decreasing recall as the threshold t is increased). This is accomplished by casting the multiple alignment problem of LC-MS peak features as a hierarchical clustering problem. Matching confidence can then be obtained based on the probabilities of co-eluting peak features to be assigned under the same mass component of the same global cluster. Experiments based on datasets from real proteomic, glycomic and metabolomic experiments show that HDP-Align is able to produce alignment results competitive to the benchmark alignment methods, with the added benefit of being able to provide a measure of confidence in the alignment quality. This can be useful in real analytical situations, where neither the optimal parameters nor the alignment ground truth is known to the user.

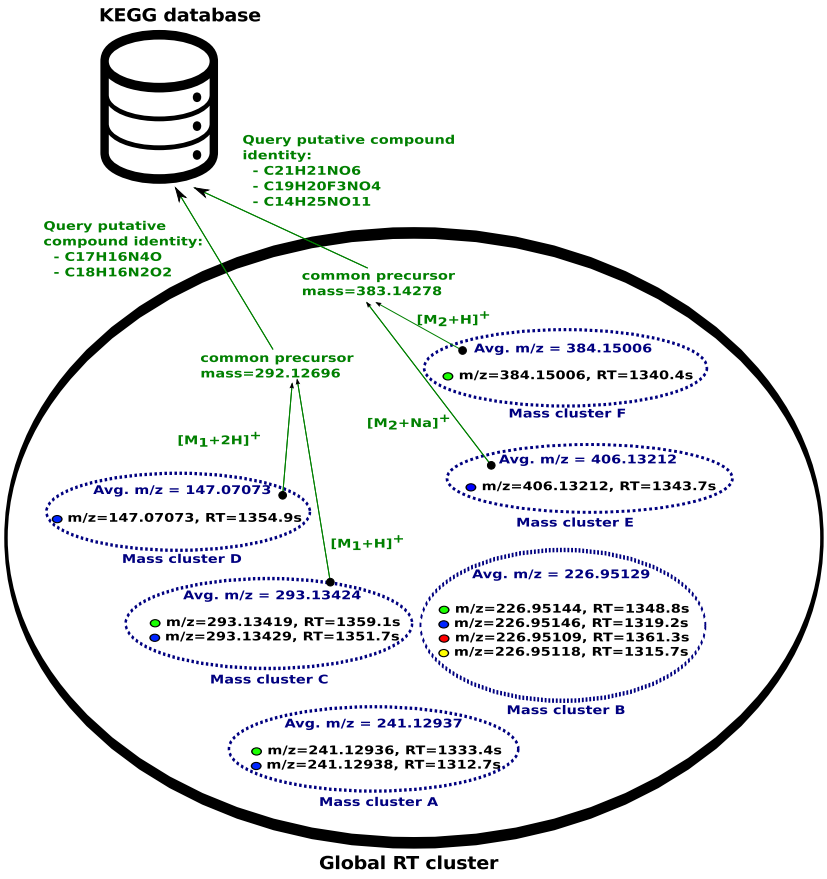


Figure 6.6: Example analysis that can be performed on the clustering objects inferred in a Gibbs sample from HDP-Align. The outer black oval denotes a global RT cluster (generally corresponding to a metabolite compound), while the smaller dotted ovals within denote mass clusters (labelled as mass cluster A, B, C, D, E, F). Peak features are denoted by the filled circle, with the fill colour indicating the originating run of a peak. Green colours denote additional analysis steps that can be performed on the mass cluster objects. **REDRAW TO LOOK NICER?**

Through comparisons against benchmark methods, our studies have also investigated the effect of sub-optimal parameter choices on alignment performance. While beyond the scope of our paper, we agree with [18, 53] that thorough investigations into the influence of numerous configurable parameters (prevalent in nearly all LC-MS data processing pipeline) on the resulting biological conclusions are of utmost importance. This should be followed by the development of methods to minimise or automatically-tune such configurable parameters. Despite the abundance of new methods proposed for LC-MS data pre-processing, relatively few studies have been done on the subject of quantifying uncertainties and alleviating the burden of parameter optimisations during actual data analysis. One way to minimise the number of parameters is through the integration of multiple steps in the typical LC-MS pipeline into fewer steps. Our proposed model in HDP-Align can potentially be extended in this manner, as evidenced by the metabolomic dataset results where we directly use the clustering objects inferred from the model to perform further analysis on putative adduct and metabolite type annotations. While the proposed annotation approach in Section 6.3.3 is fairly simple, it can be easily extended to more sophisticated annotation strategies, such as in CAMERA [39].

A primary weakness of HDP-Align lies in the long computational time required to produce results. Additional work will be required to reduce the computational burden of the model through various optimisation tricks and potentially by parallelising the Gibbs inference step using e.g. the method described in [**?**]. Another possibility is to adopt a different non-sampling-based inferential approach or perhaps even a simpler model altogether, while still retaining the essence and benefits of the HDP-Align model. The key insight here lies in modelling related peaks as within-file clusters in a single run but also allowing these within-file clusters to be generated by globally-shared clusters spanning across multiple runs. The results presented in the current chapter suggest the method shows enough promise to warrant the effort to speed it up, and indeed that is what we will discuss in the next chapter.

Another aspect worthy of investigation is determining the most effective way to present and visualise the alignment probabilities produced by HDP-Align. Additional sources of information present in the LC-MS data, such as chromatographic peak shapes, can also be used to improve alignment performance and subsequent analyses that follow.

Finally, replacing or enhancing the mixture of mass components used in HDP-Align with a more appropriate mass model, such as that in MetAssign [40] that specifically takes into account the inter-dependency structure of peaks, is an avenue for future work. This will be particularly useful when extending the proposed model in HDP-Align into a single inferential model that encompasses many intermediate steps in a typical LC-MS data processing pipeline.

Chapter 7

Substructure Discovery in Tandem Mass Spectometry Data

7.1 Introduction

As the results from Chapter 5 shows, the ionization product (IP) types of many observed peaks are often unknown and therefore the molecular mass of metabolites that generate these peaks are also unknown. This makes identification difficult as mass is often a required information when querying metabolite identities against publicly-available databases, such as KEGG [72] and PubChem [73]. In addition, while modern mass spectrometry instruments can be highly accurate up to 3 parts-per-million (ppm), even a mass accuracy of 1 ppm is not sufficient to reliably determine the elemental composition (formula) of a metabolite [74] during database queries. The presence of isomers (metabolites having the same formula and mass but are structurally different from each other) suggests that relying on mass alone, the same peak might be incorrectly matched to multiple isomeric metabolites. Retention time (RT) may help to distinguish certain isomers that have different elution profiles, but RT drift, a main challenge in alignment, means observed RT values can vary across different chromatographic platforms and cannot be easily used as a characteristic information in public databases during identification. Apart from a small number of metabolites present in a standard solution that can be identified with a high degree of confidence (as they produce measured peaks having reliably known m/z and RT values), information on the mass and RT values alone are not enough to establish the identity of many metabolites in untargeted studies.

Fragmentation spectra are the results of chaining two stages of mass spetrometry steps. In data-dependent acquisition, a precursor or parent (MS1) peak is selected according to a certain criteria, frequently the top-N most intense peaks in a scan, for further fragmentation. This produces for each fragmented parent peak a distinct pattern of fragment (MS2) peaks.

Fragmentation patterns can be used to aid identification through the matching of a query spectrum to a database of reference spectra. In recent years, a growing number of fragmentation spectra databases have been made public, including METLIN [75], ChemSpider [76] and MassBank [77]. However, mass spectral databases are not comprehensive and contain only a small number of known metabolites. The large variance in submitted spectra further limits potential matches as sensible results can only be obtained when matching spectra generated from measurement platforms having similar characteristics (for e.g., produced through the same ionization method under a similar mass accuracy). According to [78], approximately 2% of spectra in an untargeted metabolomics experiment can be matched and subsequently identified – a small number in contrast of the vast collection of metabolites that comprise the metabolic pathways of an organism.

Multiple metabolites can share the same chemical substructure. For example, carboxylic acid (Figure 7.1) is a generic substructure shared by many amino acids and organic acids. During fragmentation, a characteristic fragment peak 46 Da away from the parent peak — representing the combined loss of CO and H2O due to the breaking of the neutral carboxyl group (COOH) from the molecular ion — can be expected to occur in the spectra of metabolites sharing carboxylic acid as a substructure. Knowledge of the constituent substructures that comprise a metabolite, particularly of the larger and more specific substructures, can also be used to provide a hint as to the overall identity of the metabolite. Classification method, such as Support Vector Machine, decision tree and neural networks [79, 80, 81, 82], have been trained to learn spectral features that represent substructures and predict the presence or absence of substructures from fragmentation spectra. Combined with information from the parent peak (such as the m/z, RT values and IP types if available), this provides additional information that can aid in the identification of metabolites that cannot be resolved through the traditional method of spectral database matching alone.

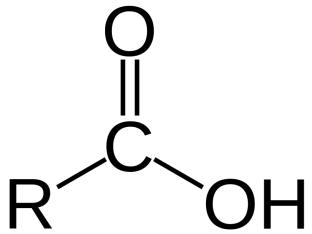


Figure 7.1: The carboxylic acid substructure. In the diagram, R refers to the residue, which is the rest of the metabolite attached to this substructure.

A common shortcoming of these classification approaches higlighted before is the need of the supervised training of the classifier (classification-based approaches may fail to generalise well to new dataset produced from different analytical platforms). Based on the assumption that fragmentation spectra contain fragment peaks that represent shared substructures

of metabolites, we propose a workflow that applies the Latent Dirichlet Allocation (LDA) model to spectral fragmentation data. The proposed workflow produces the decomposition of fragmentation spectra (equivalently a document in standard LDA) into the set of *Mass2Motifs* (equivalently a topic in standard LDA). Here, a Mass2Motif is defined to be the recurring set of fragment peaks and neutral losses that potentially correspond to a biochemically-relevant substructure shared by many metabolites. Unlike the classification-based methods highlighted earlier, the decomposition of fragmentation spectra into Mass2Motifs is achieved in an unsupervised manner. The MS2LDA workflow is introduced in Section 7.4.

7.2 Related Work

Clustering is commonly used for group fragmentation spectra that are similar to each other. Clusters of spectra can be used for identification by forming a consensus spectrum and matching it against spectral databases. Molecular networking clusters MS1 peaks by their MS2 spectral similarity such that one identifiable metabolite in a cluster facilitates structural annotation of its neighbors [83, 84, 85]. However, only MS2 spectra with high overall (e.g. cosine) spectral similarity are grouped in Molecular Networking. Consequently Molecular Networking may fail to group molecules that share small substructures. In particular, spectra may be placed in different clusters if they share a small number of fragment peaks that related to a common substruture, but their overall global similarities are too different. Even for spectra placed into the same cluster, often manual analysis (by eyes) is required to select the characteristic fragment peaks that represent a potential substructure and are shared by members of the clusters. Another package, MS2Analyzer [86] mines MS2 spectra given the prior knowledge on the fragment patterns of interest to be specified in advance. While generic features, such as CO or H2O losses, will be common to many experiments, sample-specific features can be easily overlooked if they have not been specified *a priori*.

The assumption that spectral consist of building blocks that correspond to substructures is alluded in certain works but not directly mined from the data. Prior knowledge on substructures have been used for the annotations of a small number of moelcules in fragmentation data [87] and for metabolite classification in GC-MS [88, 80]. In CSI:FingerID [82], a fragmentation tree is used to predict (using Support Vector Machine) the molecular 'fingerprint', computed through the implicit assumption that fragments share substructures, of an unknown compound. The resulting fingerprint is used to improve the matching of spectrum of the unknown compound against a vast chemical database (PubChem). Implicit in these methods are the assumption that recurring patterns of fragment peaks and neutral losses values explain the presence of common biological substructures (e.g. a hexose unit, or a CO loss) shared by metabolites.

Latent Dirichlet Allocation has not been applied to metabolomics or mass spectrometry data, but it has been applied to other fields of computational biology in e.g. genomics [89], metagenomics [90], and transcriptomics [91]. In [89], DNA sequence from genomics studies is decomposed into recurring patterns of N-mers nucleotides. A topic in this context corresponds to the set of N-mers (e.g. 'ATGC' as an instance of a 4-mers) that co-occur together across the different genomic sequences of a species, and the objective of the study is characterise the sets of N-mers that corresponds to conserved genes of the species. Similarly in [90], a metagenomic read (essentially a DNA sequence) is decomposed into its topic distribution. The unsupervised decomposition of metagenomic reads into topic distributions is used to improve the binning (clustering) of reads from the same species. In [91], a sample or gene from transcriptomics studies is decomposed into multiple processes in a manner similar to how a document is decomposed into different topics in traditional LDA for text.

7.3 Statement of Original Work

The work discussed in this chapter has been submitted for publication and is still under review. Justin van der Hooft (JvdH) performed the measurements of the Beer samples through mass spectometry, generating the set of fragmentation data that can be used for topic modelling. The author contributed to the design and development of the MS2LDA workflow. This includes the development and optimisation of the feature extraction process, the implementation and testing of inference via LDA and also model validation against multinomial mixture model.

JvdH then analysed the results from MS2LDA for biochemical significance. To assist JvdH in his analysis, the author proposed and developed the visualisation module, MS2LDAVis. To improve the visualisation module, the author integrated elemental formula annotation functionalities. This includes writing a wrapper in MS2LDA to call SIRIUS [92], a Java-based elemental formula annotator. Cristina Mihailescu (CM) implemented another Python-based elemental formula annotator, which was also customised and integrated into MS2LDA by the author.

JvdH then performed molecular networking analysis on the same dataset, which was used for comparison to MS2LDA results. The author performed the identification of metabolites through matching to reference standard compounds and also the differential analysis of Mass2Motifs, and JvdH validated the results.

7.4 A Workflow for Substructure Discoveries and Annotations

Substructure discovery through the MS2LDA workflow consists of two stages: i) the data conversion stage, which prepares the acquired fragmentation data into suitable input format for the workflow, followed by ii) the Mass2Motif discovery stage, which performs topic modelling via LDA to discover mass fragmental patterns, assigns potential candidate elemental formulae to MS1 and MS2 peaks, and visualises the Mass2Motifs in an interactive environment. The key insight of MS2LDA lies in emphasising the parallel between text and mass spectrometry fragmentation data (Figure 7.2A-B). As a text analysis pipeline relying on LDA to decompose documents into topics based on frequently co-occurring words, so MS2LDA decomposes fragmentation spectra into their constituent building blocks of frequently co-occurring fragments and neutral losses (referred to as Mass2Motifs). The complete workflow is illustrated in Figure 7.2C.

