universität wien

MASTERARBEIT / MASTER'S THESIS

Titel der Masterarbeit / Title of the Master's Thesis

„Uncovering High Resolution Mass Spectrometry Patterns through Audio Fingerprinting and Periodicity Mining Algorithms: An Exploratory Analysis "

verfasst von / submitted by

Theresa Fruhwürth, BSc

angestrebter akademischer Grad / in partial fulfilment of the requirements for the degree of

Master of Science, (MSc)

Wien, 2017 / Vienna, 2017

Studienkennzahl lt. Studienblatt / degree programme code as it appears on the student record sheet:	A 066 910
Studienrichtung lt. Studienblatt / degree programme as it appears on the student record sheet:	Computational Science
Betreut von / Supervisor:	Univ. Prof. Dipl.-Inform. Univ. Dr. Claudia Plant

Contents

List of Figures

List of Tables

Acknowledgements

I would like to thank both Dr. Basem Kanawati and Professor Dr. Philippe Schmitt-Kopplin at the Research Unit Analytical BioGeoChemistry at Helmholtz Zentrum Munich for their support and input throughout this work and the providing of data sets and chemical knowledge crucial for the development of this thesis. Furthermore I would like to thank Professor Dipl.-Inform.Univ. Dr. Claudia Plant for her advice on data mining techniques and her understanding for and of the computer scientific effort that went into this work. She helped me a lot to gain perspective on my work from a different and more positive angle that is sometimes difficult to achieve when things go wrong.

First of all I want to thank my family. I know I have taken a long winding part to get where I am. But you made it possible for me to get to work on something I truly care for and I am interested in, in a way that goes beyond my professional and material needs. My mother Barbara Fruhwürth who always believed in me and showed me how to grow and succeed as a professional woman. My father Richard Fruhwürth who has contributed to my growth maybe inadvertently on a very personal level as well. I would also like to thank my two little sisters Stephanie and Johanna, who have chosen different paths and went on to be successful and independent women in their own regards already way before me. I have taken so long I also want to thank my grandparents, who probably didn't believe I would finish this at all any more. But you see, all is well that ends well. You all made it possible for me that I go into life with a positive outlook and a certainty that I will always have something interesting to work on.
I also want to thank Professor Anastasio at the University of Illinois. He was the one who introduced me to programming and who made me discover something creative that helped express all the different pieces of knowledge I was fortunate to gather over my University career so far.

Chapter 1

Introduction

This thesis will investigate algorithms to find patterns or structure in mass spectral data that has remained hidden at present. This is done by applying new algorithms from data mining and related disciplines. In order to understand the current challenges in mass spectral data analysis, it is important to understand some of the chemical background as well as some limitations of current approaches. This section aims at providing a brief introduction on the chemical specifics and the evolution of periodicity mining algorithms for time series that are important to fully understand the objectives of this thesis.

1.1 Chemical Background

1.1.1 Mass Spectrometry

Mass spectrometry plays a vital role in chemistry, physics and biology in tasks ranging from protein analysis to atomic physics measurements (Domon and Aebersold, 2006; Levine et al., 1988). It is used to identify complex mixtures and their constituent molecular and atomic species. However mass spectral data is still difficult to understand because of the complexity of the data. Mass spectrometry is a technique that ionizes chemical species which are moved by an external magnetic field, and subsequently sorts these ions based on their mass to charge ratio (m/z), measuring the masses of constituents present in a sample of a chemical substance (De Hoffmann and Stroobant, 2007) (pp. 85-175).

High-precision frequency measurements such as the ones obtained by The Fourier Transform Ion Cyclotron Resonance Mass Spectrometer (FTICR-MS) provides data that overcomes effects of averaging that is detrimental to the capacity to separate the constituents when using different methods. This advantage is due to the FTICR-MS being capable of high mass range as well as mass resolution. Both of these qualities are achieved through measurements of characteristic cyclotron frequencies with an accuracy of 15 digits (Hertkorn et al., 2007). If this accurate time domain transient is translated into the mass domain it yields an accuracy that provides scientists with the opportunity to be able to determine the elemental composition of chemical substances with a mass resolution in the ppm range (Amster et al., 1996). The precise mass of a detected ion is the sum of the masses of its constituent atoms and hence mass determination can lead to the discovery of sum formulas (Kendrick, 1963). Data analysis tools such as the DataAnalysis software (Bruker Daltonics GmbH, Bremen, Germany) help to calibrate such

spectra and annotate some of the peaks exhibited in the spectrum with sum formulas describing the chemical species present in a sample. However most of these annotations are not unequivocal, thus it remains to rely on the experienced analytical chemist to provide a definite annotation based on additional knowledge if at all possible. Nonetheless the resolution is often times sufficient to distinguish between ions having nearly the same m/z ratio but different chemical composition, and analytic methods explained in Section 1.2 often times help to resolve ambiguous peaks. High resolution methods such as the FTICR-MS produce very information rich data. However the full potential of the FTICR can not be reaped as there is still a lag in development of methodologies for exact assignment of chemical entities and the description of the data in both its domains i.e. the frequency and the mass domain (Easterling et al., 1999). According to Barner-Kowollik et al. (2012) (pp. 241) "mass spectral peaks are defined as statistically significant excursions in the spectrum intensity from its baseline as a result of ions of a given mass to charge ratio being detected by the instrument." However usually the data will contain noise due to chemical sources such as for example improper separation in mass or even electronic sources stemming from the FTICR device itself (Barner-Kowollik et al., 2012) (pp. 241).

Usually molecular ions undergo fragmentations that often happen in multiple steps such that the first fragmentation step is not the ultimate fate of the ion. The primary product ion stemming from the parent ion produced during the first fragmentation step might undergo fragmentation repeatedly and all these fragments are recorded through their characteristic cyclotron frequencies. Recording these frequencies and subsequent translation of the time domain transient into mass space will yield a characteristic sequence of peaks indicating their mass to charge ratio on the x axis and their abundance specified as intensity on the y axis. The most intense peak in the entire spectrum is called the base peak which gets assigned an intensity of 100%. Any peaks other than the base peak are given relative values of intensity with respect to the corresponding base peak of the spectrum (De Hoffmann and Stroobant, 2007) (pp. 1-10).

The fragmented ions also referred to as products, also provide information on their precursor molecular ions. In a spectrum of a pure compound, the molecular ion appears at the highest value of m/z followed by ions containing relatively heavy isotopes which are in turn followed by ions containing lighter isotopes arranging the product ions in m/z sequence from heavy to light. However this is not as simple for more complex mixtures where relationships between the parent ion and its products need to be established through computational means such as for example the Mass difference statistics algorithm explained in Section 1.2.3 (Kunenkov et al., 2009; De Hoffmann and Stroobant, 2007) (pp.1-10). Another important concept in mass spectrometry are isotopes. These are atoms of the same type differing only in their number of neutrons. Thus, albeit they may be more difficult to identify, a characteristic pattern of peaks corresponding to a series of isotopes can be observed even in more complex mixtures indicating these characteristic fragmentation patterns of isotope series that often present themselves as nearly Gaussian shaped isotope distributions centred around the most abundant isotopes mass (Greiner et al., 1975; De Hoffmann and Stroobant, 2007) (pp. 1-10). Another important and often used concept prevalent in mass spectrometry is the nominal mass. It is the next integer floor or ceiling of the monoisotopic mass which itself is calculated using the mass of the most abundant naturally occurring isotope. In Figure 1.1 the

monoisotopic mass is the lighter mass of the isotopic pattern visible on the left side of the figure. The mass spectrum shows peaks corresponding to the isotopes and their relative abundance. It shows a typical pattern of a mass spectrum which are particularly present in spectra of natural organic matter where these patterns are often shaped like Gaussian distributions (Greiner et al., 1975) described in Section 1.1.2. The most abundantly occurring isotope is the isotope with a particular neutron number which occurs most frequently in nature and thus will show as a relatively higher peak in an isotope series (De Hoffmann and Stroobant, 2007) (pp. 1-10).

As an example given in De Hoffmann and Stroobant (2007) one can look at CH3Cl whose mass spectrum is shown in Figure 1.1 adopted from (De Hoffmann and Stroobant, 2007) (pp. 1- 10). When the mass of CH3Cl is measured in a mass spectrometer two isotopic peaks will appear at 49.992327 and 51.989365 m/z ratios and the intensity axis will indicate their respective relative abundances. The m/z ratios are derived by computing the monoisotopic mass of CH3Cl which will yield two values according to this equation $1 \times C + 3 \times H + Cl$. Plugging in the corresponding masses of the two isotopes of Cl respectively, values according to the m/z ratios will be computed. Because FTICR - MS has a resolution in the parts per million (ppm) range it is experimentally possible to distinguish between these isotopes characteristic peaks (De Hoffmann and Stroobant, 2007) (pp. 1-10).