Acquired fragmentation data cannot readily be used for the purpose of pattern searching via LDA and has to be converted into a suitable format. As input, the MS2LDA workflow accepts the combination of a single full-scan file for the MS1 peaks and a separate fragmentation file for the MS2 peaks. The data conversion process starts with the detection of MS1 peak in the input .mzXML file obtained from full-scan mode spectra using the CentWave algorithm from the XCMS library [31]. This constitutes information on the parent (MS1) level. Fragmentation data, in the form of .mzML file obtained from tandem MS mode, are processed using an R script based on the RMassBank package [93].

A linking step is required to match the most intense MS2 spectrum in a scan to a parent MS1 peak. Matching is performed via a greedy search within a specified retention time tolerance window, selecting the top few most intense peaks for the matching. This simulates the generative process that produces the spectral data in data-dependant experiments. A filtering step, based on RT and intensity, is applied to remove noisy peaks. Any MS1 peak not having paired MS2 peaks is also discarded for further processing. The aim of the filtering step is to exclude identical fragmentation spectra produced by low-intensity MS1 peaks that were fragmented multiple times, potentially forming spurious and uninformative Mass2Motifs on their own.

Following the bag-of-words assumption, LDA does not consider word orders but instead take into account only the number of times word co-occur in a document. The next step is transforming spectral data into a bag-of-word count matrix (illustrated in Figure 7.3), with entries in the matrix the co-occurrences of discrete MS2 features (‘words’) in the fragmentation spectra linked to a parent MS1 peak (‘document’). From each fragmentation spectra, two types of features can be extracted: fragment features and loss features. Fragment features

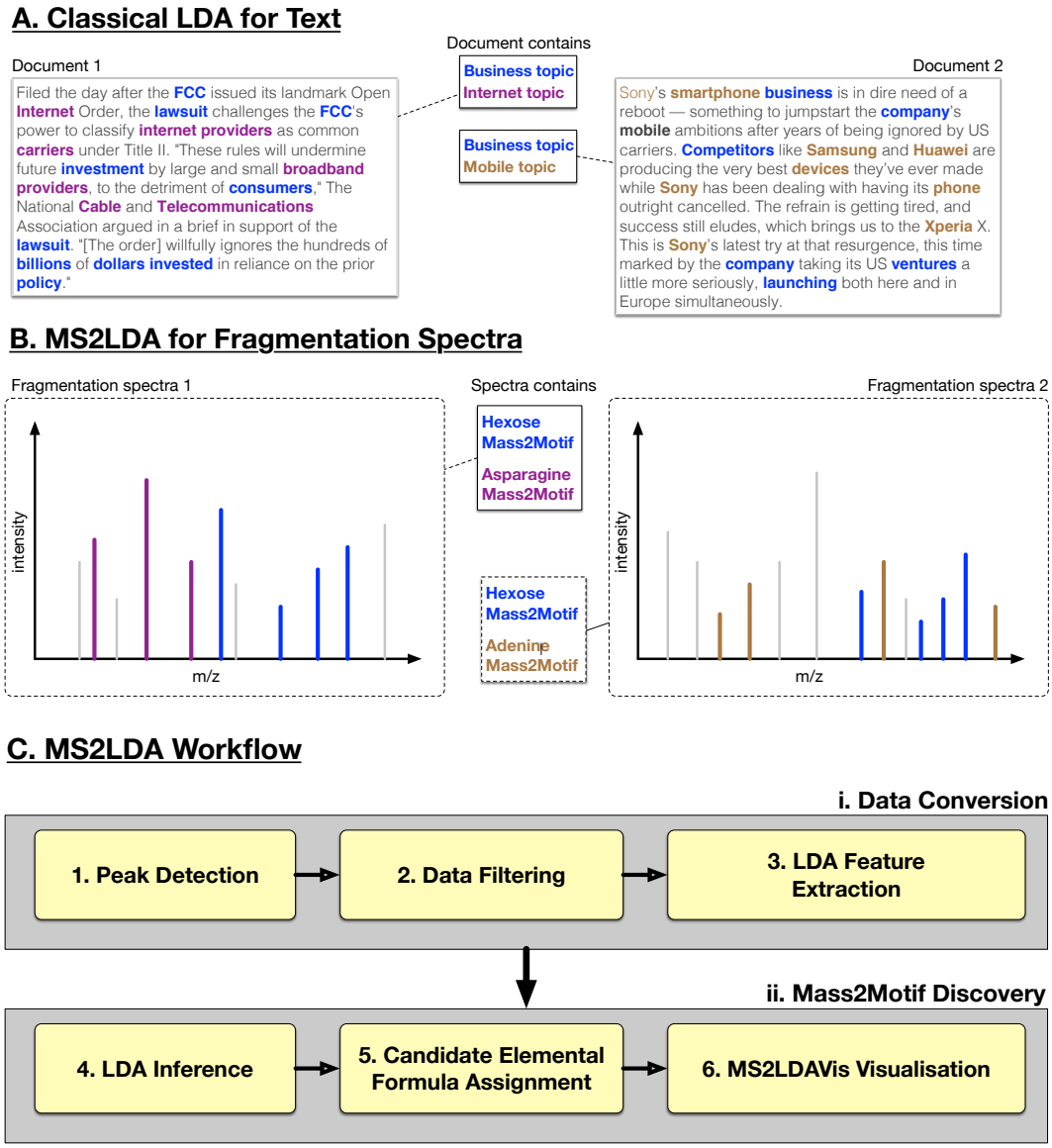


Figure 7.2: **A.** LDA applied to text decomposes a document into its topic distributions (e.g. football, business and enrivonment topics). **B.** Similarly, MS2LDA decomposes a fragmentation spectrum into its topics (Mass2Motifs) that can be characterised as asparagine, hexose and adenine related. Each fragmentation spectra comprise of one or more Mass2Motifs. **C.** Schematic overview of the MS2LDA workflow.

are the discretised m/z values of MS2 peaks, while loss features are formed by discretising neutral losses. A neutral loss, defined as the mass differences between a precursor MS1 peak and each of the child MS2 peaks in the spectrum, corresponds to the removal of a specific neutral fragment from the molecular ion.

MS1 feature extracted →

Binned MS2 fragments ↓

	MS1_a	MS1_b	MS1_c	MS1_d	MS1_e	…
Fragment_119.0351	0	100	24	37	0	
Fragment_136.0629	87	0	17	18	0	
Fragment_156.0769	55	20	0	10	100	
…						
Loss_18.0080	56	0	0	10	15	
Loss_36.0183	0	0	30	0	0	
Loss_46.0053	40	40	10	87	100	
…						

Figure 7.3: The matrix of co-occurrences of fragment and loss features (rows) in each fragmentation spectrum linked to a parent MS1 peak (columns). Entries of the matrix are the counts of the feature from the normalized (0 100 scale) intensities.

Discretisation is performed via a greedy binning process. To group continuous m/z values and create fragment features, a priority queue is used that efficiently maintains the ordering of m/z values of peaks upon insertion. Successive items are popped from the priority queue in ascending order, forming a group of contiguous features — until the next encountered item has an m/z value larger by a predefined tolerance in parts-per-million from the average values of the group, in which case a new group is created. The average m/z values of a group, rounded to 5 decimal places, becomes the discrete representation of fragment peaks in their originating spectra. The count of a fragment feature in a spectrum is computed by dividing the MS2 peak's intensity value to the largest intensity in the spectrum and multiplying by an scaling factor of 100 (equivalent to the discretisation resolution). In this manner, MS2 peaks with larger intensity values are represented more often in the spectra. Neutral loss features are discretised and computed in a manner similar to fragment features. The resulting matrices for fragment and loss features are concatenated and used as input to LDA.

In the context of fragmentation data, the standard LDA model as applied to substructure discovery is described next. The observation on the n-th fragment or loss feature in the d-th fragmentation spectra (w_{dn}) is conditioned on the assignment of feature w_{dn} to the k-th Mass2Motif multinomial distribution. This corresponds to the topic distribution over words in the original LDA model. This assignment is denoted by the indicator variable z_{dn}, so $z_{dn} = k$ if feature w_{dn} is assigned to a k-th Mass2Motif. The k-th multinomial distribution

that a feature is assigned to is characterised by the parameter vector $\boldsymbol{\phi}_{z_{dn}}$, with $\boldsymbol{\phi}_{z_{dn}}$ drawn from a prior Dirichlet distribution with a symmetric parameter β.

$$w_{dn}|\boldsymbol{\phi}_{z_{dn}} \sim Multinomial(\boldsymbol{\phi}_{z_{dn}}) \tag{7.1}$$

$$\boldsymbol{\phi}_k|\beta \sim Dirichlet(\beta) \tag{7.2}$$

The probability of seeing certain Mass2Motifs for each d-th fragmentation spectra is drawn from a multinomial distribution with a parameter vector $\boldsymbol{\theta}_d$, corresponding to the topic decomposition of a document in the original LDA model. This parameter vector $\boldsymbol{\theta}_d$ is in turn drawn from a prior Dirichlet distribution having a symmetric parameter α.

$$z_{dn}|\boldsymbol{\theta}_d \sim Multinomial(\boldsymbol{\theta}_d) \tag{7.3}$$

$$\boldsymbol{\theta}_d|\alpha \sim Dirichlet(\alpha) \tag{7.4}$$

A collapsed Gibbs sampling scheme is implemented in Python for inference (details in Section 3.6). The output from inference is a set of Mass2Motifs and assignments of Mass2Motifs to each MS1 peak.

MS2LDAVis

Given its hypothesis-generating nature, the analysis of Mass2Motifs to characterise and examine their correspondence to actual biochemical substructures is an iterative and exploratory process. This is made possible through the MS2LDAVis module, an interactive web-based visualisation build upon the combination of the Javascript and the D3 library (http://d3js.org). MS2LDAVis is extended from the Python port of the topic modelling visualisation interface LDAVis [94] used in the text domain, but our adaptation MS2LDAVis introduces fragmentation-specific views.

Similar to the original LDAVis, the left panel of MS2LDAVis module shows a global view of the model, whilst the right panel zooms into a specific Mass2Motif (see Figure 7.4A). However, unlike LDAVis where topics are displayed on the left panel through multidimensional scaling that projects topics to two dimensions, the two axes in MS2LDAVis panel are the log-degree and the h-index of Mass2Motifs. The degree of a Mass2Motif as the number of fragmentation spectra explained by the Mass2Motif at the user-defined threshold t_θ on the fragmentation-spectra-to-Mass2Motif distributions ($\boldsymbol{\theta}$). The h-index of a Mass2Motif is defined in a similar manner to the conventional h-index for scientific publications of a researcher. A Mass2Motif has an index of h if it has h fragment or loss features obtained after setting a user-defined threshold t_ϕ on the Mass2Motif-to-features distributions ($\boldsymbol{\phi}$), each of which occur in the set of thresholded spectra at least h times. Intuitively, a Mass2Motif with high degree but low h-index may potentially correspond to simple substructures that occur

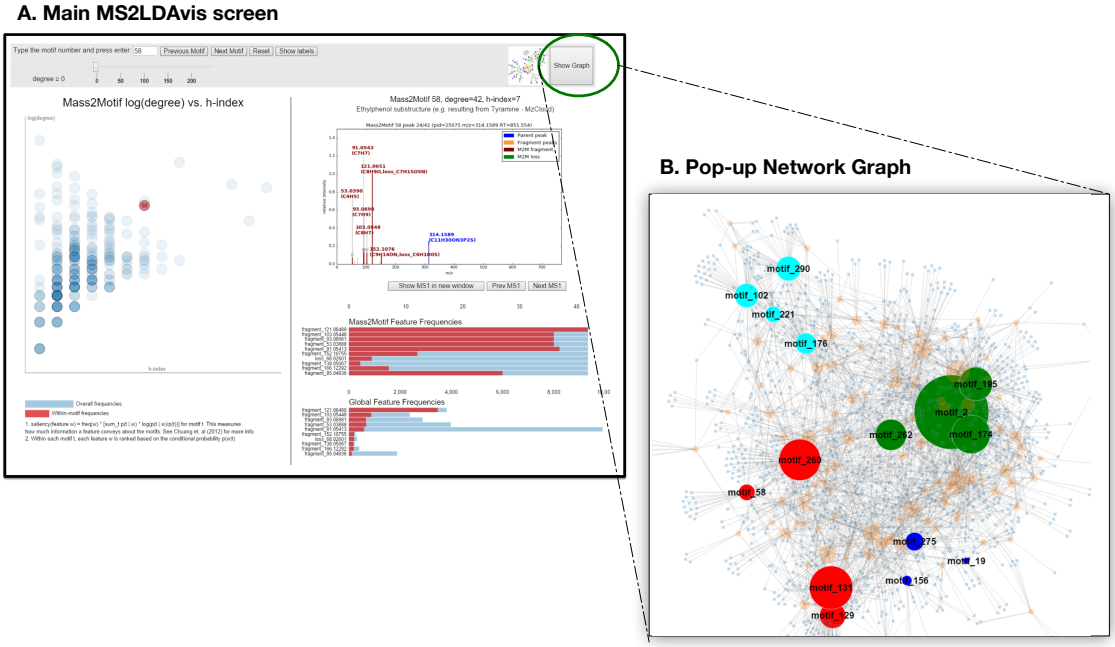


Figure 7.4: Screenshot of MS2LDAVis. See text for explanations of the different panels.

in many fragmentation spectra, while a Mass2Motif with high h-index but low degree are more unique and complex substructures shared by fewer MS2 spectra.

Selecting a Mass2Motif on the left panel of Figure 7.4A changes the specific information displayed on the right panel. Fragmentation spectra that can be explained by the currently selected Mass2Motif (above the threshold t_θ) are plotted, and clicking the Previous MS1 and Next MS1 buttons allows the flipping through consecutive spectra plots. Fragment and loss features that can be explained by the selected Mass2Motif (above the threshold t_ϕ) that also occur in the plotted spectra are highlighted in bold. Two barplots can be found on the bottom right panel: the Mass2Motif Feature Frequencies displays the counts of each fragment or loss features within the entire fragmentation spectra explained by the currently selected Mass2Motif, while the Global Feature Frequencies displays the counts of the fragments or loss features within the complete data set that can be explained by the currently selected Mass2Motif.

To complement the main visualisation view, inference results can also be visualised in a pop-up network graph (Figure 7.4B) by clicking the Show Graph button. In the network view, Mass2Motifs and fragmentation spectra, represented by their parent MS1 peaks in the graph, form the nodes in the graph, and edges are drawn between the nodes if a spectra can be explained by a Mass2Motif above the threshold t_θ. To minimise clutter in the graph, a slider is provided to filter nodes based on their degree values. Nodes in the graph can also be annotated and coloured according to user specifications before the visualisation interface

is called. The two complementary views are linked such that clicking a Mass2Motif node on the network graph will select the corresponding Mass2Motif on the main view and vice versa. The network graph is particularly useful in exploring the relationships between Mass2Motifs and investigating which spectra can be explained by multiple Mass2Motifs.

To aid data interpretation, putative elemental formulae is displayed on the plots of fragmentation spectra explained by a certain Mass2Motif (top-right panel, Figure 7.4). Two methods are integrated within MS2LDA to assign candidate elemental formulae. SIRIUS [92] employs a dynamic programming approach, termed 'Round Robin' [95], to solve elemental formula assignment as an integer decomposition problem. SIRIUS is freely-available and, as it is written in Java, can in theory be run platform-independently on any Windows, Unix and Mac environment (in practice, library dependencies have to be satisfied before SIRIUS can run). Integration of SIRIUS into the MS2LDA workflow is achieved by wrapping calls to the Java classes of SIRIUS through a separate sub-process, passing it a temporary MGF file that corresponds to a fragmentation tree. SIRIUS assigns elemental formulae to each fragmentation tree independently, which may lead to mass fragments of similar m/z value being assigned an elemental formula in some spectra, but not in all.

As an alternative to elemental formula annotation via SIRIUS, CM developed EF-Assigner, a pure Python implementation of an elemental formula assigner based on the Round Robin algorithm on which SIRIUS is based on. In EF-Assigner, candidate formulae are filtered using an implementation of the 7-golden rules, a set of heuristic rules introduced in [74] to remove chemically-unlikely elemental formula compositions from the candidate list. The advantages of EF-Assigner are its easy integration with the rest of the workflow (it is also written in Python) and it can assign elemental formulae to an entire group of MS2 peaks as represented by their discrete fragment and loss features at once. Unlike SIRIUS that uses the complete information of the precursor ion and fragments peaks in a spectrum for annotation, EF-Assigner assigns the elemental formulae for the MS1 peaks and MS2 fragment and loss features independently. The author included EF-Assigner in the MS2LDA workflow, passing it the necessary MS1 peaks and MS2 fragment and loss features for annotation. EF-Assigner is also modified to limit the maximum atom occurrences of certain elements in a candidate formula. For a greater annotation coverage, a second stage process is implemented. After an initial pass of EF-Assigner using a list of common chemical elements of CHNOPS, unannotated MS1 peaks and MS2 features are then re-annotated using an expanded list of possible elements that includes less common elements, such as the C-13 isotope of Carbon, Fluorine and Chlorine.

7.5 Evaluation Study

7.5.1 Evaluation Dataset

To evaluate MS2LDA, four beer samples representative of complex mixtures of diverse biochemically relevant compound classes (such as amino acids, nucleotides, and sugars) typical in metabolomics studies are used. The beer extracts, acquired from one home-brewed beer and three different commercially available beers, are shown in Table 7.1. One of the beer samples (Beer3) is also used for the evaluation of the alignment methods in Chapter 4. Approximately 10 ml of beer was sampled from each bottle directly after opening. As well as the four individual extracts, a pooled aliquot of the four beer extracts was prepared. A Thermo Scientific Ultimate 3000 RSLCnano liquid chromatography system, coupled to a Thermo Scientific Q-Exactive Orbitrap mass spectrometer comprise the overall LC-MS setup.

Following mass spectrometry, blank runs, quality control samples, and 3 standard mixes containing 150 reference compounds were run to assess the quality of the mass spectrometer and aid in metabolite annotation and identification [34]. The pooled sample was run prior to and across the batch to monitor the stability and quality of the LC-MS runs. Beer samples were run in a randomized order. Immediately after acquisition, all RAW files containing information stored in a proprietary vendor-dependant format were converted into the open mzXML format. Mass spectra are centroided and separated into positive and negative ionization modes using the command line version of MSconvert (ProteoWizard). Fragmentation files were also converted into .mzML formats using the GUI version of MSconvert. Accurate masses of standards were obtained well within 3 ppm accuracy and intensities of the quality control samples (a beer extract and a serum extract) were as expected.

Label	Source
Beer1	A home-brewed bottle of German Wheat Beer.
Beer2	A bottle of ‘Jaw Glyde Ale brewed by JAW Brew.
Beer3	A bottle of ‘Seven Giraffes Extraordinary Ale brewed by William Bros. Brewery Company.
Beer4	A bottle of ‘Black Sheep Ale brewed by Black Sheep Brewery.

Table 7.1: Beer samples used for evaluation dataset.

7.5.2 Model Comparison

We performed model selection via a 4-folds cross validation approach on one of the data file (Beer3 positive ionization mode). For each test fold being held out in the Beer3 data file, an estimate of the model evidence is computed after training the model on the remaining

training folds in the file. The number of Mass2Motifs was also selected in this manner from cross-validation.

A crucial difference between LDA and the multinomial mixture-model (clustering) lies in the modelling assumption that a document is a mixture of one or more topics (LDA) as opposed to each document having exactly one topic (clustering). To validate one of our key assumptions of Mass2Motifs represent biological building blocks (i.e. fragmentation spectrum contains more than one Mass2Motifs), we compared the LDA model to a multinomial mixture model that can also be used for the clustering of fragmentation spectra. A comparison of LDA to a multinomial mixture model was performed by the author to assess and validate model fit, evaluated based on perplexity on the held-out data. Perplexity measures how well a probability distribution or probability model predicts a sample and is defined as:

$$perplexity(W) = exp\left(\frac{\sum_d log(P(w_d)}{\sum_d N_d}\right)$$

where $perplexity(W)$ is the perplexity on the whole held-out test collection, $P(w_d)$ is the marginal probability of a testing spectra d (integrating over all the parameters of the model), approximated via an importance sampling method [96] and N_d is the number of features in each testing spectra d. Following [97], the hyper-parameters were set to $\alpha = K/50$ and $\beta = 0.1$ during cross-validation. For mixture model clustering, a non-informative Dirichlet prior ($\alpha = K/50$, where K is now the number of clusters) is set on the proportions of the mixture components and another Dirichlet prior ($\beta = 0.1$) is set on cluster-specific word distributions. The Gibbs sampler for LDA and multinomial mixture model is run for 1000 samples, discarding the first 500 for burn-in. The last sample is used computing the posterior estimates. Minimal differences were found when inferred model parameters were averaged over samples in comparison to using the last sample.

7.5.3 Biochemical Analysis

JvdH performed analyses on each of the beer samples described in Section 7.5.1. Each beer sample was processed independently of the others through MS2LDA. The aim of the analysis was to structurally characterized and annotate any chemically-relevant Mass2Motifs that potentially correspond to actual substructures shared by metabolites.

Validation to Reference Standard Molecules

Mixtures of known standard molecules were run along the beer extracts. On the beer data, the resulting accurate of these standards molecules were within 3 ppm accuracy, making their identification possible. As the identity of these molecules is known, we can use them to

validate our structurally annotated Mass2Motifs. Given the database of exact mass and RT values of the standard molecules, a simple greedy matching scheme is used to establish the identity of MS1 parent peaks in MS2LDA. For each database entry of a standard molecule, we loop over all MS1 peaks in MS2LDA finding peaks that match the accurate mass of the standard molecule within the mass tolerance of 3 ppm and RT tolerance of 5 seconds. If there are multiple candidate MS1 peaks, the peak nearest in mass to the database accurate mass is selected. As these identified MS1 peak have linked spectra that are explained by characterised Mass2Motifs, this allows JvdH to validate the consistency of characterised Mass2Motifs against the identification information of reference standard molecules.

Comparison to Spectral Clustering

Molecular Networking [83, 84, 85] analysis can be used to compare inferred Mass2Motifs from MS2LDA against the clusters produced through the cosine clustering of fragmentation spectra. Spectral clustering (molecular networking analysis) of the four Beer samples was performed by JvdH using the **G**lobal **N**atural **P**roducts **S**ocial (GNPS) environment. The resulting fragmentation spectra for each Beer's .mzXML file was clustered using the MS-Cluster module with a precursor mass tolerance of 0.25 Da and a MS/MS fragment ion tolerance of 0.005 Da. Clustered fragmentation spectra originating from different files are merged to create the consensus spectra (consensus spectra containing less than 2 spectra were discarded). A graph network is created where nodes are consensus spectra and edges are drawn if the cosine similarities between nodes are above 0.55. For identification, spectra in the graph were searched against GNPS' spectral libraries, with a cosine threshold of 0.6 and having at least 4 matched fragment peaks. The resulting graph was exported into Cytoscape and visualised using the FM3 graph layout. Comparison against MS2LDA results were performed manually by JvdH. We also examined the data to find exemplar spectra that can be used to highlight the differences between MS2LDA results and spectral clustering.

Differential Analysis of Mass2Motifs

By linking MS2LDA analysis with the fold changes of MS1 peaks, the differential expression of Mass2Motifs can be assessed. This allows for the comparison of biochemical changes across groups of samples based on which metabolites can be explained by a Mass2Motif. As we hypothesise that more fragmentation spectra can be explainable by MassMotifs — in comparison to the number of spectra that can be annotated or identified through conventional matching to spectral library — the presence of shared substructures can reveal a shared pattern of differential expression among the set of metabolites explained by a Mass2Motif. This is possible even if these metabolites do not share a large degree of overall spectral

similarity, which is often a necessary prerequisite in the identification of groups metabolites that share the same substructure.

The full-scan (MS1) LC-MS run for each Beer extract was processed using an in-house metabolomics pipeline based on XCMS [31] and MzMatch [54]. A peak table, containing information on the MS1 peak intensities, was exported to .csv files and linked to the parent MS1 peaks in MS2LDA through a greedy matching scheme that establishes the correspondence of parent peaks in MS2LDA to the MS1 peaks in the exported peak table within a specified m/z and RT tolerance values (3 ppm, 30 seconds). If there are multiple possible matches, the one with the nearest m/z difference is selected. Following this, for each Mass2Motif, a matrix is constructed where each row is a linked MS1 peak that can be explained by that Mass2Motif and the columns are intensity values from the different case and control groups. This matrix is used as input to our implementation of PLAGE [98]. PLAGE is selected as it is evaluated to be the best method in [99], however this does not preclude using any other methods surveyed in e.g. [99] from being applied to the differential analysis of Mass2Motifs.

7.6 Results & Discussions

7.6.1 Model Comparison

Figure 7.5 shows the perplexity for the two models on one of the Beer extracts (Beer3) as a function of K, the number of Mass2Motifs (for LDA) or clusters (for the mixture model). The mixture model is essentially equivalent to LDA with each spectrum being forced to consist of only one Mass2Motif. As such, if LDA is indeed finding structural features as conserved patterns of fragments and losses, it should explain the data with fewer Mass2Motifs than the mixture model. This is because the mixture model has to create separate Mass2Motifs for all observed combinations of structural features. The lower perplexity in Figure 7.5 demonstrates that LDA provides a better model fit on the held-out data compared to multinomial mixture model due to its lower perplexity. This validates our assumption that allowing multiple conserved blocks to be present in small molecule fragmentation data is a better representation of the biochemical properties of the fragmented molecules. The perplexity result on the held-out data in Figure 7.5 suggests a reasonable value for K to be in the range of 200 to 400, at the elbow of the curve where increasing the number of topics does not result in further decrease of perplexity.

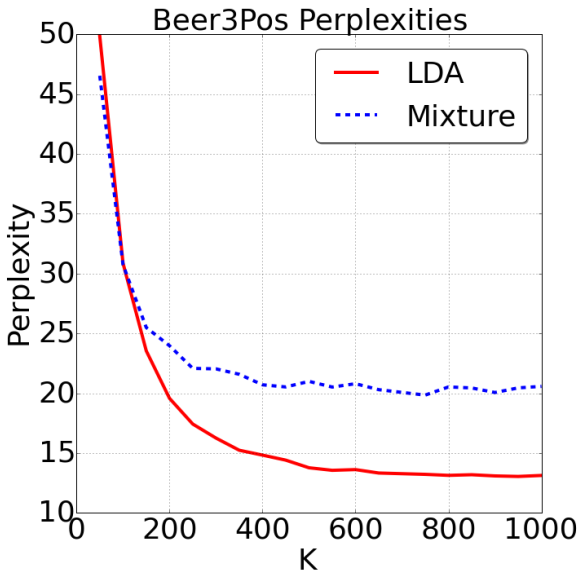


Figure 7.5: Results of model comparisons of LDA and multinomial mixture model on the Beer3 data. The lower perplexity values for $K > 100$ demonstrates that LDA provides a better model fit on the held-out data when compared to the mixture model.

7.6.2 Biochemical Analysis

With K the number of Mass2Motifs set to 300 and other hyperparameters set to be the same as in cross-validation, Mass2Motifs were extracted for each Beer data and characterised by JvdH for biochemical relevance. As discussed in Section 7.4, the distributions over the features that make up the Mass2motifs and the distributions over Mass2motifs for each fragmentation spectrum can be thresholded in MS2LDAVis for results interpretation. For analysis, the threshold values of 0.05 and 0.01 for t_θ and t_ϕ were set, but they can easily be varied. The selection of these threshold values was based on JvdH's expert knowledge to allow for the extraction of a chemically-plausible set of features that comprise a Mass2Motif.

In the subsequent analysis that follows, Mass2Motifs with degrees 10 (i.e. that were present in ten or more spectra after thresholding) were manually inspected and annotated at different levels of confidence through integrating multiple supporting evidence such as the matching to a database of known reference standard compounds and spectral matching of the MS2 spectra containing the associated fragments and/or neutral losses to the reference spectra in MzCloud (www.mzcloud.org). Key fragment or loss features from the annotated Mass2Motifs in one sample were then searched against the list of Mass2Motifs in other samples and their correspondences established if those key fragment or loss features were present in both.