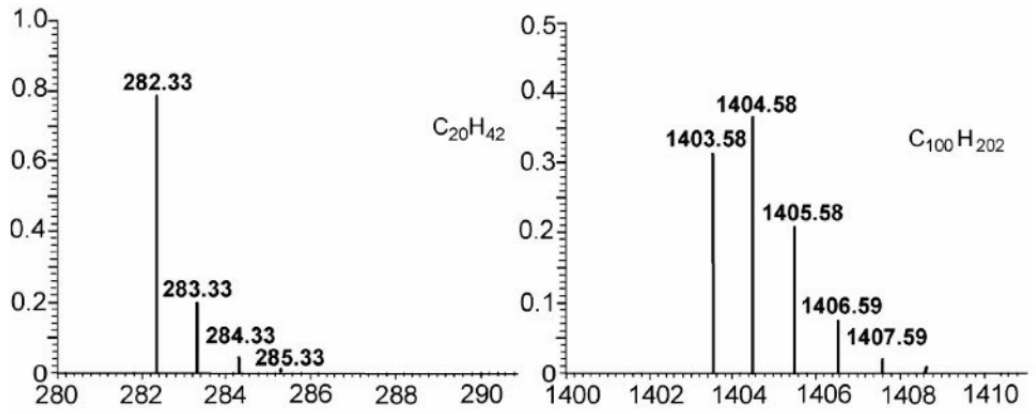

Figure 1.1: Mass spectrum of C20H40 and C100H202 by electron ionization, showing the isotopic patterns of two alkanes. The monoisotopic mass is the lighter mass of the isotopic pattern.

Periodicity in peaks can also arise because of detected polymer structures, for peptides it is expected to see a characteristic series of peaks whose masses are distributed in two series of charged ions which can be seen as signatures of their precursor molecules that exhibits periodicity stemming from properties of the amino acids masses (Hubler and Craciun, 2012). Periodicity is furthermore deemed important because they should in theory often point towards peaks that differ by a mass difference that corresponds to the mass differences between amino acid residues that show bond breaking for example by the loss of a $CH2$ group resulting in two isotopic distributions at two distinct m/z values separated by the mass of the $CH2$ group (Schlunegger, 2016; Yu et al., 2011)

(pp. 108-139). Thus periodicity has already been employed to analyse properties of mass spectra particularly in biological samples and was found to be indicative of chemically relevant peak patterns in repetitive systems. This picture of periodicity however is not so clear when it comes to non-repetitive samples such as for example Natural organic matter (NOM) samples as will be explained below.

1.1.2 Natural Organic Matter

NOM is a term that subsumes a wide range of materials that is known to be complex in nature, and is important for a number of important processes in biogeochemical context. It is found in many environments including but not limited to soil, natural water and sediments. NOM materials contain mostly carbon, hydrogen and oxygen atoms thus are referred to exist in CHO space, however they may also contain heteroatoms such as for example sulphur (Hertkorn et al., 2007) . Generally natural molecules can be classified in two groups either according to their functions or how they are synthesized. Hertkorn et al. (2007) mentions that "natural complex organic materials divide into either functional biomolecules which eventually derive from a genetic code or complex biogeochemical non-repetitive materials which are formed according to the general constraints of thermodynamics and kinetics from geochemical or ultimately biogenic molecules." (Hertkorn et al., 2007) (pp. 1312) Compounds belonging to the first group are repetitive systems which have known fragmentation signatures emerging from precursor molecules. This will result in characteristic peak patterns which makes it possible to assign them to distinct classes such as proteins, lipids and carbohydrates. These signatures come for example in the form of patterns that show the typical bond breaking in peptides at the carbonyl group (Paizs and Suhai, 2005; Hertkorn et al., 2007). Non repetitive systems on the other hand emerge from processes of biotic and abiotic degradation which result in a nearly intractable number of possible reactions and thus resulting compounds only confined by kinetic and thermodynamic laws. Hence NOM as a non repetitive system is covering a large part of the theoretically feasible molecular composition space. However because of the processes under which NOM compounds emerge they loose their signatures and emerge as an seemingly unstructured array of peaks in the mass spectrum. The problem is that NOM substances are much more abundant in their variety, than the functionally classified biomolecules which are usually well described because of known systematic classifications and known properties gathered by systematic empirical approaches (Hertkorn et al., 2007). Because of this abundance it is still difficult to provide a description of NOM space and existing definitions lack in precision. This is also true because numerical descriptions are still missing for many compounds in NOM. At the core for establishing such a model of NOM stands the empirical approach of purification. However such an approach proofs to be difficult and expensive. The empirical bottom up approach involves for example FTICR-MS measurements and and nuclear magnetic resonance (Hertkorn et al., 2007; Cook, 2004). Unfortunately even with these empirical methods it proves to be difficult to find a complete characterization of NOM compounds, foremost because of the heterogeneous origin and interaction which implies also a heterogeneity in composition (Chen et al., 2002). It follows that a more detailed description in terms of a model of NOM space is imperative. However there are as of yet no such protocols on which to isolate, separate and purify all components contained

in a NOM spectrum (Cook, 2004). Due to the abundance of distinct NOM compounds it would presumably be infeasible simply from a financial point of view to isolate and purify and thus identify all compounds using empirical approaches alone. This makes the development of novel computational methods for the analysis of already attained NOM spectra all the more interesting. Computational methods are often based on assumptions about the functional relationship between data points. Data mining offers different techniques to identify and understand often complex functional and structural relationships in scientific datasets, thus helps to uncover new information.

A common theme in NOM mass spectra are mass differences between peaks corresponding to the repeat units given in Table 1.1 as established by Kujawinski and Behn (2006). Different computational methods already exploit part of this known structure in NOM space. Often times these computational methods use the concept of mass differences. As an example we could look at heptaketide pyrone intermediate with a sum formula of C14H11O6 which is used for the calibration of mass spectral data. The loss of one carbonyl CO group would result in a sum formula of C13H11O5 and a mass difference of 27.994915. We would observe two peaks one at a value around 275.0561117 and one at 247.0611967 and could take this as evidence that a characteristic sequence of chemical substances is being traced. Similarly extrapolation using mass difference analysis allows assignment of m/z peaks occurring at higher m/z ratios in Natural organic matter (NOM) spectra. This is done by extrapolating from lower mass numbers within recurring molecular series through the means of adding repeat units or subtracting repeat units. Thus annotation of chemical species within the series can be done (Kunenkov et al., 2009). The specifics of this algorithm will be explained in Section 1.2.3. The abundance of possible chemical structures in NOM data also implies that for given peaks in a mass spectrum usually a multitude of molecules will match the same nominal mass especially in higher m/z range. These isobaric ions sum formulas incur further difficulty to annotate individual peaks in the relatively narrow mass range of NOM isolates (Reemtsma, 2009). This is due to the fact that in higher mass ranges there is a combinatorial explosion for peak annotation due to an increase in possible elemental compositions that can cause a particular peak at a specific particular m/z value. This means that a peak can be caused by a multitude of chemicals made up of C, H, O and the heteroatomic constituents N and S which are some of the known atomic constituents of NOM. However the FTICR provides sufficient mass resolution and it was shown that a mass accuracy of 0.1 milliDalton is sufficient to unambiguously detect the true elemental composition for all theoretically possible ions in NOM in a mass range up to 500 Dalton (Kunenkov et al., 2009).

Analytical methods discussed in Section 1.2 below give insight into means currently used to disambiguate and annotate peaks observed in NOM spectra .

unit name	CH2	O	H2	H2O	CO2	CO
mass	14.01565	15.994915	2.01565	18.010565	43.98983	27.994915

Table 1.1: Masses and molecular formulas of known examples for repeat units found in NOM mass spectra.

1.2 Analysis and Annotation of Mass Spectral Data

Generally speaking software often selects multiple potential sum formulas for peak annotation. Not least because of the occurrence of many isobaric substances especially in higher mass ranges and the need to assign heavy multi-element ions, guesses need to be refined using multiple analytical tools the essence of which are outlined below (Kunenkov et al., 2009).It is also worth noting that although there will be one sum formula with the lowest mass error this is not necessarily the best annotation (Reemtsma, 2009). Hence several analytical methods are available that help to inform these annotations that need to be done by scientists on a regular basis.

1.2.1 Kendrick Mass Defect

One option to understand possible structures in NOM space is Kendrick Mass defect analysis. The masses of most elements are known to very high accuracy but the combinatorial explosion of the elemental composition warrants the adoption of the masses of certain repeat units as mass units. Series can be constructed by using the m/z value of an initiator i.e. a deterministically known chemical substance taken from a reference substance used for mass spectra calibration or any other known substance and adding or subtracting one of the repeat units respectively constructing a series of masses connected via identical mass differences. Known examples for repeat units in NOM space are summarized in Table 1.1 (Kujawinski and Behn, 2006). As an example we could take the mass of $1,3,5-Pentanetricarboxylate$ with the sum formula of $C8H9O6$ and m/z ratio of 201.0404617. If we now have the experimentally annotated peak of $1,3,5-Pentanetricarboxylate$ we can extend this as a series by adding or subtracting any of the repeat units in an attempt to extrapolate from a known substance. Creating such series will result in $201.0404617 + n * repeatunit$ if we are successively adding one type of repeat unit. As an example by adding $H2$ with its weight of 2.02565 successively we would yield a series of $201.0404617 + 2.01565 + 2.01565 + 2.01565 + ...$

The Kendrick mass is defined as:

$$Kendrickmass = IUPACmass \times (nominalmass/exactmass) \tag{1.1}$$

While the Kendrick mass defect is defined as:

$$Kendrickmassdefect = nominalKendrickmass - Kendrickmass \tag{1.2}$$

Thus the Kendrick mass scale rescales the mass spectrum from the exact International Union of Pure and Applied Chemistry (IUPAC) mass scale 2.01565 that we used in our example, to the Kendrick mass scale using the nominal mass of 2.0 as a mass for $H2$. Mass defects of a chemical compound is the difference between the exact mass converted to the Kendrick mass through normalization and its nominal Kendrick mass. Series can then be constructed in an analogous way with a same mass defect separated by the mass of for example $H2$ (Hughey et al., 2001). Members of the series in our example will then be described by having the same Kendrick mass defects repeatedly separated by the Kendrick mass of $H2$ (Kendrick, 1963). As such we have a plot of a complex mass spectrum separated by the nominal mass on the x-axis and the mass defect on the y-axis. This enables one to show sequences for each class of compounds that will have

the same mass defect plotted at the same position vertically while spaced differently on the nominal mass axis, yielding a visualization of the mass spectrum that facilitates analysis and enables the search for substances that belong to the same family (Sleno, 2012). Many peaks differ by nominal masses corresponding to the repeat units mentioned in Table 1.1 (Kujawinski and Behn, 2006). Hence applying Kendrick mass difference analysis can be utilized for peak annotation, through this extrapolation. However for Kendrick mass defect analysis to work, one needs repetitive systems with a priori known functional groups. One of the limitations in employing Kendrick Mass defect analysis is thus that it does not reveal any new building blocks, a drawback that is solved to a certain extent by the algorithm outlined in Section 1.2.3.

1.2.2 Van Krevelen Diagrams

Van Krevelen diagrams help to gather a rather broad overview of of the characteristics of a NOM spectrum. In van Krevelen diagrams the molar ratios of H/C on the x axis are plotted against respective O/C ratios on the y axis, effectively normalizing to the carbon number in a compound loosing much of the information contained in the original data (Reemtsma, 2009). Each class of compounds tend to plot in a specific areas of the diagram making the identification of such classes easier. However this also leads to misinterpretations as not all compounds that plot in similar areas actually belong to the same class. There is always a possibility that peaks of different compounds coincide with respect to their P/C and H/C ratios but belong to a different class altogether (Reemtsma, 2009). Since peaks with certain m/z values correspond to specific H/C and O/C ratios, these ratios can be inferred and can be then plotted in a van Kreleven diagram. Some areas of the van Kreleven diagram will naturally remain empty as for example only even number of hydrogen atoms can occur due to chemical constraints. Furthermore following lines in a typical van Kreleven diagram show reaction paths involving loss or gain of elements (Kim et al., 2003). On the other hand drawbacks of these diagrams are that they normalize to the carbon number sacrificing information on heteroatoms. Furthermore it is worth noting that the axis plotted to be orthogonal to each other are not fully independent of each other, as the number of hydrogen atoms is clearly influenced by the number of carbon atoms present in a substance (Reemtsma, 2009). Hence van Krevelen diagrams can be used to observe and compare average global properties of different NOM samples but have their drawbacks that need to be carefully taken into account when using them in analysis tasks. The van Krevelen diagrams again provide a way to facilitate identification of low mass differences through the loss of molar ratios of C, O, H and possibly N or S. However just as Kendrick mass defect analysis, van Krevel diagrams are limited to lower mass ranges because it requires all individual ions to be previously assigned a molecular formula (Kunenkov et al., 2009).

1.2.3 The Total Mass Difference Statistics Algorithm

The total mass difference statistics algorithm is suited for automated finding of repetitive patterns of mass differences in mass spectra and can find previously unknown compounds in relatively high mass ranges by means of extrapolation. Repetitive patterns are found by computing mass difference appearance probabilities. Firstly monoisotopic peaks are

determined and peaks without any neighbouring peaks are excluded for further statistical analysis but not from the raw data itself. A difference matrix containing all pairwise mass differences between the remaining isotopic peaks is then being constructed. Subsequently probabilities of mass differences are inferred from the matrix giving indication on which mass differences occur frequent enough to be considered recurring events and peaks corresponding to low probability mass differences can then be excluded. If available sum formulas giving indication of the elemental composition that correspond to the m/z value are then assigned to members of the abundant mass differences peaks. There can still be multiple formulas assigned to one peak, however the algorithm constraints this space in such a way that facilitates the annotation because of the assumption that only frequently occurring mass differences show peaks emerging from the characteristic successive fragmentation of functional groups from the parent ion. It is worth noting that these frequently occurring peaks can potentially also be expressed in terms of periodicity in the spectrum's intensity the investigation of which is subject of this thesis. After masses have been assigned to some of the peaks mass differences then may be used to connect ions with lower molecular weight and ions with higher molecular weight effectively extrapolating and inferring new sum formulas from known ones (Kunenkov et al., 2009). The implementation of the Initiatortrees described in Section 2.3.2 may be seen as an extension of these mass differences based on known substances used in calibration of FTICR spectra. The drawbacks of these algorithm are clear. For reasons of computational complexity peaks are already removed before computing the similarity matrix, and although this might be reasonable from an efficiency perspective it might still remove peaks that occur repeatedly but are not members of isotope or homologous series. By exploiting periodicity we infer probabilities of occurrence in a more efficient way as explained in Section 2.2.2 and do not a priori exclude any data points. This might yield information that has been omitted by this algorithm.

1.2.4 Algorithms Exploiting Correlation for Mass Spectral Data Analysis

Usually signal autocorrelation methods, which are effectively mass autocorrelation functions in the context of mass spectra, are used to detect periodicities in mass spectral (MS) data in a local or global context as desired. By autocorrelating in different regions of the data and overlaying the results changes in polymer architecture can be discovered. The periodic recurrence of peaks at the extrapolated values of a repeat unit mass helps to discover peaks corresponding to polymer structures in a noisy spectrum by means of autocorrelation functions that compare intensities at all mass differences across the spectrum. When peaks match at specific lag the autocorrelation coefficient increases (Wallace and Guttman, 2002). Identifying the mass difference between such matching pairs of peaks is usually difficult in noisy MS data and visual inspection of the data is often tedious because different mass resolutions have to be inspected manually and this process can lead to errors due to memory limitations of the scientist.

Autocorrelation is useful to describe periodic patterns found in mass spectral data, and it is often used to identify repeat units of a polymer. It helps to identify the periodicities in the spectrum without the baseline noise, by representing a way of averaging the spectrum that often times still contains noise which obstructs analysis even after

peak picking based de-noising has been applied to the data (Wallace and Guttman, 2002). Other work investigated how to exploit cross-correlation, i.e. correlation between two distinct signals, functions to provide a measure of similarity between protein sequences obtained from a database and fragment ions obtained via mass spectrometry (Eng et al., 1994). However to date none of the approaches used autocorrelation to find exact patterns but rather individual peaks and their periodic repeats to find repeat units of polymers. Algorithms that could be used to extract such patterns are outlined in the section below Section 1.3.

1.3 Algorithms in Periodic Pattern Mining in Time Series

One type of functional relationship in dataseries that has not yet been explicitly exploited in mass spectral data are frequent patterns. Because of the recurring mass difference between peaks it might be that the regular structure sometimes exhibited by mass spectra points towards chemically relevant data and can be used for denoising as has been done in (Wallace and Guttman, 2002). On the other hand periodicity could also be used to find the exact peaks that are involved in creating such mass differences without prior knowledge about the repeat unit if periodicity determines chemical relevancy.