Across the four Beer data, an average of 70% of spectra (Table 7.2) include at least one annotated Mass2Motif, with Mass2Motifs related to the same substructure consistently found across multiple beers (e.g. hexose-related Mass2Motifs were present in all positive ionization mode files with degrees from 58 to more than 100), despite the fact that each sample

File	Total MS1 peaks	Linked to at least one structurally annotated M2M	%
Beer1Pos	1282	951	74
Beer2Pos	1567	1160	74
Beer3Pos	1422	1055	74
Beer4Pos	1363	930	68

Table 7.2: Mass2Motif coverage of MS1 peaks by percentage of MS1 peaks that can be explained by at least one structurally annotated Mass2Motif for the files acquired in positive ionization mode.

was processed through the workflow independently. Between 30 to 40 Mass2Motifs in each of the Beer sample could be structurally annotated as corresponding to a diverse set of biochemical substructures, including amino acid related (i.e. histidine, leucine, tryptophan, and tyrosine), nucleotide related (i.e. adenine, cytosine, and xanthine), and other molecules such as cinnamic acid, ferulic acid, ribose and N-acetylputrescine. In general, the more Mass2Motifs present in a particular spectrum, the more specific our annotations can potentially become. An exhaustive identification effort to characterise all spectra (metabolites) present in the data was not attempted by JvdH, as it would be a major undertaking on its own, however it is noted that annotating just 30 to 40 of the discovered Mass2Motifs provide some structural biochemical insights into 70% of the spectra. This suggests that a large percentage of metabolites can be automatically classified according to function (based on presence of functional groups or as a part of biological pathways).

As an example of the biochemical insights that can be obtained by an expert from MS2LDAVis, Figure 7.6 shows three of the eleven spectra that include Mass2Motif 19, characterised as corresponding to ferulic acid substructure. Ferulic acid is a compound found in the hard outer layer of grain (the bran) of cereals (an ingredient of beer) and is expected to be shared by the metabolites in beer as a substructure. Across the three spectra, we see conserved fragment and loss features shared by the spectra explained by Mass2Motif 19, with the most conserved features highlighted in Figure 7.6D. Unlike MS2Analyzer [86] where the prior information on the fragment features of interest has to be specified in advance, the discovery of conserved features in MS2LDA is performed in an unsupervised manner. In addition, JvdH verified that the *loss˙176.1086* feature in Figure 7.6B is an informative feature related to the complete ferulic acid substructure. While this is easily observed from the visualisation, information on conserved patterns of neutral loss will be difficult to extract from any other tools apart from MS2LDA. The entire results in Figure 7.6 shows that through MS2LDA, we can extract a biochemically relevant pattern present in just eleven of the entire set ($>$1000) of spectra, although the individual spectra can be quite different.

In a comparison to metabolite identification via spectral library matching using the NIST MS/MS database for small molecules (http://chemdata.nist.gov/mass-spc/msms-search/) and MassBank [77], only one from the eleven spectra explained by Mass2Motif 19 returns a fer-

ulic acid related hit. This is despite the clear presence of fragment and loss features corresponding to ferulic acid substructure across the eleven spectra. Similarly, the beer metabolites explained by Mass2Motifs related to histidine, tyrosine, and tryptophan were subjected to spectral matching. In the verification by JvdH, matches to reference spectra were found for 33 spectra with 15 matches consistent with their characterised Mass2Motifs. These results demonstrate how MS2LDA effectively recognizes core substructures in mass spectral data and can serve as an aid to the classification and annotation of metabolites. Critically, it does this by matching only small portions of the spectra (substructures) rather than relying on complete spectral matches. In summary, for this subset of four Mass2Motifs, spectral matching allows classification of 45% of the associated metabolites whereas MS2LDA is able to functionally annotate all of them. In addition, MS2LDA can annotate and group spectra based on neutral losses (e.g. the loss of a free carboxylic acid group) which is not possible via spectral matching.

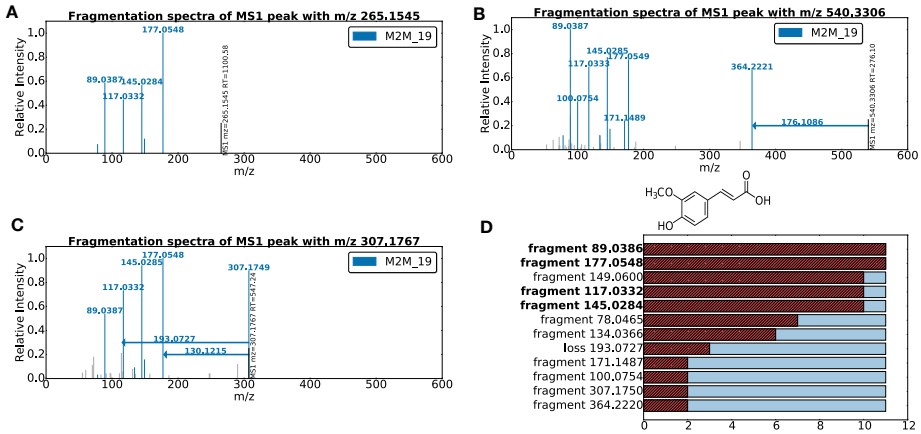


Figure 7.6: Three spectra, from the beer3 positive ionization mode file, each of which includes Mass2Motif 19, annotated as the plant derived ferulic acid substructure. A-C highlight mass fragments and neutral losses (arrows originating at the precursor ions) included in Mass2Motif 19 (fragments not explained by Mass2Motif 19 are light grey). Ferulic acid substructure is illustrated at the top of D, while the boxplot in D shows how common each fragment or loss features (representative of the substructure) are found in the 11 spectra explained by Mass2Motif 19 found in the dataset. Features highlighted in bold are consistently present in Mass2Motifs inferred across the four beer samples.

Validation to Reference Standard Molecules

Of the 45 molecules we were able to identify as standard molecules in one or more of the beer extracts, 38 can be explained by one or more annotated Mass2Motif, and 32 of the annotated Mass2Motifs correspond to known biochemical features that are consistent with the standard molecules. This demonstrates that characterised Mass2Motifs represent conserved patterns of metabolites' fragmentation spectra in authentic standard mixtures. Figure 7.7 shows some

examples for these fragmentation spectra coloured by characterised Mass2Motifs. The spectra for phenylalanine (Figure 7.7A) and histidine (Figure 7.7B) share Mass2Motif 262, and indeed feature *loss_46.0054*, which has been verified by JvdH as informative that a carboxylic acid group (CHOOH) is lost from the molecular ion during fragmentation, is a common characteristic of phenylalanine and histidine. Similarly, other Mass2Motifs (115, 241) in Figures 7.7A and 7.7B are related to phenylalanine and histidine compounds. Finally, Figure 7.7D is the MS2 spectrum of adenosine, which consists of an adenine molecule conjugated to a ribose sugar molecule. The two associated Mass2Motifs 156 and 220 correctly represent the two biochemically relevant substructures (i.e., adenine substructure and a loss corresponding to a ribose sugar).

Note that in our analysis on these standard molecules, the inferred Mass2Motifs were characterised first without any prior knowledge on the identities of standard compounds, but we still observe a high level of agreements between the identifications of standard compounds and the independent characterisation of Mass2Motifs which explains identified spectra. This suggests an alternative to the usual procedure where identification is performed first and the common substructures, shared by the small set of identified compounds, are deduced. In the complementary approach, characterised Mass2Motifs can be used as a starting point for analysis. The large number of spectra that can be explained by Mass2Motifs are further examined and their putative identities deduced through collaborating multiple evidences, such as substructure annotation, matching against standard database, MS2 spectral library, etc.

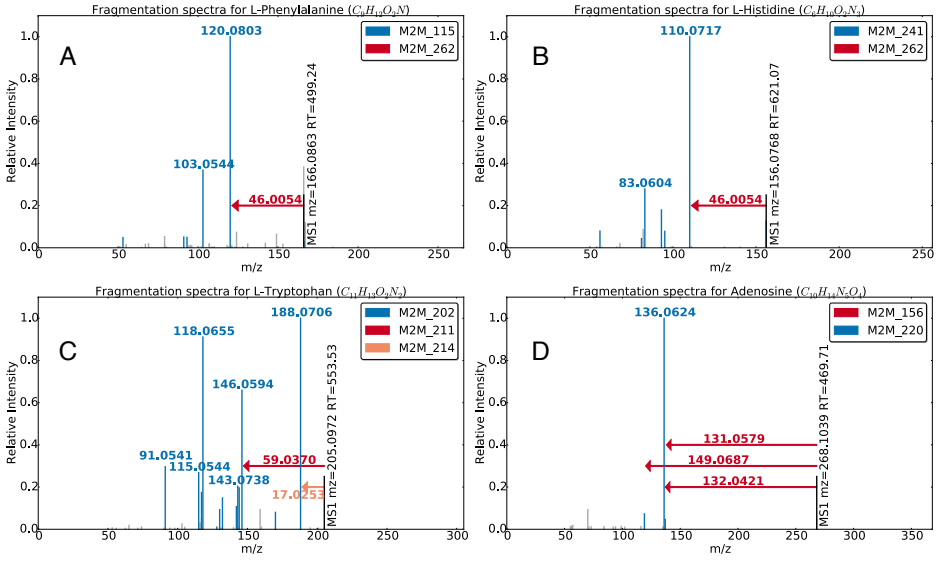


Figure 7.7: Mass2Motif spectra of identified standard molecules A) L-histidine, B) L-phenylalanine, C) L-tryptophan, and D) adenosine, with their characterized motifs (see Table 7.3) indicated by colours.

Mass2Motif	Annotation	Degree	Fragment or Loss Features	Elemental Formula
115	[phenylalanine-CHOOH]-based substructure.	28	fragment_120.0808, fragment_103.0546, fragment_91.0541	C8H10N, C8H7, C7H7
156	[ribose (pentose, C5-sugar)-H2O]-related loss.	22	loss_132.0421	C5H8O4
202	[tryptophan-NH3]-related substructure.	15	fragment_118.0654, fragment_117.0571, fragment_91.0541, fragment_130.0645, fragment_188.0706	C8H8N, C8H7N, C7H7, C9H8N, C11H10NO2
211	N-acetylputrescine substructure.	24	loss_59.0370, fragment_114.0912, fragment_72.0447, fragment_60.0448	C2H5NO, C6H12NO, C3H6NO, C2H6NO
214	amine loss.	57	loss_17.0247	NH3
220	adenine substructure.	32	fragment_136.0629, fragment_119.0351	C5H6N5, C5H3N4
241	histidine substructure.	21	fragment_110.0718, fragment_156.0769, fragment_93.0450, fragment_95.0608	C5H8N3, C6H10N3O2, C5H5N2, C5H7N2
262	combined loss of H2O and CO – indicative for free carboxylic acid group (COOH).	90	loss_46.0053	CH2O2

Table 7.3: Annotations of the Mass2Motifs associated to the fragmentation spectra of the peaks generated by the standard molecules shown in Figure 7.7. The degree of a Mass2Motif indicates the number of MS2 fragmentation spectra in Beer3 positive ionization mode data having the fragment or loss features that can be explained by the Mass2Motif.

Comparison to Spectral Clustering

Spectral clustering approaches (e.g. Molecular Networking) can also help in molecular annotation by propagating identifications through the network. For example, if one spectrum can be identified, it can be used to putatively annotate the spectrums neighbours in the network. MS2LDA differs from this approach in three key ways. Firstly, MS2LDA does not require any complete spectra to be identified (they can be putatively annotated from Mass2Motifs). Secondly, MS2LDA does not require a high degree of total spectral similarity to allow spectra to share annotations; it just relied on the presence of a shared Mass2Motif. Finally, because spectra can include multiple Mass2Motifs, they can be given multiple annotations while in spectral clustering, each spectrum can only belong to one cluster. A key characteristic of MS2LDAis the ability to decompose MS2 spectra into multiple (potentially biochemically relevant) components. For example, in each of Figures 7.7A to 7.7D, we observe the spectra being decomposed into 2 or more Mass2Motifs. To our knowledge, no other methods can do this in an unsupervised manner without training spectra consisting of known structures or *a priori* knowledge of interesting combinations of fragment and/or loss features.

Similarly in MS2LDA, a fragmentation spectrum can now be described by one or more Mass2Motifs. Figure 7.8 demonstrates this with an example of a subset of the network produced by MS2LDAVis, consisting of spectra explained by two Mass2Motifs characterised as ferulic acid and ethylphenol. All but one spectrum can be explained by just one of the Mass2Motifs but one spectrum is generated by a molecule that contain both substructures and can therefore be explained by both Mass2Motifs. In the Molecular Networking analysis by JvdH, this spectrum is placed into the ethylphenol cluster, but its relationship with ferulic acid is lost. This results in a much less specific annotation of that spectrum. In contrast, the knowledge on the presence of both Mass2Motifs in the spectrum allows JvdH to assign it a putative compound identification of feruloyltyramine ([C18H20NO4]+) despite spectral matching producing no relevant hits. In general, the more Mass2Motifs present in a particular spectrum, the more specific our annotations can potentially become.