Apriori algorithms are a prominent group in the class of frequent pattern mining algorithms. Many improvements have been made in the history of apriori algorithms since apriori based algorithms were invented by Agrawal and Srikant (Agrawal et al., 1994). Apriori algorithms are based on the apriori heuristic which states that if a length i pattern is not frequent than respective super patterns of length $i+1$ can never be frequent as well. This property is also known as downward closure property. Candidate patterns of growing length $k+1$ are usually generated only from the set of frequent patterns of length k which is considered the seed set. Thus at every step the apriori heuristic is used to prune the number of candidates that need to be generated. In many but importantly not all cases apriori algorithms manage to reduce the size of candidates that need to be generated. Apriori algorithms however suffer in the face of prolific frequent patterns which are often caused by rather long period lengths Han et al. (1999). Other algorithms employ the FP-growth method which helps to alleviate this issue, by employing a divide and conquer strategy. First scanning the data to derive a list of frequent items and ordering them according to their frequency effectively compressing it into a Frequent pattern FP-tree data structure (Han et al., 2007; Zou et al., 2001). Because mass spectral data often contains many data points that occur periodically and because the length of the periods is often quite extensive this leads to an extremely large number of candidate patterns that would need to be generated. It would presumably be better to use algorithms that start from the maximum subpattern rather than from the minimum frequent patterns, such as the maximum subpattern hitset algorithm (Han et al., 1999). Details on the implementation of this algorithm will be given in Section 2.2.

This is interesting for mass spectral data because we are generally interested in the maximum periodic pattern that contains a mass difference corresponding to one of the repeat units in NOM space rather than the minimum patterns or their frequencies.

Since all the important information will be contained in these maximum periodic pattern and no additional information is gained when retrieving subpatterns or the counts of frequent patterns. It is foremost interesting to us to retrieve periodic patterns because periodicity often hints at chemically relevant peaks in the mass spectral data as was described in Section 1.1.1 Thus the apriori like property will usually not suffice for a time efficient implementation to mine patterns in mass spectral data. However we could exploit the linear time property of the maximum subpattern hitset algorithm which will be explained in Section 2.2.2. Even the maximum subpattern hitset algorithm as well as the other algorithms mentioned above are considered single period algorithms i.e. they find periodicities when given a single period length as an input parameter. Thus the first task in order to identify periodicities is to find candidate period lengths that serve as an input for the maximum subpattern algorithm to avoid using a brute force approach applying the maximum subpattern hitset algorithm for all possible period lengths. An outline of the autocorrelation based algorithm used to find such periodicityhints is given in Section 2.2.1 (Berberidis et al., 2002a). Given periods of length p the maximum subpattern hitset algorithm could then provide a viable option in the face of prolific patterns due to long period lengths. This algorithm employs a single-period apriori algorithm to find all frequent one-cycles i.e. recurring patterns of length 1. This is done for a given period length mined by the previous algorithm employed to find periodicityhints. If these frequent one-cycles satisfy a confidence threshold in a time series they can be deemed frequently recurring peak heights on the intensity domain in the mass spectral data we will consider. The maximum subpattern hitset algorithm promises linear time as it only scans the time series twice in total to retrieve all frequent one-cycle patterns and create a maximum pattern from the frequently recurring one-cycle patterns. Hence this is the algorithm that will be employed in this thesis, foremost for its efficiency even in the face of long periods and also for the reason that it starts with the maximum pattern and successively retrieves shorter patterns instead of starting with smaller patterns and successively generating candidates for longer patterns (Han et al., 1998, 1999). This is important because effectively we are interested in the maximum pattern i.e. a pattern consisting of series of peaks differing by nominal masses corresponding to the repeat units. The maximum periodic pattern in the mass spectrum that could hint at a series should be retrieved, as no more additional information will be generated by extracting subpatterns from this maximum pattern that exists due to periodically recurring peak intensities in a mass spectrum.

1.4 Objective

Computational and analytical methods at present help to elucidate some parts of this theoretical space, however because all these tools do not yet uncover all properties of mass spectra unequivocally there is still space to investigate new computational methods to extract information from mass spectral data. Although software exists to find correlated peaks (Wallace and Guttman, 2002) and meaningful mass differences (Kunenkov et al., 2009). The approach taken in this study will aim at yielding the exact patterns at different length of periodicities and see if periodicity can be used as an unsupervised way to extract important parts describing the data. Hence the question at hand is if periodicity in the mass differences of NOM spectra could be used to identify chemically

relevant peaks. However as of yet there is no knowledge if these mass differences on the mass axis can also translate to periodicity on the intensity y-axis. Scientists at Helmholtz Zentrum where primarily interested in mining patterns in the frequency domain as this is an area that is still largely unexplored. This thesis is aimed to provide a first insight if mining periodicity on the intensity domain could be used to mine such frequency patterns. If this would be the case we could use the knowledge to mine these patterns on both the mass and frequency axis which can be set interchangeably as the x-axis of the data. Hence this approach if successful would also provide a means to explore patterns in the frequency domain corresponding to known mass differences in the mass domain, by comparison of patterns containing mass differences corresponding to known repeat units and their frequency counterparts. Many of the computational methods above have one considerable drawbacks that is primarily linked to the reduced mass resolution in higher mass ranges but also to the employment of methods that work based on known fragmentation of repeat units rather than a more unsupervised exploration of fragmentation patterns. Unsupervised methods that yield insight by employing another functional relationship of the data, such as periodicity, could alleviate the issue of reduced mass resolution as they would not be too sensitive to such artefacts. Rather they are sensitive to the values at the intensity dimension, disregarding the mass dimension which is usually ignored in favour of the mass dimension. It is at present unknown if unsupervised methods like these would yield insight to any unknown structures in NOM data. The approach of finding periodicity will be different from the approach taken by Wallace and Guttman (2002) as it not just exploiting the autocorrelation function of the mass spectrum but furthermore aims to retrieve exact periodic patterns that cause these relatively higher autocorrelation values to appear in the autocorrelation function. We would like to investigate if it is possible to retrieve the exact periodic patterns and see if they yield data that can be considered chemically relevant using a periodic pattern mining algorithm commonly employed in time series analysis. Furthermore even though the total mass difference algorithm elucidates a larger part of the theoretical space in NOM it is still a supervised method using known repeat units, hence an unsupervised approach if successful could also yield insight in previously unknown mass differences that recur throughout the data. The algorithm explored in this thesis is applied to mass spectral data for the first time and runs in linear time. It is geared towards uncovering periodic patterns without any knowledge of the chemical problem. Thus it could potentially help to uncover compounds that are missed with current methods as all of them suffer from imprecision in higher mass ranges and depend on information of known fragmentation patterns. This is based on the assumption that periodicity should be insensitive to higher mass ranges, thus remain accurate even in higher mass ranges. In conjunction with a tree based annotation approach in Section 2.3.2 these patterns should be investigated.

An additional way to explore frequency space can be found by means of sonification and spectral analytical methods that are commonly employed in the musical domain. This also links to the problem that there is still a gap between data available to individual research groups and data accessible to the scientific community (Reemtsma, 2009). Audio fingerprinting algorithms are a known means to store and retrieve data (Wang et al., 2003) and we will explore this mechanism for sonified frequency based data from mass spectra. Providing mass spectral data publicly will likely lead to a surge of the de-

velopment of new computational methods and advances in the analysis of mass spectral data. As the available amount of mass spectral data grows, the need for a mechanism that could help organize huge amounts of data potentially stored in databases is called for. These mechanism would need to exert similarity judgements based on frequency data's properties as well to be useful for organizing large amounts of spectral data. Mapping of mass spectral data from the frequency domain to the musical domain is called sonification.

Sonification has previously been used to aid analysis of scientific data in diverse fields (Hermann, 2008; Nasir and Roberts, 2007). However so far this has been done mostly in order to add an additional dimension or highlight one dimension during data exploration and interpretation (Milczynski et al., 2006). Hence at the moment we are not reaping the full potential of sonification to reveal structures and relationship that would otherwise remain hidden. We propose that if comparison is done by a computer algorithm, it will become more efficient to identify similar samples. This concept may then be applied to any kind of data that resides in the frequency domain, and thus automatically lends itself to sonification. The approach taken in this thesis is to first convert the mass spectral original frequency domain signal into audio data (sonification), after which finding similarity is done by employing algorithms conventionally used on audio data. Because mass spectral data has commonly been analysed using the mass domain scientists at Helmholtz Zentrum wanted to understand if there are any benefits of using the frequency domain as a basis for analysis. Hence a comparative analysis of the mass and frequency domain will be conducted in order to understand the benefits and drawbacks that exist between these modes of data representation. A procedure using an algorithm employed in the musical domain that derives features from the frequency time domain representation of sonified data to store and retrieve the data from a database is employed to investigate the use of the frequency space to represent meaningful features of the data. As such this thesis aims to providing a proof of concept study of the idea of using the frequency domain and sonification in context of scientific data. Furthermore we use of unsupervised methods to mine periodic patterns on the intensity axis, that if successful offers a way to extract patterns in both the mass and frequency domain. Thus we explore another potential avenue to provide links between the mass and frequency space in mass spectral data.

Chapter 2

Materials and Methods

This section will describe the implementation of data mining solutions to the two problems stated above. It involves several data mining techniques, part of which have been implemented manually in Anaconda a scientific python distribution [1].

2.1 Data Warehousing and Data Cube

According to Han et al. (2011) (pp. 105) "consolidate data in multidimensional space. The construction of data warehouses involves data cleaning, data integration, and data transformation and can be viewed as an important preprocessing step for data mining." A data cube is a descriptive data mining technique that helps to create a multidimensional data model for data warehousing and offers a data structure that facilitates data generalization a process where large datasets are summarized at different levels of detail for subsequent analysis (Han et al., 2011) (pp. 105-157). Since a data cube is a multidimensional representation of the data, dimensions are attributes that one might want to investigate further. Usually multidimensional data models are organized around a central theme, which in our case is the Mass (m/z) values and Frequency intensity fluctuation with increasing nominal mass. Facts are individual measures i.e. individual frequency/mass values or their corresponding intensities in our data series. However there can also be non mass/frequency related attributes that stand in connection with the mass/frequency related attribute such as for example a split of one dimension into periods of length p. The data cube is commonly n-dimensional. In our case the data cube for a data series representation is two dimensional with one dimension being mass or frequency values which represent an analogy to the commonly used time dimension and as such obey an inherent order. The other dimension are the corresponding mass or frequency dependent intensity values. In our case we will have a 2 dimensional data cube with different levels of granularity. Firstly a dimension containing the discretized values of the Intensity dimension. And secondly binary vectors for each discrete bin, with an entry being 1 if the intensity value falls within a certain bin and 0 otherwise. A schematic of the data cube is displayed in Figure 2.1 (Han et al., 2011) (pp. 105-157).

[1] see: https://continuum.io

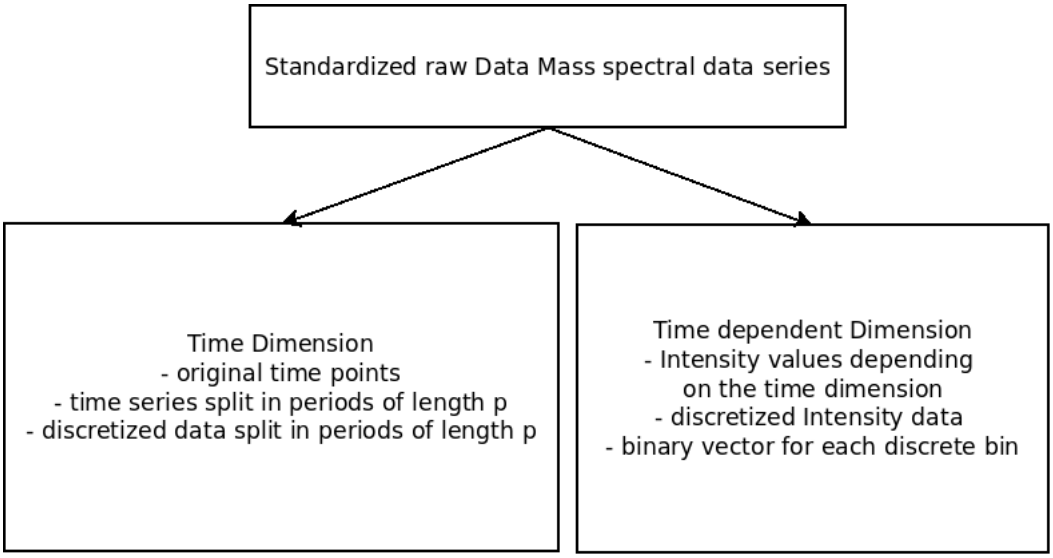

Figure 2.1: This figure shows a schematic of the datacube.

Concept Hierarchies in Data Cubes

According to Han et al. (2011) (pp.121) "A concept hierarchy defines a sequence of mappings from a set of low-level concepts to higher-level, more general concepts." These mappings can be such that a continuous numerical attributes can be generalized to concepts like categories, discrete representations or groups onto which these continuous numerical attributes for a given dimension could be mapped which inherently provides a means of dimensionality reduction in time series data (Han et al., 2011) (pp. 105-157). The time related dimension in time series data is often inherently ordered. This is also true for the mass or frequency value range respectively which will replace the time related dimension. Mass and frequency values are related in an inverse non linear manner. The data points are sorted ranging from values of low mass and high frequency to values of high mass and low frequency respectively. This also means that one needs to obey the ordering of the data points originally obtained by measurements in order to not reduce the data's meaning (Han et al., 2011) (pp. 105-157).

Data mining tasks often need to operate on a different representation of the data as they are specifically geared towards handling for example string data with a strictly confined alphabet. Hence discretization is often necessary such that the generalization of the data will still represent a meaningful sequence of events (Han et al., 2011) (pp. 86-105). In our case we will employ time series mining algorithms which often heavily depend on a discrete representation of the data as they often were invented for applications in frequent item set mining in transactional databases Han et al. (2007). As this comes at the cost of loosing some detail in the representation of the dynamic range of the data the need for discretization has to be evaluated carefully and depends primarily on the need of subsequently employed algorithms. In case of discretization the data cube provides references to the original data series in the reference cube, hence it offers a means to retrieve the original data series whenever necessary. The original data series remains accessible as it is saved in the reference cube which provides the basis to generate different level of the concept hierarchy (Han et al., 2011) (pp. 86-105).

Manual definition of such concept hierarchies are known to be repetitive tasks hence automatized ways to discretize the data are employed. Since class information is not available in our case unsupervised discretization methods need to be employed for the discretization of the time dependent intensity attribute dimension which we will use to determine periodicity. Unsupervised binning is considered a top-down splitting technique and specifying a number of bins entails how detailed the representation we want to obtain remains (Han et al., 2011) (pp. 86-105). These unsupervised binning methods are usually based on the histogram representing the frequency distribution of the intensity dimensions values. The partitioning of this frequency distribution depends on the method chosen. One can for example bin values using equal-width buckets where the value range captured by each bin remains the same. Because of its simplicity this method is implemented in our algorithm. The number of bins can be specified by the user and the data will be binned in equal-width bins accordingly. When used for discretization, bins will usually be numbered and the number of the respective bin will replace the continuous value of the attribute. This leads to the continuous intensity attribute being represented by a smaller number of possible values (Han et al., 2011) (pp. 86-105). The results of this discretization procedure can furthermore be used to map the intensity attribute onto a series of binary vectors. Specifically, each bin can additionally be represented by a binary vector with 1 indicating the value is present at this particular bin and m/z value position and 0 indicating the value at this bin and position in the data series is absent. Later on this representation will play an important role in finding periodicity hints according to the algorithm explained in Section 2.2.1 as these binary vectors will serve as input for finding the period lengths p in the data series.

Concept hierarchies can also be generated for the m/z and frequency dimension, that stand synonymous for the time dimension in time series mining. In our case we will use the period length p that has been acquired through the periodicity hint algorithm as a level in the concept hierarchy of the working cube. Effectively one can create this kind of hierarchy by partitioning the data series in segments S_i of length p. This splitting of the data series into segments of length p can be done on all different hierarchical levels of the m/z dependent and independent attribute dimension. Hence one can generate period segments S_i on the m/z attribute dimension, but also on the binary vectors and the discretized data series. This implementation facilitates retrieval of different representations needed by different steps of the algorithms used to determine periodic patterns in the data series.

Our data cube is implemented as a multidimensional dictionary where we created a nested structure of dictionaries to represent the concept hierarchies. Firstly the original time series' intensity values that are plotted on the ordinate in the original mass spectrum shown in Section 2.5, are saved in an array as the time dependent attribute dimension. Furthermore the original time dimension (i.e. in our case m/z and frequency values) are saved as the uppermost level in the time dimensions concept hierarchy. These two sequences form the reference cube which is generally used to construct a working cube which contains concept hierarchies that facilitate working with the data in different steps of the mining algorithms subsequently employed. The working cube can be considered an expanded reference cube in terms of the concept hierarchies (Han et al., 1998). As an example the values of the original time series will be residing in the working cube in form of a discretized and segmented representation and simple aggregates can be computed

through this representation in the working cube. When discretizing and establishing a concept hierarchy for the time dependent intensity dimension, one will receive one binary array for each bin value, with each binary array having the same length as the original time dependent dimensions array. Since the data cube is implemented as a python dictionary efficient retrieval within its structure is possible by addressing the content using combinations of keys in subsequent mining steps.