The same spectral clustering result is also reproduced in Figure 7.9 where a matrix of cosine similarities of the spectra, placed in the ferulic acid based cluster and the ethylphenol based cluster (from Molecular Networking). Two distinct groups of spectra, based on their cosine similarities, can be seen — corresponding to each cluster. Members of each cluster can also be explained by a single Mass2Motif (the ferulic acid cluster by M2M˙19, and the ethylphenol cluster by M2M 58). However, one spectrum (the last row in Figure 7.9) can also be jointly explained by the two Mass2Motifs. In cosine clustering, this spectrum would have to go into one cluster or the other based on its cosine similarity and valuable information is lost. Since a compound consists of multiple substructures, allowing each spectra to be explained by multiple Mass2Motifs naturally results in a greater potential of producing a

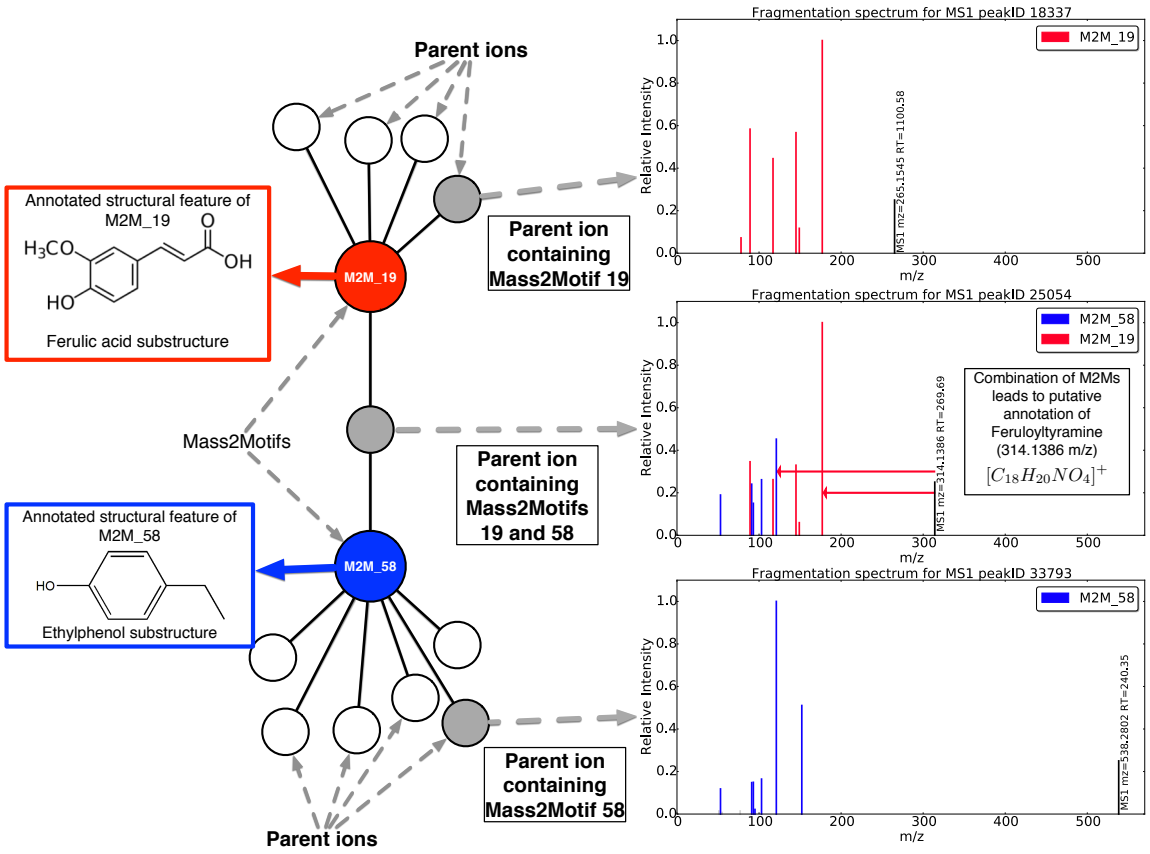


Figure 7.8: Mass2Motifs 19 and 58 were found to be representative of ferulic acid and ethylphenol, respectively. 11 fragmentation spectra can be explained by M2M˙19, while 42 spectra can be explained by M2M˙58. However, one spectra (shown as a gray node in the Figure) can be explained by both Mass2Motifs, but this is not possible in spectral clustering.

more comprehensive characterisations of the substructures of a compound.

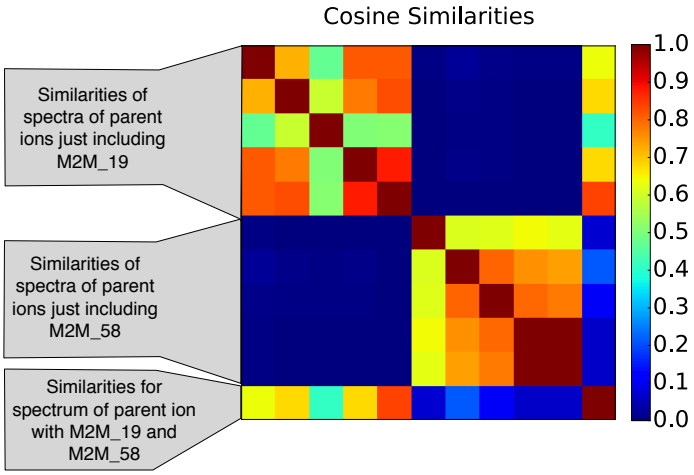


Figure 7.9: Cosine clustering results of spectra drawn from the ferulic acid based cluster and the ethylphenol based cluster (similar to M2M˙19 and M2M˙58). The last row represents a fragmentation spectrum that contains both substructures, but in the clustering approach, the spectra will be placed into one of the clusters based on its cosine similarity. In LDA, this spectrum can be explained by Mass2Motifs that characterise both substructures.

Differential Analysis of Mass2Motifs

We have shown that MS2LDA analysis can group molecules according to a shared Mass2Motif. As spectra can include multiple Mass2Motifs, so molecules can belong to multiple functional groups. In transcriptomic studies, it is common to consider the shared differential expression (DE) of a group of transcripts that are related through the sharing of the same Gene Ontology classification. The equivalent case in metabolomics are metabolites that share the same functional substructures and can potentially be mapped onto related pathways. The presence of the same functional substructure across these metabolites naturally suggest that their spectra can be described by the same Mass2Motif. If all metabolites sharing the same substructure are differentially expressed across samples, hypothesis can be generated as to the underlying biochemical significance causing the expression changes. From performing differential analyses on the expressions (intensity values) of metabolites having spectra explained by the same Mass2Motif, it is therefore possible to assess the biochemical changes of groups of metabolites across samples. Note that this does not depend on the small number of metabolites having spectra that can be identified through spectral matching, instead it relies on the much larger sets of MS1 peaks having spectra that can be jointly explained by a Mass2Motif.

Using PLAGE [98], we assessed the DE of each Mass2Motif based on the intensity changes of the relevant MS1 peaks between beers 2 and 3. Figure 7.10 shows MS1 intensities of

metabolites explained by two Mass2Motifs (characterised as guanine and pentose loss) with high PLAGE scores. In each case, the change in intensity across the two beer extracts are very clear (note that PLAGE considers changes in both directions when scoring). Within the molecules having spectra explained by the guanine Mass2Motif, we could annotate 5-guanine containing metabolites and identify 2 to through matching to reference standards (Figure 7.10A). For the pentose Mass2Motif, we could annotate 8 and identify 5 pentose-containing metabolites from the Mass2Motif (Figure 7.10B). These biochemically relevant metabolites show interesting patterns in the DE between the two beers. As an example of how MS2LDA differential analysis can support hypothesis generation for an expert, JvdH noted that in Beer3, the free guanine is present more often, whereas in Beer2, the conjugates of guanine are more abundant (Figure 7.10A). This reflects the differences in the chemical components of the two beers. Similarly, as metabolites can include multiple Mass2Motifs, JvdH observed that the four spectra (in Figures 7.10) annotated as guanine-related metabolites (i.e., guanosine, two methyl-guanosine isomers, and a pentosyl-hexosylguanine) are also connected to the pentose loss Mass2Motif, which itself was also differentially expressed between the two beers. Indeed, the structures of those metabolites all share both a guanine and a pentose substructure. A comparison made by JvdH to Molecular Networking results revealed that in the standard spectral similarity approach, these spectra were distributed over 10 spectral clusters. In other words, the interesting structural and intensity similarity between these molecules exposed by MS2LDA would not be found via spectral clustering.

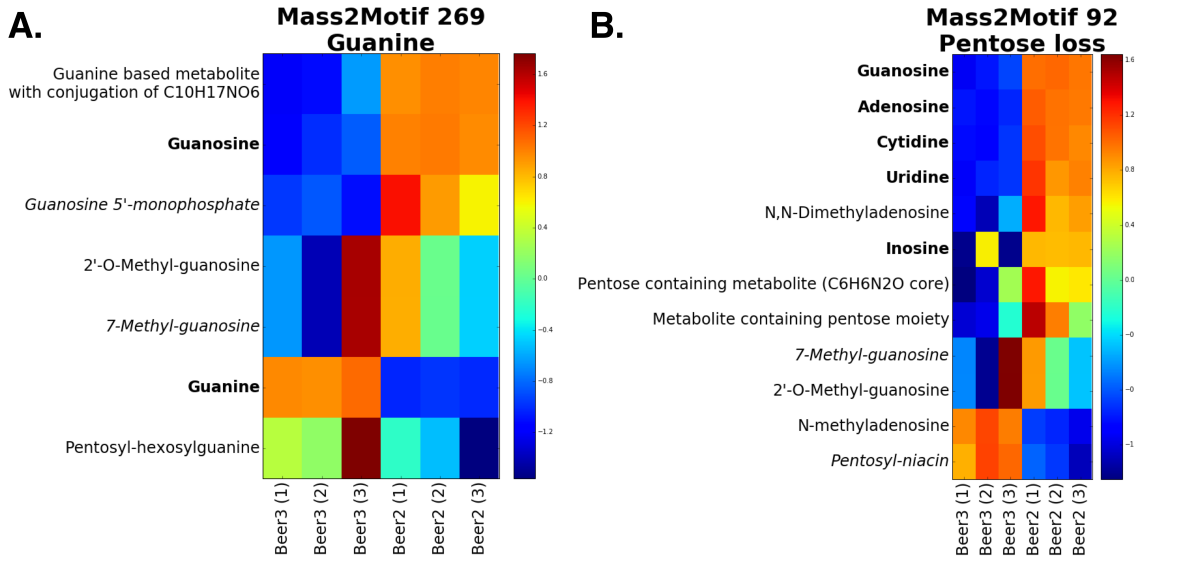


Figure 7.10: Log fold change heat-maps for the A) guanine and B) pentose loss Mass2Motifs. Each row is an annotated parent MS1 peak and columns represent different beer extracts. Bold names for parent MS1 peaks could confidently be matched to reference compounds, while italic names are for those that are annotated at a lower degree of confidence.

7.7 Substructure Discoveries Across Many Fragmentation Files

Manual inspection of the results revealed that many Mass2Motifs, related to the same substructures, are consistently present in two or more beers. This is despite each sample being processed independently through MS2LDA. For example, the hexose-related Mass2Motifs are present in all positive ionization mode beer files with degrees from 58 to more than 100 in each beer. The results suggest that we can jointly model the presence or absence of Mass2Motifs across many input files at once, eliminating the necessary but tedious matching of Mass2Motifs across files if they were to be inferred independently for each input file.

7.7.1 Multi-file LDA Model

Metabolomics dataset consist of fragmentation spectra in multiple input files, where each file is generated from measurements of a technical or biological replicate. In this manner, Mass2Motif distributions over fragment and loss features are shared across files, but within each file, fragmentation spectra have their own file-specific probabilities of observing certain Mass2Motifs. When only a single input file is provided, the multi-file LDA model reduces to the standard LDA model.

In the proposed multi-file LDA model, the observation on the n-th fragment or loss feature in the d-th fragmentation spectra in file f (w_{dn}^f) is conditioned on the assignment of feature w_{dn}^f to some k-th global Mass2Motif multinomial distribution that is shared across files. This assignment is denoted by the indicator variable $\boldsymbol{z}_{dn}^f$, so $\boldsymbol{z}_{dn}^f = k$ if feature n from fragmentation spectra d in file f is assigned to the k-th Mass2Motif. The probability of seeing certain Mass2Motifs for each d-th fragmentation spectra in file f is then drawn from a multinomial distribution with a parameter vector $\boldsymbol{\theta}_d^f$. This parameter vector $\boldsymbol{\theta}_d^f$ is in turn drawn from a prior Dirichlet distribution having the parameter vector $\boldsymbol{\alpha}^f$.

$$\boldsymbol{z}_{dn}^f|\boldsymbol{\theta}_d^f \sim Multinomial(\boldsymbol{\theta}_d^f) \tag{7.5}$$

$$\boldsymbol{\theta}_d^f|\boldsymbol{\alpha}^f \sim Dirichlet(\boldsymbol{\alpha}^f) \tag{7.6}$$

As in the case of standard LDA, the k-th multinomial distribution for a Mass2Motif is characterised by the parameter vector $\boldsymbol{\phi}_{\boldsymbol{z}_{dn}^f}$, with $\boldsymbol{\phi}_{\boldsymbol{z}_{dn}^f}$ drawn from a prior Dirichlet distribution having the parameter vector $\boldsymbol{\beta}$.

$$\boldsymbol{w}_{dn}^f|\boldsymbol{\phi}_{\boldsymbol{z}_{dn}^f} \sim Multinomial(\boldsymbol{\phi}_{\boldsymbol{z}_{dn}^f}) \tag{7.7}$$

$$\boldsymbol{\phi}_k|\boldsymbol{\beta} \sim Dirichlet(\boldsymbol{\beta}) \tag{7.8}$$

Inference in the multi-file LDA model is again performed via a collapsed Gibbs sampling scheme. The conditional probability of $P(\boldsymbol{z}^f_{dn} = k|\boldsymbol{w}^f_{dn}, ...)$ of the assignment of feature n in spectra d file f to Mass2Motif k is given by eq. (7.9).

$$P(\boldsymbol{z}^f_{dn} = k|\boldsymbol{w}^f_{dn}, ...) \propto P(\boldsymbol{w}^f_{dn}|\boldsymbol{z}^f_{dn} = k, ...)P(\boldsymbol{z}^f_{dn} = k|...) \tag{7.9}$$

where ... denotes any other parameters being conditioned upon but not explicitly listed. Similar to the derivation of standard LDA, we can marginalise over all $\boldsymbol{\phi}_k$ parameters in the likelihood term, $P(\boldsymbol{w}^f_{dn}|\boldsymbol{z}^f_{dn} = k, ...)$ of eq. (7.9), to obtain:

$$P(\boldsymbol{w}^f_{dn}|\boldsymbol{z}^f_{dn} = k, ...) \propto \frac{\sum_f c^f_{kn} + \boldsymbol{\beta}_n}{\sum_n \sum_f c^f_{kn} + \boldsymbol{\beta}_n} \tag{7.10}$$

where $\sum_f c^f_{kn}$ is the total number of the n-th feature from all files currently assigned to Mass2Motif k (this count excludes the current feature being sampled in the current iteration of Gibbs sampler). For the prior term $P(\boldsymbol{z}^f_{dn} = k|...)$, marginalising over all $\boldsymbol{\theta}^f_d$ parameters produces as in the standard LDA:

$$P(\boldsymbol{z}^f_{dn} = k|...) \propto c^f_{dk} + \boldsymbol{\alpha}^f_k \tag{7.11}$$

with c^f_{dk} the number of features from document n in file f currently assigned to Mass2Motif k, excluding the current feature being sampled. Putting the prior and likelihood terms together, the following predictive distribution is obtained for the assignment of feature n from document d file f to Mass2Motif k:

$$P(\boldsymbol{z}^f_{dn} = k|\boldsymbol{w}^f_{dn}, ...) \propto (c^f_{dk} + \boldsymbol{\alpha}^f_k) \cdot \frac{\sum_f c^f_{kn} + \boldsymbol{\beta}_n}{\sum_n \sum_f c^f_{kn} + \boldsymbol{\beta}_n} \tag{7.12}$$

In each iteration of the Gibbs sampling, the information on the current feature n in spectra d file f being sampled is removed. Reassignment of the feature to a Mass2Motif is then performed by sampling $\boldsymbol{z}^f_{dn}$ from the distribution specified by eq. (7.12). Given $\boldsymbol{z}$, the predictive distribution for the d-th spectrum over the Mass2Motifs, $\boldsymbol{\theta}^f_d$, is obtained from the expectation of the Dirichlet-Multinomial distribution defined in eqs. (7.5)-(7.6):

$$\hat{\boldsymbol{\theta}}^f_{dk} = \frac{c^f_{dk} + \boldsymbol{\alpha}^f_k}{\sum_k c^f_{dk} + \boldsymbol{\alpha}^f_k} \tag{7.13}$$

where c^f_{dk} is the count of features from spectra d in file f assigned to Mass2Motif k.