Thirdly a T-slice can be constructed from the working cube representation, which serves as an input for the computations needed in particular steps of the algorithms. The T-slice provides additional aggregates such as for example integer value sums of a position over all periods for a certain time point. For a specific period length, aggregates can be formed by stacking all binary vectors which were previously sliced into segments of period length p in a vertical manner. By computing sums for each position in the binary period vector, we compute aggregate values for each position in a period of length p (Han et al., 1998). These aggregates can be used to identify frequent one cycles, the significance of which will be explained in Section 2.2.2, that are needed to form $Cmax$ patterns which will be explained in Section 2.2.2. Here a simple example is given for periods of a discretized times series of length twelve for period length $p = 4$ and four discrete bins. The original continuous time series is specified as: $S = (1.66, 5.77, 10.55, 15.98, 6.67, 12.34, 16.98, 15.76, 0.55, 6.83, 14.55, 19.64)$. Discretized period segments stored in the working cube would be $S_1 = (1, 2, 3, 4), S_2 = (2, 3, 4, 4)$ and $S_3 = (1, 2, 3, 4)$. The corresponding binary vectors after binning would be stored for the entire Time series and each of the four bins using the following representation: $Bin1Vec = (1000, 0000, 1000), Bin2Vec = (0100, 1000, 0100), Bin3Vec = (0010, 0100, 0010), Bin4Vec = (0001, 0011, 0001)$. These vectors can again be split into segments of length p. The corresponding aggregate plane would simply add the numbers of occurrences over all period indices and thus find the frequent one-cycles according to a threshold. In our example aggregation would yield $Aggregatebin1 = (1, 0, 0, 1), Aggregatebin2 = (1, 2, 0, 0), Aggregatebin3 = (0, 1, 2, 0)$ and $Aggregatebin4 = (0, 0, 1, 2)$.

The discretization procedure described is equivalent to converting a given time series to a symbolic representation i.e. convert it to a finite alphabet which in my case consists of a set of numbers. By this transformation the continuous intensity series S of length and thus dimensions N is converted to a discrete string of numbers of length N, with dimensions $W < N$. Where W is the number of bins (Lin et al., 2007). It is worth noting that the algorithm for periodicity hint finding explained in Section 2.2.1 scales linearly with the size of the alphabet i.e. the number of bins as well as the length of the time series. Hence the size of the alphabet limits the number of FFT computations of size N. And we will need W FFT computations, which are assumed to be a major contribution to the computational complexity, to find possible period lengths p in a time series (Berberidis et al., 2002a).

2.2 Finding Exact Periodic Patterns in a Mass Spectrum

2.2.1 Finding period lengths

Periodic events are considered important in time series data, thus it is interesting if they can be regarded equally important in mass spectral data. The aim is to discover if periodicity would supply a way of extracting data points from the mass spectrum that can be considered chemically relevant. Algorithms developed to find periodic events often work based on circular autocorrelation functions (CAF) (Wallace and Guttman, 2002; Berberidis et al., 2002b). In mass spectrometry data period length for periodicities is generally unknown. It follows that circular autocorrelation based algorithms needs to be employed in order to give hints on possible period lengths in the data to avoiding an exhaustive search over all period lengths. However they can not substitute more detailed algorithms in order to investigate which are the exact periodic patterns occurring in the time series a method that will be explored in Section 2.2.2 (Berberidis et al., 2002b).

It is worth noting that periodic events in naturally occurring time series are rarely perfectly periodic in the strict mathematical sense. Hence even events need to be considered to be periodic in spite of imperfect CAF values implying weak or partial periodicity. From a chemical point of view the most important cause for periodicity is that chemical entities tend to be fragmented and exhibit periodically recurring isotope distributions that are made of a sequence of peaks. This sequence of peaks shows often times periodically increasing and decreasing amplitude on the intensity dimension due to the relative abundance of ions fragmented under the influence of the magnetic field in the FTICR-MS (Greiner et al., 1975; De Hoffmann and Stroobant, 2007). Similar to many naturally occurring time series they often show some kind of imperfect periodic behaviour. As such finding periodicities in Mass Spectrometry data has to be considered a problem of finding approximate periodicities and not periodicity in a strict mathematical sense. These approximate periodicities are also known as partial periodicities, and are patterns that would have peaks that contribute to periodic behaviour only at some but not all positions in the pattern Berberidis et al. (2002b). Lets assume that a pattern can be considered frequent in two cases in our discretized data series. Either we have the exact pattern 11123 recurring separated by a period length p within a time series, or the patterns 11123 ,33123 and 22123 recur separated by p. In the first case we have periodicity in the strict mathematical sense and at certain lags of a multiple of the period length p we will see peaks in our normalized CAF that correspond to a perfect correlation of 1.0 at these lags if perfect periodicity is present. In the second case these patterns share a common subpattern 123 at the last three positions of the patterns and some algorithms employing the strict mathematical definition of periodicity would not find them to be periodic, because they differ in some positions and the autocorrelation score would be imperfect. Partial pattern mining algorithms find these partial periodic subpatterns and provide a conservative estimate of possible period lengths by filtering out periods that certainly do not contain any periodic events using the relatively higher values of the CAF at lags assuming these lags to correspond to a period (Berberidis et al., 2002a; Han et al., 1999). The drawback of the use of CAF's to find periodicity hints is that parametrization tends to be difficult and that the computation of CAFs is

expensive (Cao et al., 2004; Vlachos et al., 2005). The circular in CAF implies we shift out the product at every step, and that the vector is then again moved to the beginning. This results in the vector being correlated with itself at different lags. Such that the computational complexity would be $O(n^2)$, however an FFT based CAF provides an alternative that reduces the computational complexity to $O(n \log n)$ (Berberidis et al., 2002a). The CAF will result in a function that will show a series of peaks and valleys. As for the parametrization difficulties it needs to be noted that setting a threshold parameter that distinguishes between peaks in the CAF that point towards partial periodicity and peaks that are noise is rather difficult. One problem exhibited by such an approach is that detecting periodicities using CAF's tends to introduce false positives by introducing integer multiples of the same basic period. This might manifest itself for example as CAF peaks above the noise threshold at lag 30, 60 and 90 which might be regarded as individual period lengths by the algorithm, whereas the only true period length is 30. Another difficulty is that low amplitude events that occur more frequently, can be ignored in favour of less frequent high amplitude events if a high threshold value is chosen which would ignore low amplitude events. This leads to a paradoxical situation where periodicities that are more frequent are not registered, while less frequent periodicities might be registered as periodic events. Some algorithms extend this concept and use the periodogram in order to find potential periodicities in a more coarse grained manner, whereas others rest solely on the use of CAFs which detect periodicities in a more fine grained way (Vlachos et al., 2005). The algorithm was applied using an autocorrelation threshold of 0.2 which is low but increases the likelihood that any potential periodic patterns would be found even if their corresponding peaks in the CAF are rather low but high in frequency. This implies that we would rather have some false positives than ignore high frequency but low amplitude events in the CAF. One of the benefits of the algorithm devised by Berberidis et al. (2002a) is that it overcomes the problem of noise by using the binary vectors of each bin for the computation of the CAF. This leads to the elimination of non periodic events through the multiplication with zero at positions that are not filled within that particular bin (Berberidis et al., 2002a). It follows that we decided to use this algorithm for its insensitivity to noise which is present in wide parts of the continuous mass spectrum. Using the original data series as a basis for the CAF based detection would potentially lead to a noisy CAF and difficulties in distinguishing between true periodic events and noise.

According to Berberidis et al. (2002a) the algorithm for finding partial periodic patterns has the following filter steps.

1. Scan the time series once and create a binary vector representing the discrete time series separately for every symbol in the alphabet of the time series. In our case this will be implemented in a data cube that contains binary vectors for each bin. In this first step we use the binary vector of size N of all our bins stored in a T-slice we extracted from the working cube.

2. In the second step we compute the CAF of each binary vector representing one bin and normalized it. This is helping us to derive period lengths that are later used in the refinement step. The autocorrelation function is the sum of N dot products between the original binary vector with itself shifted by all possible lags $k = 1, ..., N$. If vectors would be shifted in harmony once every five positions, resulting

in peaks of one for every lag five shift the periodicity would be five and a period of length $p = 5$ would be retrieved according to the threshold procedure outlined in step 3. As an example we could look at the binary vector $V = 010000100001000$ if we compute the autocorrelation we would see a peak at every fifth position because it would be perfectly aligned at every fifth shift provided we shift the vector one position at a time.

3. Because $N/2$ is the maximum length of a period in a time series of length N we only need to scan half of the autocorrelation vector and retrieve the values that are considered higher than the threshold. In the CAF peaks that are higher than the threshold value indicate that there is a period candidate at a specific lag and we filter out those values that do not satisfy the minimum confidence threshold while keeping the rest as candidate periods. It is worth noting that the first position of the CAF gives an approximation of the relative frequency count of the bin value in the entire discrete time series. Hence it needs to be removed before inspecting the CAF for values satisfying the threshold.