For each spectra, the multinomial count vector $\mathbf{c}^f_d$, of features from the spectra that are assigned to the different Mass2Motifs, is a sample from the Dirichlet-Multinomial distribution

defined in eqs. (7.5)-(7.6). Given all the $\mathbf{c}_1^f, \mathbf{c}_2^f, ...\mathbf{c}_D^f$ vectors in the file, the parameter $\boldsymbol{\alpha}^f$ of the Dirichlet-Multinomial distribution of spectra-to-Mass2Motifs in file f can be estimated by maximizing the log likelihood $log \prod_{d=1}^{D} p(\boldsymbol{c}_d^f|\boldsymbol{\alpha}^f)$. An iterative procedure to approximate this is described in [100]. The lower bound on the log likelihood of the multinomial data given $\boldsymbol{\alpha}^f$ is obtained from the iterative update:

$$\boldsymbol{\alpha}_k^f = \boldsymbol{\alpha}_k^f \frac{\sum_d \Psi(c_{dk}^f + \boldsymbol{\alpha}_k^f) - \Psi(\boldsymbol{\alpha}_k^f)}{\Psi(c_d^f + \sum_k \boldsymbol{\alpha}_k^f) - \Psi(\sum_k \boldsymbol{\alpha}_k^f)} \tag{7.14}$$

where $\Psi(x) = \frac{\Gamma'(x)}{\Gamma(x)}$ is the digamma function.

In a similar manner, for each k-th Mass2Motif, the predictive distribution over features, $\boldsymbol{\phi}_k$, can be obtained as the expectation of the Dirichlet-Multinomial distribution defined in eqs. (7.7)-(7.8):

$$\hat{\boldsymbol{\phi}}_{kn} = \frac{c_{kn} + \boldsymbol{\beta}_n}{\sum_n c_{kn} + \boldsymbol{\beta}_n} \tag{7.15}$$

where c_{kn} is the count of the n-feature from all files that are assigned to Mass2Motif k.

7.7.2 Results & Discussion

On the dataset of four Beer extracts in positive ionisation mode processed through multi-file LDA using the same hyperparameters as the individual LDA. For data interpretation, initially, the same threshold values on t_θ and t_ϕ were selected as the previous single-file analysis (0.05 and 0.01 respectively). Table 7.4 shows the results of five global Mass2Motifs that could be matched to the individual LDA results in Section 7.6.2. The results in Table 7.4 shows that multi-file LDA produces comparable results on the Mass2Motifs composition. This is entirely expected given that the four Beer extracts used for evaluation share similar metabolic profiles and correspondingly, have many substructures in common.

Information from all files now contribute to the inference of global Mass2Motifs. The fact that global Mass2Motifs that are consistent with our previous characterisation in Section 7.6.2 still emerge suggests the same underlying patterns of fragment and loss features to be present in each Beer extract. Figure 7.11 shows four example fragmentation spectra originating from different Beer extracts — jointly inferred by multi-file LDA as containing the Mass2Motif characterised as the ferulic acid substructure. While this can be achieved from independently running LDA on each file, the tedious matching process of common Mass2Motifs across files can now be eliminated. Inspections on the degree (the number of spectra associated to a Mass2Motif above the user-defined threshold t_θ) of the five Mass2Motifs in Table 7.4 revealed that with a minor adjustment to t_θ, the same sets of

Mass2Motif	Annotation	Top Features Above Threshold
M2M_17	Ferulic acid substructure	**fragment_177.05478**, **fragment_89.03865**, **fragment_145.02844**, **fragment_117.03319**, loss_58.98941, fragment_163.03887, **fragment_149.05998**, loss_88.09967
M2M_155	Histidine substructure	**fragment_110.07161**, **fragment_156.07687**, fragment_83.06041, **fragment_93.04511**, fragment_82.05246, fragment_209.10558, **fragment_95.06057**, loss_167.08663, fragment_81.04494, loss_191.0615
M2M_115	Leucine substructure	**fragment_86.09653**, **fragment_132.10165**, fragment_69.07013, fragment_332.112, fragment_143.11763
M2M_95	Water loss substructure	**loss_18.01031**, fragment_314.0859, fragment_296.07259
M2M_232	Asparagine substructure	**fragment_136.06231**, loss_162.03459, **fragment_119.0354**, loss_162.00534, fragment_137.04623

Table 7.4: Five global Mass2Motifs inferred from multi-file LDA. For each Mass2Motif, the top features above threshold are listed. Features characterised as key to the substructure from the previous individual LDA analyses are shown in bold.

fragmentation spectra previously associated to the listed Mass2Motifs can all be recovered.

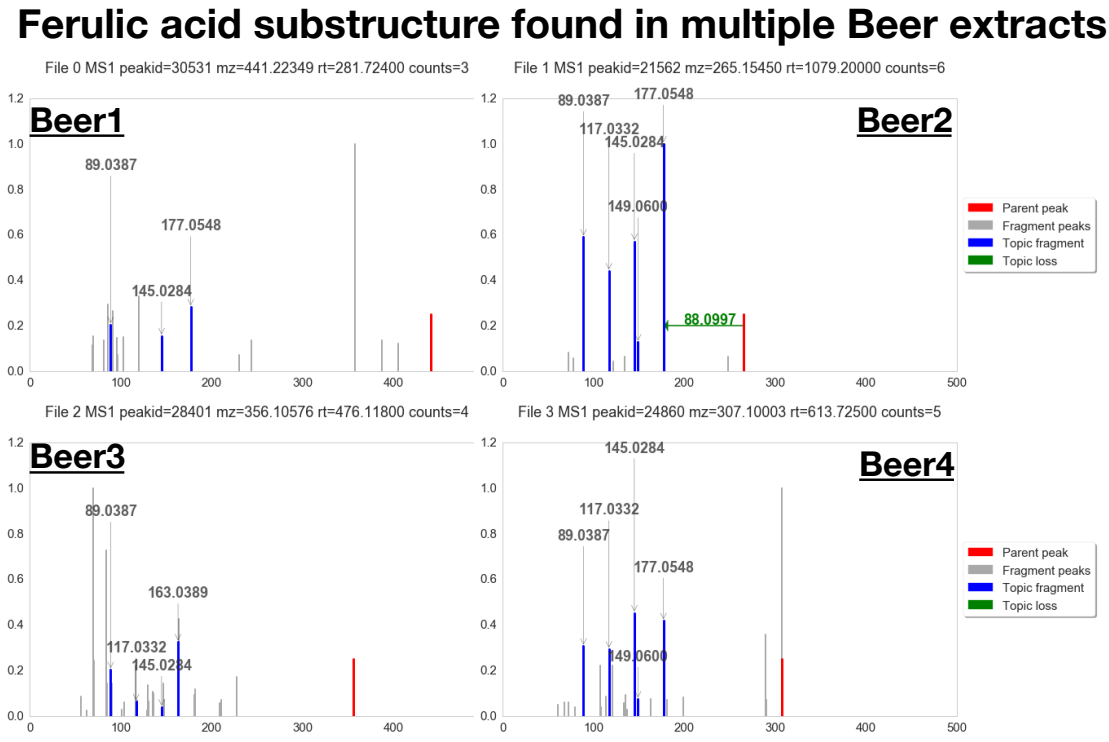


Figure 7.11: Fragmentation spectra from different Beer extracts found by multi-file LDA to contain the same Mass2Motif (M2M˙17) characterised as the ferulic acid substructure.

From each posterior sample, we can also obtain the updated $\boldsymbol{\alpha}^f$ for the different Mass2Motif across all files. As $\boldsymbol{\alpha}^f$ is the asymmetric parameter that serves as the pseudo-count in the Dirichlet-Multinomial distribution of spectra-to-Mass2Motifs, a high value of $\boldsymbol{\alpha}^f_k$ for a particular k means that a specific Mass2Motif is more likely for each spectra in file f. Figure 7.12 shows the plot of posterior alpha values for the Mass2Motifs characterised as the ferulic acid, histidine and leucine substructures. Inspections of the comparisons in Figure 7.12 may lead to interesting biological hypothesis that explains e.g. why the ferulic acid substructure is more likely for the spectra in the third beer file compared to the others.

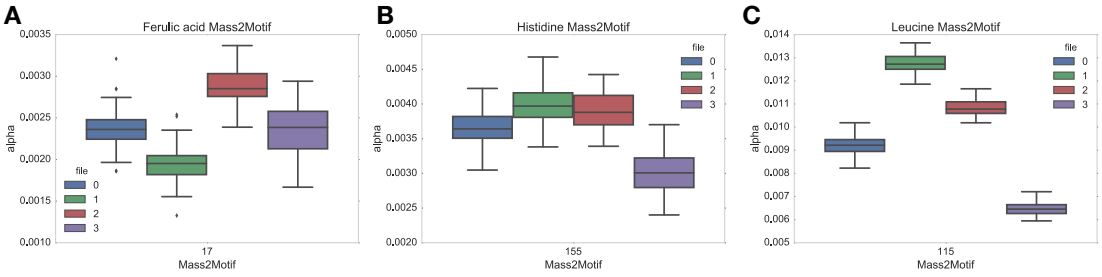


Figure 7.12: Posterior alpha values for the **A)** ferulic acid, **B)** histidine and **C)** leucine Mass2Motifs across the different beer files.

7.8 Conclusion

We have introduced MS2LDA, a pipeline that simplifies fragmentation data by exploiting the parallels between MS fragmentation data and text documents. The pipeline performs all steps required in the analysis: the preparation of a co-occurrence matrix of fragment and loss features in fragmentation spectra, the LDA analysis, and the graphical visualization of the resulting output. Evaluation of the workflow on beer extracts result in numerous informative patterns of concurrent mass fragmental and neutral loss, termed Mass2Motifs, which we could annotate as biochemically-relevant substructures. The MS2LDA approach is markedly different from other advanced spectral analysis tools as multiple Mass2Motifs can be associated with one metabolite, and determination of the key mass fragments or neutral losses that are part of a conserved structural motif is unsupervised. The application of LDA to modelling the fragmentation spectra produced by mass spectrometry instrument is exhaustively explored in this chapter. We have shown how spectra comprise of multiple substructures which can be explained by characterised Mass2Motifs. Through comparison to Molecular Networking, we demonstrated through examples how MS2LDA allows us to explain parts of a spectrum, producing a better functional annotation in contrast to spectral clustering where a spectrum can only be placed in one cluster. The differential analysis of parent ions having fragments sharing Mass2Motifs introduces the possibility of assessing changes in the expression levels of metabolites — sharing substructures explained by a characterised Mass2Motif — despite the identities of the metabolites unknown. This is particularly useful in the case of untargeted metabolomics experiments.

As future work, we envision developing a larger library of characterised Mass2Motifs from data sets produced on a diverse range of analytical platforms and different sample types. An extension of the standard LDA model, in form of the multi-file LDA model, is proposed to handle Mass2Motif inference from multiple data sets, and conditioned on having an efficient implementation, such a model can be used in large-scale clinical and metabolomic studies. The prior information on which candidate Mass2Motifs an MS2 spectrum might include could also be incorporated into the MS2LDA workflow, resulting in a semi-supervised model. A challenge to this approach in in dealing with different features due to the fact that mass spectrometry instruments have varying accuracy and therefore require different binning thresholds. One possible solution is define a common space of chemical vocabulary; rather than using binned fragment and loss features; a Mass2Motif can now be defined as the distribution over chemical formulae words. Such an approach is hampered by the fact that *de novo* elemental formula assignment itself is a difficult problem, with large uncertainties as to the correctness of annotated formulae of a fragment or loss feature. A probabilistic model of formula annotation that can offers confidence values on the formulae annotation of a fragment or loss feature will be useful in this scenario.

Other LDA-based techniques developed for text (e.g. hierarchical LDA [101]) are also likely to offer benefits in this domain as Mass2Motifs can now be defined in a hierarchy. For instance, generic patterns such as the loss of CO2 may lie at the top of the hierarchy of Mass2Motifs, while the more specific Mass2Motifs are formed at the bottom. It is anticipated that visualisation and the meaningful presentation of inference results will be a challenging task in such a model. We anticipate that MS2LDA to be particularly useful in research areas such as clinical metabolomics, pharmacometabolomics, environmental analysis, natural products research and nutritional metabolomics, as it can quickly and in an unsupervised manner recognize substructure patterns related to drugs, pollutants, and food-derived molecules, respectively.

Chapter 8

Conclusion

Note:*[About 5 pages?]*

8.1 Summary of Contributions

8.2 Future Work

8.3 Summary and Conclusions

Appendix A

An Appendix

This is an appendix.

Bibliography

[1] T.-H. Tsai, M. G. Tadesse, C. Di Poto, L. K. Pannell, Y. Mechref, Y. Wang, and H. W. Ressom, "Multi-profile Bayesian alignment model for LC-MS data analysis with integration of internal standards." *Bioinformatics*, vol. 29, no. 21, pp. 2774–80, 2013.

[2] T. S. Lee, Y. S. Ho, H. C. Yeo, J. P. Y. Lin, and D.-Y. Lee, "Precursor mass prediction by clustering ionization products in LC-MS-based metabolomics," *Metabolomics*, vol. 9, no. 6, pp. 1301–1310, apr 2013.

[3] E. de Hoffmann and V. Stroobant, *Mass spectrometry: Principles and applications*, 3rd ed., L. John Wiley & Sons, Ed., West Sussex, England, 2007.

[4] J. H. Gross, *Mass spectrometry: a textbook*. Springer Science & Business Media, 2006.

[5] D. A. Megger, T. Bracht, H. E. Meyer, and B. Sitek, "Label-free quantification in clinical proteomics," *Biochimica et Biophysica Acta - Proteins and Proteomics*, vol. 1834, no. 8, pp. 1581–1590, 2013.

[6] M. Sandin, J. Teleman, J. Malmström, and F. Levander, "Data processing methods and quality control strategies for label-free LC-MS protein quantification." *Biochimica et biophysica acta*, vol. 1844, no. 1 Pt A, pp. 29–41, 2014. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S1570963913001398

[7] R. Smith, A. D. Mathis, D. Ventura, and J. T. Prince, "Proteomics, lipidomics, metabolomics: a mass spectrometry tutorial from a computer scientist's point of view," *BMC Bioinformatics*, vol. 15, no. Suppl 7, p. S9, 2014. [Online]. Available: http://www.biomedcentral.com/1471-2105/15/S7/S9

[8] S. Castillo, P. Gopalacharyulu, L. Yetukuri, and M. Orešič, "Algorithms and tools for the preprocessing of LC-MS metabolomics data," *Chemometrics and Intelligent Laboratory Systems*, vol. 108, no. 1, pp. 23–32, aug 2011. [Online]. Available: http://linkinghub.elsevier.com/retrieve/pii/S0169743911000608

[9] J. Xiao, B. Zhou, and H. Ressom, "Metabolite identification and quantitation in LC-MS/MS-based metabolomics," *TrAC Trends in Analytical Chemistry*, pp. 1–14, 2012. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0165993611003165

[10] H. G. Gika, G. A. Theodoridis, R. S. Plumb, and I. D. Wilson, "Current practice of liquid chromatography-mass spectrometry in metabolomics and metabonomics," *Journal of Pharmaceutical and Biomedical Analysis*, vol. 87, no. March 2016, pp. 12–25, 2014.

[11] M. L. Metzker, "Sequencing technologiesthe next generation," *Nature reviews genetics*, vol. 11, no. 1, pp. 31–46, 2010.

[12] M. Mann and O. N. Jensen, "Proteomic analysis of post-translational modifications," *Nature biotechnology*, vol. 21, no. 3, pp. 255–261, 2003.

[13] M. Katajamaa and M. Oresic, "Data processing for mass spectrometry-based metabolomics." *Journal of chromatography. A*, vol. 1158, no. 1-2, pp. 318–28, jul 2007. [Online]. Available: http://www.ncbi.nlm.nih.gov/pubmed/17466315

[14] T. M. Annesley, "Ion suppression in mass spectrometry." *Clinical chemistry*, vol. 49, no. 7, pp. 1041–4, jul 2003. [Online]. Available: http://www.ncbi.nlm.nih.gov/pubmed/12816898

[15] "A common open representation of mass spectrometry data and its application to proteomics research." *Nature biotechnology*, vol. 22, no. 11, pp. 1459–66, nov 2004. [Online]. Available: http://www.ncbi.nlm.nih.gov/pubmed/15529173

[16] J. Zhang, E. Gonzalez, T. Hestilow, W. Haskins, and Y. Huang, "Review of peak detection algorithms in liquid-chromatography-mass spectrometry." *Current genomics*, vol. 10, no. 6, pp. 388–401, sep 2009. [Online]. Available: http://www.pubmedcentral.nih.gov/articlerender.fcgi?artid=2766790{&}tool=pmcentrez{&}rendertype=abstract

[17] B. O. Keller, J. Sui, A. B. Young, and R. M. Whittal, "Interferences and contaminants encountered in modern mass spectrometry." *Analytica chimica acta*, vol. 627, no. 1, pp. 71–81, oct 2008. [Online]. Available: http://www.ncbi.nlm.nih.gov/pubmed/18790129

[18] R. Smith, D. Ventura, and J. T. Prince, "{LC}-{MS} alignment in theory and practice: a comprehensive algorithmic review," *Briefings in Bioinformatics*, 2013.

[19] H. Sakoe and S. Chiba, "Dynamic programming algorithm optimization for spoken word recognition," *IEEE Transactions on Acoustics, Speech, and Signal Processing*, vol. 26, no. 1, pp. 43–49, feb 1978. [Online]. Available: http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=1163055

[20] N.-P. V. Nielsen, J. M. Carstensen, and J. Smedsgaard, "Aligning of single and multiple wavelength chromatographic profiles for chemometric data analysis using correlation optimised warping," *Journal of Chromatography A*, vol. 805, no. 1-2, pp. 17–35, may 1998. [Online]. Available: http://linkinghub.elsevier.com/retrieve/pii/S0021967398000211

[21] P. H. C. Eilers, "Parametric time warping." *Analytical chemistry*, vol. 76, no. 2, pp. 404–11, jan 2004. [Online]. Available: http://www.ncbi.nlm.nih.gov/pubmed/14719890

[22] C. Christin, A. K. Smilde, H. C. J. Hoefsloot, F. Suits, R. Bischoff, and P. L. Horvatovich, "Optimized time alignment algorithm for LC-MS data: correlation optimized warping using component detection algorithm-selected mass chromatograms." *Analytical chemistry*, vol. 80, no. 18, pp. 7012–21, sep 2008. [Online]. Available: http://www.ncbi.nlm.nih.gov/pubmed/18715018

[23] E. Lange, R. Tautenhahn, S. Neumann, and C. Gröpl, "Critical assessment of alignment procedures for {LC}-{MS} proteomics and metabolomics measurements," *BMC Bioinformatics*, vol. 9, p. 375, 2008.

[24] T. Pluskal, S. Castillo, A. Villar-Briones, and M. Oresic, "MZmine 2: modular framework for processing, visualizing, and analyzing mass spectrometry-based molecular profile data." *BMC bioinformatics*, vol. 11, no. 1, p. 395, jan 2010.

[25] N. Hoffmann, M. Keck, and H. Neuweger, "Combining peak-and chromatogram-based retention time alignment algorithms for multiple chromatography-mass spectrometry datasets," *BMC bioinformatics*, vol. 13, p. 214, 2012. [Online]. Available: http://www.biomedcentral.com/1471-2105/13/214/

[26] R. Ballardini, M. Benevento, and G. Arrigoni, "MassUntangler: A novel alignment tool for label-free liquid chromatography-mass spectrometry proteomic data," ... *of Chromatography A*, vol. 1218, no. 49, pp. 8859–68, dec 2011. [Online]. Available: http://www.ncbi.nlm.nih.gov/pubmed/21783198http://www.sciencedirect.com/science/article/pii/S0021967311008776

[27] B. Voss, M. Hanselmann, B. Y. Renard, M. S. Lindner, U. Köthe, M. Kirchner, and F. a. Hamprecht, "SIMA: simultaneous multiple alignment of LC/MS peak lists." *Bioinformatics (Oxford, England)*, vol. 27, no. 7, pp. 987–93, apr 2011.

[28] a. L. Duran, J. Yang, L. Wang, and L. W. Sumner, "Metabolomics spectral formatting, alignment and conversion tools (MSFACTs)," *Bioinformatics*, vol. 19, no. 17, pp. 2283–2293, nov 2003. [Online]. Available: http://bioinformatics.oxfordjournals.org/cgi/doi/10.1093/bioinformatics/btg315

[29] J. Wang and H. Lam, "Graph-based peak alignment algorithms for multiple liquid chromatography-mass spectrometry datasets." *Bioinformatics (Oxford, England)*, vol. 29, no. 19, pp. 2469–2476, aug 2013. [Online]. Available: http://www.ncbi.nlm.nih.gov/pubmed/23904508

[30] H. Lin, L. He, and B. Ma, "A Combinatorial Approach to the Peptide Feature Matching Problem for Label-Free Quantification." *Bioinformatics (Oxford, England)*, pp. 1–7, may 2013. [Online]. Available: http://www.ncbi.nlm.nih.gov/pubmed/23665772

[31] C. a. Smith, E. J. Want, G. O'Maille, R. Abagyan, and G. Siuzdak, "XCMS: processing mass spectrometry data for metabolite profiling using nonlinear peak alignment, matching, and identification." *Analytical chemistry*, vol. 78, no. 3, pp. 779–87, feb 2006. [Online]. Available: http://www.ncbi.nlm.nih.gov/pubmed/16448051

[32] "Retention time alignment algorithms for LC/MS data must consider non-linear shifts." *Bioinformatics (Oxford, England)*, vol. 25, no. 6, pp. 758–64, mar 2009. [Online]. Available: http://www.ncbi.nlm.nih.gov/pubmed/19176558

[33] R. C. H. De Vos, S. Moco, A. Lommen, J. J. B. Keurentjes, R. J. Bino, and R. D. Hall, "Untargeted large-scale plant metabolomics using liquid chromatography coupled to mass spectrometry." *Nat. Protoc.*, vol. 2, no. 4, pp. 778–791, jan 2007.

[34] D. J. Creek, A. Jankevics, R. Breitling, D. G. Watson, M. P. Barrett, and K. E. V. Burgess, "Toward global metabolomics analysis with hydrophilic interaction liquid chromatography–mass spectrometry: improved metabolite identification by retention time prediction," *Analytical Chemistry*, vol. 83, no. 22, pp. 8703–8710, 2011.

[35] A. Chawade, M. Sandin, J. Teleman, J. Malmström, and F. Levander, "Data processing has major impact on the outcome of quantitative label-free LC-MS analysis." *Journal of proteome research*, vol. 14, no. 2, pp. 676–87, 2015. [Online]. Available: http://dx.doi.org/10.1021/pr500665j

[36] W. B. Dunn, D. Broadhurst, P. Begley, E. Zelena, S. Francis-McIntyre, N. Anderson, M. Brown, J. D. Knowles, A. Halsall, J. N. Haselden *et al.*, "Procedures for large-scale metabolic profiling of serum and plasma using gas chromatography and liquid chromatography coupled to mass spectrometry," *Nature protocols*, vol. 6, no. 7, pp. 1060–1083, 2011.

[37] W. Dunn, A. Erban, R. Weber, and D. Creek, "Mass appeal: metabolite identification in mass spectrometry-focused untargeted metabolomics," *Metabolomics*, 2012. [Online]. Available: http://www.springerlink.com/index/0718581530254PG6.pdf

[38] T. Kind and O. Fiehn, "Metabolomic database annotations via query of elemental compositions: mass accuracy is insufficient even at less than 1 ppm." *BMC bioinformatics*, vol. 7, p. 234, jan 2006. [Online]. Available: http://www.pubmedcentral.nih.gov/articlerender.fcgi?artid=1464138{&}tool=pmcentrez{&}rendertype=abstract

[39] C. Kuhl, R. Tautenhahn, C. Böttcher, 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*, vol. 84, no. 1, pp. 283–9, jan 2012. [Online]. Available: http://www.ncbi.nlm.nih.gov/pubmed/22111785

[40] R. Daly, S. Rogers, J. Wandy, A. Jankevics, K. E. V. Burgess, and R. Breitling, "MetAssign: probabilistic annotation of metabolites from LC-MS data using a Bayesian clustering approach." *Bioinformatics (Oxford, England)*, vol. 30, no. 19, pp. 2764–2771, jun 2014.

[41] J. Xia and D. S. Wishart, "MetPA: a web-based metabolomics tool for pathway analysis and visualization." *Bioinformatics (Oxford, England)*, vol. 26, no. 18, pp. 2342–4, sep 2010. [Online]. Available: http://www.ncbi.nlm.nih.gov/pubmed/20628077

[42] J. Krumsiek, K. Suhre, and T. Illig, "Gaussian graphical modeling reconstructs pathway reactions from high-throughput metabolomics data," *BMC systems ...*, 2011. [Online]. Available: http://www.biomedcentral.com/1752-0509/5/21/

[43] J. Krumsiek, K. Suhre, A. M. Evans, M. W. Mitchell, R. P. Mohney, M. V. Milburn, B. Wägele, W. Römisch-Margl, T. Illig, J. Adamski, C. Gieger, F. J. Theis, and G. Kastenmüller, "Mining the unknown: a systems approach to metabolite identification combining genetic and metabolic information." *PLoS genetics*, vol. 8, no. 10, p. e1003005, oct 2012. [Online]. Available: http://www.ncbi.nlm.nih.gov/pubmed/23093944

[44] "Genetics meets metabolomics: a genome-wide association study of metabolite profiles in human serum." *PLoS genetics*, vol. 4, no. 11, p. e1000282, nov 2008. [Online]. Available: http://www.pubmedcentral.nih.gov/articlerender.fcgi?artid=2581785{&}tool=pmcentrez{&}rendertype=abstract

[45] M. Mamas, W. B. Dunn, L. Neyses, and R. Goodacre, "The role of metabolites and metabolomics in clinically applicable biomarkers of disease," *Archives of toxicology*, vol. 85, no. 1, pp. 5–17, 2011.

[46] "Metabolomics in human nutrition: opportunities and challenges." *The American journal of clinical nutrition*, vol. 82, no. 3, pp. 497–503, sep 2005. [Online]. Available: http://www.ncbi.nlm.nih.gov/pubmed/16155259

[47] D. B. Kell, "Systems biology, metabolic modelling and metabolomics in drug discovery and development." *Drug discovery today*, vol. 11, no. 23-24, pp. 1085–92, dec 2006. [Online]. Available: http://www.ncbi.nlm.nih.gov/pubmed/17129827

[48] C. Rasmussen, "The infinite Gaussian mixture model," in *Advances in Neural Information Processing Systems 12*. MIT Press, 2000, pp. 554–560.