After step 3 we have yielded an estimate for possible period lengths which we will use to retrieve the exact partial periodic patterns. It is important to note that a particular bin can have more than one period, and as such the maximum subpattern hitset algorithm described below will have to search for periodic patterns for multiple period lengths and the periodic pattern search algorithm needs to be implemented quite efficient (Berberidis et al., 2002a).

2.2.2 The Maximum Subpattern Hitset Algorithm for Mining Periodic Patterns

After candidate period lengths have been retrieved using the algorithm explained above, Han et al. (1999) proposed the maximum subpattern hit set algorithm to mine exact partial periodic patterns in the dataseries. Hence we will continue to treat the intensity data series like a discretized time series of a finite alphabet. As input data for the partial periodic pattern mining algorithm we will consider a T-slice of the reference cube for each period length p. The T-slice needs to contain the periodindex and aggregation plane, as well as the original m/z series and frequency series and the intensity values which are the time dependent property of our data series (Han et al., 1998, 1999). This section should provide a brief introduction into the general motivation as to why the maximum subpattern is the method of choice for our problem. Furthermore we want to introduce some important concepts necessary to understand the maximum subpattern hitset algorithm.

Before delving into the specifics of finding periodic patterns let us inspect some definitions that are prevalent in any pattern mining application.
A pattern as defined by Han et al. (1999) as a non empty sequence s of period length p. The concept of L-length commonly denoted as i is the number of positions that are filled with a specific bin value instead of the wild card character $*$. The wild card character denotes an optional event a feature that enables us to find partial periodic patterns. If a wild card character occurs at position s_i the pattern can take on any value of the features defined in the discrete time series. Additionally we will need to understand the

concept of frequent one-cycles. Examples of one-cycles for $p = 4$ would be $1 * **$ or $***2$. Notably these one-cycles have $L - length = 1$ as all but one character in the pattern are the wild card character. The set of frequent one-cycles $F1$, is mined using the single period apriori algorithm. A periodic pattern of length i is always a union of a set of one-cycles for a given period length p, this follows from the apriori property that says that frequent $i+1$ patterns can only be frequent if the i pattern is already frequent. It follows that the set $F1$ contains the building blocks for frequent $i - patterns$ of any length thus also the maximum periodic pattern that can be identified in a data series (Han et al., 1999).

The maximum subpattern hitset algorithm is efficient because according to Han et al. (1999) the total number of scans over the time series will be 2 independent of period length. This stands in stark contrast to the single period apriori approach where in each new cycle $i + 1 - patterns$ will be compared to the time series and determined to be frequent according to a threshold (Han et al., 1998). Instead of repeatedly checking the pattern existence, the maximum subpattern hitset approach yields the maximum periodic patterns already in the second scan of the time series. If one wanted to one could yield any frequent $i - patterns$ by forming union between the maximum patterns retrieved. The rationale as to why the maximum subpattern hitset approach is much more feasible for mining partial periodic patterns is that for a given period length the number of frequent patterns of length i shrinks only slowly as there is a strong correlation between frequencies of patterns and their subpatterns. This is especially problematic as the period length p gets larger leading to an extensive number of candidate patterns that need to be generated and checked for their existence at each step. It follows that the single period apriori algorithm which starts with a pattern length of $i = 2$, after finding $F1$, and would generate patterns of length $i + 1$ in successive steps until the pattern is as long as the period length. This arguably would lead to an explosion of the number of times the time series needs to be scanned and would be rather inefficient regarding the large period lengths which we can expect in mass spectral data (Han et al., 1999).

Generally we will modify the maximum subpattern hitset routine such that we only form the candidate pattern set once specifically for the maximum pattern that can be derived for the frequent one cycles found. This is enough because we can safely assume that any information contained in periodic patterns of length p in a mass spectrum is contained in its maximum pattern $Cmax$ that can be formed using the frequent one cycles mined. The procedure to find $Cmax$ will be explained in Section 2.2.2. After $Cmax$ is formed we can check if the pattern actually exists in the mass spectrum and return those patterns that are indeed found as a result of the maximum subpattern hitset algorithm. Before constructing $Cmax$ we need to determine the frequent one-cycles which can be seen as an equivalent to frequently recurring peak intensities in the raw data. They can be used to reduce the set of possible frequent patterns efficiently by creating $Cmax$ that can only consist of frequently occurring peaks in the raw data. Usually only a small number of such one-cycles are frequent at a particular position, hence this reduces the large number of one-cycles that can potentially be present at a particular position. To put it into simpler terms the position i can be filled with any of the values possible after the discretization procedure, however only few of them will actually be considered frequent when looking at all period segments of length p. Hence frequent patterns can be formed after the set of frequent one-cycles $F1$ has been determined. A

detailed explanation of the procedure to mine $F1$ will be given in Section 2.2.2 (Han et al., 1999). In order to determine the frequent maximum patterns in a mass spectrum we will need only two scans of the dataseries. One to determine the frequent one-cycles, that can however be acquired using the aggregation plane, and one scan to determine if the $Cmax$ formed from these one-cycles actually exists in the dataseries.

Single Period Apriori for finding One Cycle Patterns

The concepts given in the last section, will help us to mine the set of frequent one-cycles which serve as a basis for subsequent steps of the algorithm. The single period apriori algorithm can output the complete set of periodic pattern for a given period p, however in a less efficient way then the maximum subpattern hitset algorithm, hence we will use it only in order to determine frequent one-cycles as a subroutine of the maximum subpattern hitset algorithm.

The maximum subpattern hitset algorithm uses a T-slice of the working cube as its input containing the aggregate plane besides other levels of the concept hierarchy. This aggregate plane makes it easy to retrieve the frequent one cycles according to the confidence threshold Γ. This can simply be done by using the periodindexall aggregation plane of the cube. We can summarize the binary vectors by stacking them vertically and summing up the positions that are filled for a given period length. Using this count one compares it to the $support = \Gamma * (N/p)$. If the support is larger than confidence threshold, then we deem this one-cycle frequent. In this case we put it into the candidate set $F1$. The confidence for finding frequent one-cycles was 0.2 which is rather low similar to the threshold for the periodicity hint algorithm. However it is geared towards retrieving a large number of patterns. This potentially influences the precision as more irrelevant patterns might be retrieved. Furthermore we will use a threshold of 0.8 in order to determine the algorithms behaviour when using a more stringent requirement on the periodicity of the building blocks of our periodic patterns. Finding frequent one-cycles is implemented in our approach according to Han et al. (1998) by outputting a set containing patterns in a way that the cycles that are formally represented by $C = (periodlength, offset, value)$. When looking at the content describing the cycle C the first integer value is the position at which the value is expected to be repeated in a periodic fashion. The second integer value represents the offset or the mass point at which the pattern first occurs. The last value will represent the bin at which the value can be found i.e. the address of the fact in the working cube or T-slice. The first iteration of candidate set generation in the single period apriori will yield a set of such cycles of $L - length = 1$ and will help us to retrieve the values in order to construct patterns that will be used to check for the pattern existence in Section 2.2.2 (Han et al., 1998).

The single period apriori algorithm subroutine will terminate after the frequent one cycles are found, as the maximum subpattern hitset algorithm provides a more efficient implementation of the remaining steps (Han et al., 1999). We will give a motivation as to why the apriori algorithm would lead to a combinatorial explosion in the number of candidate patterns that would need to be generated when searching for the maximum periodic pattern in a time series. Consider for example a period length of 27, which notably is quite short in the context of mass spectral data. Thus the following example does certainly not show the upper bounds in space complexity. However it gives us

a reasonable idea as to why problems might arise and motivates the further steps of the algorithm. We consider a binning procedure that bins the data into 5 bins, then if each position of the pattern is filled with either one of the 5 bin values, i.e. we would have found the once-cycles of each bin at each position to be frequent, we would have $5^{27} = 7450580597 \times 10^{18}$ distinct candidate patterns. These candidate patterns would need to be generated and checked for their existence in the time series S. This is clearly not a permissive number of candidate patterns to keep in main memory. Furthermore it is infeasible in terms of time complexity to scan the time series for all of these candidate patterns. Hence we will resort to the maximum subpattern hitset algorithm which provides the advantage of reducing the number of scans, thus providing linear time complexity for a given period length to find the maximum periodic patterns in a time series by producing the maximum subpattern immediately after finding $F1$ Han et al. (1999, 1998).