[49] R. Smith, J. T. Prince, and D. Ventura, "A coherent mathematical characterization of isotope trace extraction , isotopic envelope extraction , and LC-MS correspondence," *BMC Bioinformatics*, vol. 16, no. Suppl 7, p. S1, 2015. [Online]. Available: http://www.biomedcentral.com/1471-2105/16/S7/S1

[50] D. Gusfield and R. Irving, *The stable marriage problem: structure and algorithms*, 1989.

[51] H. Kuhn, "The Hungarian method for the assignment problem," *Naval research logistics quarterly*, 1955. [Online]. Available: http://onlinelibrary.wiley.com/doi/10.1002/nav.3800020109/full

[52] R. Duan, S. Pettie, and H. Su, "Scaling algorithms for approximate and exact maximum weight matching," *arXiv preprint arXiv:1112.0790*, pp. 1–36, 2011. [Online]. Available: http://arxiv.org/abs/1112.0790

[53] R. Smith, D. Ventura, and J. T. Prince, "Novel algorithms and the benefits of comparative validation." *Bioinformatics (Oxford, England)*, vol. 29, no. 12, pp. 1583–5, jun 2013. [Online]. Available: http://www.ncbi.nlm.nih.gov/pubmed/23589651

[54] R. a. Scheltema, A. Jankevics, R. C. Jansen, M. a. Swertz, and R. Breitling, "PeakML/mzMatch: a file format, Java library, R library, and tool-chain for mass spectrometry data analysis." *Analytical Chemistry*, vol. 83, no. 7, pp. 2786–93, 2011.

[55] S. Rogers, R. Daly, and R. Breitling, "Mixture model clustering for peak filtering in metabolomics," in *Ninth International Workshop on Computational Systems Biology, WCSB 2012, June 4-6, Ulm, Germany*, 2012, p. 71.

[56] E. Melamud, L. Vastag, and J. D. Rabinowitz, "Metabolomic analysis and visualization engine for LC-MS data." *Analytical chemistry*, vol. 82, no. 23, pp. 9818–26, dec 2010. [Online]. Available: http://www.ncbi.nlm.nih.gov/pubmed/21049934

[57] L. Brodsky, A. Moussaieff, N. Shahaf, A. Aharoni, and I. Rogachev, "Evaluation of peak picking quality in LC-MS metabolomics data." *Analytical chemistry*, vol. 82, no. 22, pp. 9177–87, nov 2010. [Online]. Available: http://www.ncbi.nlm.nih.gov/pubmed/20977194

[58] G. Landan and D. Graur, "Characterization of pairwise and multiple sequence alignment errors." *Gene*, vol. 441, no. 1-2, pp. 141–7, Jul. 2009.

[59] J. D. Thompson, D. G. Higgins, and T. J. Gibson, "CLUSTAL W: improving the sensitivity of progressive multiple sequence alignment through sequence weighting, position-specific gap penalties and weight matrix choice." *Nucleic Acids Res.*, vol. 22, no. 22, pp. 4673–80, Nov. 1994.

[60] C. Notredame, D. G. Higgins, and J. Heringa, "T-Coffee: A novel method for fast and accurate multiple sequence alignment." *J. Mol. Biol.*, vol. 302, no. 1, pp. 205–17, Sep. 2000.

[61] O. Penn, E. Privman, G. Landan, D. Graur, and T. Pupko, "An alignment confidence score capturing robustness to guide tree uncertainty." *Mol. Biol. Evol.*, vol. 27, no. 8, pp. 1759–67, Aug. 2010.

[62] B. D. Redelings and M. a. Suchard, "Joint Bayesian estimation of alignment and phylogeny." *Syst. Biol.*, vol. 54, no. 3, pp. 401–18, Jun. 2005.

[63] R. K. Bradley, A. Roberts, M. Smoot, S. Juvekar, J. Do, C. Dewey, I. Holmes, and L. Pachter, "Fast statistical alignment." *PLoS Comput. Biol.*, vol. 5, no. 5, p. e1000392, May 2009.

[64] J. Listgarten, R. M. Neal, S. T. Roweis, and A. Emili, "Multiple alignment of continuous time series," in *Advances in neural information processing systems*, 2004, pp. 817–824.

[65] X. Kong and C. Reilly, "A Bayesian approach to the alignment of mass spectra." *Bioinformatics*, vol. 25, no. 24, pp. 3213–20, Dec. 2009.

[66] B. Fischer, J. Grossmann, V. Roth, W. Gruissem, S. Baginsky, and J. M. Buhmann, "Semi-supervised LC/MS alignment for differential proteomics." *Bioinformatics*, vol. 22, no. 14, pp. e132–40, Jul. 2006.

[67] J. Jeong, X. Shi, X. Zhang, S. Kim, and C. Shen, "Model-based peak alignment of metabolomic profiling from comprehensive two-dimensional gas chromatography mass spectrometry." *BMC Bioinformatics*, vol. 13, no. 1, p. 27, Jan. 2012.

[68] M. Ghanat Bari, X. Ma, and J. Zhang, "PeakLink: a new peptide peak linking method in LC-MS/MS using wavelet and SVM." *Bioinformatics*, vol. 30, no. 17, pp. 2464–70, Sep. 2014.

[69] M. Sandin, A. Ali, K. Hansson, O. Månsson, and E. Andreasson, "An Adaptive Alignment Algorithm for Quality-controlled Label-free LC-MS," *Mol Cell Proteomics*, pp. 1407–1420, 2013.

[70] D. P. De Souza, E. C. Saunders, M. J. McConville, and V. a. Likić, "Progressive peak clustering in GC-MS Metabolomic experiments applied to Leishmania parasites." *Bioinformatics*, vol. 22, no. 11, pp. 1391–6, Jun. 2006.

[71] Y. W. Teh, M. I. Jordan, M. J. Beal, and D. M. Blei, "Hierarchical dirichlet processes," *Journal of the american statistical association*, vol. 101, no. 476, 2006.

[72] M. Kotera, M. Hirakawa, T. Tokimatsu, S. Goto, and M. Kanehisa, "The kegg databases and tools facilitating omics analysis: latest developments involving human diseases and pharmaceuticals," *Next Generation Microarray Bioinformatics: Methods and Protocols*, pp. 19–39, 2012.

[73] E. E. Bolton, Y. Wang, P. A. Thiessen, and S. H. Bryant, "Pubchem: integrated platform of small molecules and biological activities," *Annual reports in computational chemistry*, vol. 4, pp. 217–241, 2008.

[74] T. Kind and O. Fiehn, "Seven Golden Rules for heuristic filtering of molecular formulas obtained by accurate mass spectrometry." *BMC bioinformatics*, vol. 8, p. 105, jan 2007. [Online]. Available: http://www.pubmedcentral.nih.gov/articlerender.fcgi?artid=1851972{&}tool=pmcentrez{&}rendertype=abstract

[75] C. a. Smith, G. O'Maille, E. J. Want, C. Qin, S. a. Trauger, T. R. Brandon, D. E. Custodio, R. Abagyan, and G. Siuzdak, "METLIN: a metabolite mass spectral database." *Therapeutic drug monitoring*, vol. 27, no. 6, pp. 747–51, dec 2005. [Online]. Available: http://www.ncbi.nlm.nih.gov/pubmed/16404815

[76] H. E. Pence and A. Williams, "Chemspider: an online chemical information resource," *Journal of Chemical Education*, vol. 87, no. 11, pp. 1123–1124, 2010.

[77] H. Horai, M. Arita, S. Kanaya, Y. Nihei, T. Ikeda, K. Suwa, Y. Ojima, K. Tanaka, S. Tanaka, K. Aoshima *et al.*, "Massbank: a public repository for sharing mass spectral

data for life sciences," *Journal of mass spectrometry*, vol. 45, no. 7, pp. 703–714, 2010.

[78] R. R. da Silva, P. C. Dorrestein, and R. A. Quinn, "Illuminating the dark matter in metabolomics," *Proceedings of the National Academy of Sciences*, vol. 112, no. 41, p. 201516878, 2015. [Online]. Available: http://www.pnas.org/content/early/2015/09/30/1516878112.extract

[79] K. Varmuza and W. Werther, "Mass Spectral Classifiers for Supporting Systematic Structure Elucidation," *Journal of Chemical Information and Modeling*, vol. 36, no. 2, pp. 323–333, 1996. [Online]. Available: http://pubs.acs.org/cgi-bin/doilookup/?10.1021/ci9501406

[80] J. Hummel, N. Strehmel, J. Selbig, D. Walther, and J. Kopka, "Decision tree supported substructure prediction of metabolites from GC-MS profiles," *Metabolomics*, vol. 6, no. 2, pp. 322–333, 2010.

[81] M. Heinonen, H. Shen, N. Zamboni, and J. Rousu, "Metabolite identification and molecular fingerprint prediction via machine learning," *cs.helsinki.fi*, pp. 1–8, 2012. [Online]. Available: http://www.cs.helsinki.fi/u/mqheinon/mlsb{_}paper{_}2012{_}preprint.pdf

[82] K. Dührkop, H. Shen, M. Meusel, J. Rousu, and S. Böcker, "Searching molecular structure databases with tandem mass spectra using CSI:FingerID," *Proceedings of the National Academy of Sciences*, p. 201509788, 2015. [Online]. Available: http://www.pnas.org/lookup/doi/10.1073/pnas.1509788112

[83] J. Y. Yang, L. M. Sanchez, C. M. Rath, X. Liu, P. D. Boudreau, N. Bruns, E. Glukhov, A. Wodtke, R. de Felicio, A. Fenner *et al.*, "Molecular networking as a dereplication strategy," *Journal of natural products*, vol. 76, no. 9, pp. 1686–1699, 2013.

[84] D. D. Nguyen, C.-H. Wu, W. J. Moree, A. Lamsa, M. H. Medema, X. Zhao, R. G. Gavilan, M. Aparicio, L. Atencio, C. Jackson *et al.*, "Ms/ms networking guided analysis of molecule and gene cluster families," *Proceedings of the National Academy of Sciences*, vol. 110, no. 28, pp. E2611–E2620, 2013.

[85] J. J. Van Der Hooft, S. Padmanabhan, K. E. Burgess, and M. P. Barrett, "Urinary antihypertensive drug metabolite screening using molecular networking coupled to high-resolution mass spectrometry fragmentation," *Metabolomics*, 2016.

[86] Y. Ma, T. Kind, D. Yang, C. Leon, and O. Fiehn, "Ms2analyzer: A software for small molecule substructure annotations from accurate tandem mass spectra," *Analytical chemistry*, vol. 86, no. 21, pp. 10 724–10 731, 2014.

[87] D. L. Sweeney, "A Data Structure for Rapid Mass Spectral Searching," *Mass Spectrometry*, vol. 3, no. Special_Issue_2, pp. S0035–S0035, 2014. [Online]. Available: http://jlc.jst.go.jp/DN/JST.JSTAGE/massspectrometry/S0035?lang=en{&}from=CrossRef{&}type=abstract

[88] D. R. Scott, "Pattern recognition/ expert system for identification of toxic compounds from low resolution mass spectra *," vol. 23, pp. 351–364, 1994.

[89] X. Chen, X. Hu, X. Shen, and G. Rosen, "Probabilistic topic modeling for genomic data interpretation," in *Bioinformatics and Biomedicine (BIBM), 2010 IEEE International Conference on*. IEEE, 2010, pp. 149–152.

[90] R. Zhang, Z. Cheng, J. Guan, and S. Zhou, "Exploiting topic modeling to boost metagenomic reads binning," *BMC bioinformatics*, vol. 16, no. Suppl 5, p. S2, 2015.

[91] S. Rogers, M. Girolami, C. Campbell, and R. Breitling, "The latent process decomposition of cdna microarray data sets," *IEEE/ACM Transactions on Computational Biology and Bioinformatics (TCBB)*, vol. 2, no. 2, pp. 143–156, 2005.

[92] S. Böcker, M. C. Letzel, Z. Lipták, and A. Pervukhin, "SIRIUS: decomposing isotope patterns for metabolite identification." *Bioinformatics (Oxford, England)*, vol. 25, no. 2, pp. 218–24, jan 2009. [Online]. Available: http://www.pubmedcentral.nih.gov/articlerender.fcgi?artid=2639009{&}tool=pmcentrez{&}rendertype=abstract

[93] M. a. Stravs, E. L. Schymanski, H. P. Singer, and J. Hollender, "Automatic recalibration and processing of tandem mass spectra using formula annotation," *Journal of Mass Spectrometry*, vol. 48, no. 1, pp. 89–99, 2013.

[94] C. Sievert and K. Shirley, "LDAvis: A method for visualizing and interpreting topics," *Proceedings of the Workshop on Interactive Language Learning, Visualization, and Interfaces*, pp. 63–70, 2014. [Online]. Available: http://www.aclweb.org/anthology/W/W14/W14-3110

[95] S. Bocker and Z. Liptak, "A Fast and Simple Algorithm for the Money Changing Problem," *Algorithmica*, vol. 48, no. 4, pp. 413–432, jul 2007. [Online]. Available: http://www.springerlink.com/index/10.1007/s00453-007-0162-8

[96] H. M. Wallach, I. Murray, R. Salakhutdinov, and D. Mimno, "Evaluation methods for topic models," in *Proceedings of the 26th Annual International Conference on Machine Learning*. ACM, 2009, pp. 1105–1112.

[97] T. L. Griffiths and M. Steyvers, "Finding scientific topics." *Proceedings of the National Academy of Sciences of the United States of America*, vol. 101 Suppl, pp. 5228–5235, 2004.

[98] J. Tomfohr, J. Lu, and T. B. Kepler, "Pathway level analysis of gene expression using singular value decomposition," *BMC bioinformatics*, vol. 6, no. 1, p. 1, 2005.

[99] A. L. Tarca, G. Bhatti, and R. Romero, "A comparison of gene set analysis methods in terms of sensitivity, prioritization and specificity," *PloS one*, vol. 8, no. 11, p. e79217, 2013.

[100] T. P. Minka, "Estimating a Dirichlet distribution," *Annals of Physics*, vol. 2000, no. 8, pp. 1–13, 2003. [Online]. Available: http://research.microsoft.com/en-us/um/people/minka/papers/dirichlet/minka-dirichlet.pdf

[101] D. Griffiths and M. Tenenbaum, "Hierarchical topic models and the nested chinese restaurant process," *Advances in neural information processing systems*, vol. 16, p. 17, 2004.