Formation of the Candidate Frequent Maximum Pattern

After finding frequent one-cycles in the previous step, which remains an effective way to reduce the candidate set, because there are often still only a small number of features being frequent at a particular position we continue onward to find the maximum periodic pattern in the time series (Han et al., 1999). The maximum subpattern hitset algorithm decreases the number of candidate patterns that need to be generated by forming the frequent candidate maximum pattern $Cmax$ from $F1$ containing the one-cycles and thus the basic building blocks of any periodic patterns. For a particular $F1 = \{1 * * * *, *2 * **, *5 * **, * * 8 * *, * * *3*\}$, we could create $Cmax$ for a pattern of L-length four to be $12,583*$. Four positions are filled with a character other than the wild card character. Here the set representation $\{2, 5\}$ signifies a logical disjunction where the optional event is either 2 or 5. If we would be missing the pattern $*2 * **$ in $F1$, we could simply represent the maximum pattern that can be generated from the union of $F1$ pattern as $1583*$ without the use of optional events Han et al. (1999).

$Cmax$ will need to be represented in a notation that makes disjunctions and efficient generation of candidate patterns from it possible. In the next step described in Section 2.2.2 we will need a representation that makes it feasible for us to search the pattern in the discrete time series S in an efficient manner. Hence we translate $Cmax$ to a regular expression in our implementation that allows for these logical disjunctions and represents the set $Cmax$ in a manner that allows the exploration of the discrete time series. This is due to the fact that python in the current form does only allow for frozen sets within sets and these frozen sets are immutable. The implementation using sets as suggested by Han et al. (1999) would be tedious as we would need to repeatedly construct mutable sequences from the representation as an in place modification of sets within sets is not possible. Hence we resorted to the use of regular expressions. Each position s_i in a pattern is represented as a regular expression group. A disjunction at a particular position is denoted with the or operator in the regular expression syntax of python $|$. Any other position will be a group containing the exact digit representing the bin value at this potion. In case of an optional event in terms of a wild card character any bin values are matched with the regex group. A regex is dynamically formed from $Cmax$ to be used in the procedure to find existing patterns in the time series. This representation however is limited because python regular expressions only allow for a

maximum of 100 groups. On the other hand because this is the first application of this algorithm to mass spectral data, we will leave it up to future work to improve on this issue if the algorithm proves to be useful to extract meaningful information.

Check Pattern Existence in the Time Series

After we found $Cmax$ we need to verify whether a candidate patterns manifestations are in fact present in the discrete representation of the time series S. It is inherently a pruning step within the algorithm as it will remove patterns that turn out not to be present in the discrete representation and as such reduce the set of frequent patterns. This would be particularly important if one would like to further mine subpatterns and their frequencies and construct a maximum subpattern tree to retrieve their respective frequencies, a step that will be omitted in this thesis because $Cmax$ contains all relevant information (Han et al., 1999). Verifying the patterns existence is done efficiently by retrieving the discretized representation of the entire time series in the T-slice of the time series that is already split into segments S_i of length p. That is all S_i are segments consistent with the period length currently under investigation (Han et al., 1999). The representation of $Cmax$ is then translated into a regular expression according to the procedure explained in Section 2.2.2.

We progress through the time series looking at the period segments S_i of length p sequentially. This time we want to confirm that the exact patterns that can be derived from $Cmax$ in fact exists in the time series. If one manifestation of $Cmax$ is found in S_i checking the period segment with the regular expression representation of $Cmax$ we would call the pattern a hit in S_i Han et al. (1999). In theory for our approach it would be enough to retrieve the patterns from the hitset. However as we would like to generate a possibility for retrieving the periodic masses and corresponding frequencies as well as the mass and frequency dependent intensity values we decided to store the results in a tree data structure. The nodes of the tree contain information on frequency and mass values and their associate intensity values as well as the frequency of the pattern found. This also makes it easy to retrieve the information after checking the pattern existence for subsequent post processing steps. An example of a maximum pattern hit if $Cmax$ is $12, 583*$ is given by Han et al. (1999). In this case the maximum hit subpattern for a period segment $S_i = 12, 58**$ because it is present in at least one of the S_i in S. If this is the case then we can infer that its superpattern $12, 583*$ is not present in any S_i and thus not present in the time series. It is worth noting that through these hit maximum patterns we could derive the complete set of partial periodic patterns by forming joins of these patterns (Han et al., 1999). However this step will be omitted as the maximum patterns already contain all possible mass differences we could be interested in. Hence no new information other than the frequency which is irrelevant in the light of the chemical question we try to answer. The only information necessary to determine the periodic patterns are the $Cmax$ generated by periodic peaks appearing in a mass spectrum.

Because the patterns were constructed using frequent one-cycles periodicity was implied as the construction of $Cmax$ is based on these periodically recurring peak intensities. The confidence threshold of the patterns was kept at 0.0 because we wanted to retrieve maximum patterns as opposed to smaller subpatterns. The likelihood of smaller subpatterns, provided one would us the join step to find partial periodic patterns, to achieve the confidence threshold is higher. However when looking at the maximum pat-

terns only we can infer that they would by definition yield a low confidence because they are often large and will often only be found once in the data series. The low threshold implicated that we would also retrieve the maximum number of possible patterns generated from frequent one-cycles.

The algorithm presented for mining partial periodic patterns for a given period p in a time series S is based on the max-subpattern hit set algorithm and a summary will be given in Section 2.3.3. However before we approach this section we still need to understand the specifics of the post processing and analysis of the patterns we found as well as initiatortrees which are described in Section 2.3.2 and are used to determine the sum formulas that correspond to peaks in the patterns.

2.3 Analysis of the Patterns found

2.3.1 Filtering of Chemically Relevant Patterns

For the evaluation of the patterns found it was necessary to filter patterns according to their interestingness for chemical interpretation. This step in the pattern mining approach renders the process a supervised pattern mining approach (Van Leeuwen, 2014), since we specifically target patterns that contain mass differences corresponding to one of the repeat units in Table 1.1. So in terms of the analysis we loose the benefit of the unsupervised pattern mining approach. However an analysis of the interestingness in a chemical sense would otherwise need the expertise of an analytical chemist, and as such the post processing steps here are a means to see if we can at least extract the information that can currently be derived by known repeat units (Kujawinski and Behn, 2006).The reasoning behind this is that at first we would like to investigate if many patterns are extracted and many patterns are useful when compared to the current state of the art algorithm from (Kunenkov et al., 2009). If our approach yields a near optimum retrieval of such patterns, the rest of the data should be investigated by an analytical chemist in order to see if retrieved patterns uncover some new insight that is chemically relevant. However if this would not be the case a conclusion would be that periodicity is not an interesting way to investigate chemically relevant patterns at least in NOM spectra. This does not occlude the possibility of periodicity being relevant in other types of spectra such as those derived from biopolymers where investigation of periodicity has yielded promising results (Yu et al., 2011). The assumption is such that periodicity enables unsupervised retrieval of the chemically relevant data. Because patterns could contain any number of masses and could partially contain irrelevant mass differences besides relevant ones we computed the mass differences contained in a similar manner as the ground truth for all patterns. That is in a double for loop we looked if mass differences between peaks in the retrieved patterns correspond to an integer multiple of one of the repeat units in a brute force approach with an error threshold of 0.000009. The error threshold was enforced as a modulus operation on the difference using the known repeat units. This made sure that all mass differences found in the patterns were accounted for. Furthermore it made it possible for us to analyse if patterns that were a result of periodically occurring peaks are a means to mathematically model the important parts of the data. If this would be the case, then we would expect nearly all patterns to contain mass differences corresponding to an integer multiple of repeat

units, and periodic patterns would recur at a constant spacing described by such mass differences as has been shown by Yu et al. (2011).

2.3.2 Initiatortrees in Natural Organic Matter Space

Initiatortrees as seen in Figure 2.2 are built using the monoisotopic masses of known reference substances measured by the FTICR. These reference substances shown in Table 2.1 are commonly used because FTICR measures can be calibrated based on their known flight times and thus their measured m/z ratios as recorded by the detector in the FTICR (De Hoffmann and Stroobant, 2007) (pp. 128). These known reference substances in conjunction with known repeat units in NOM space can be used to cover part of the search space (Kujawinski and Behn, 2006). This search space is commonly searched using approaches like the mass difference statistics algorithm and other methods described in Section1.2. In order to annotate mass spectral peaks with a given mass the sum formula of a substance needs to be found. Starting from one of the reference substances in the Table 2.1 below as a root node, initiator trees can be built by adding and subtracting known repeat units. As an example adding and subtracting the mass of the repeat unit $H2$ (+/- 2.01565) corresponding to a hydrogenation and dehydrogenation respectively can be done and thus progression of peaks throughout the mass range measured by the FTICR-MS can be simulated. A visual representation of this example can be inspected in Figure 2.3 which provides insight into a part of an initiatortree. When we add and subtract a repeat unit, the sum formula of the node corresponding to this added or subtracted mass is updated accordingly with the number of H, O and C atoms corresponding to the reaction at hand. Again while building the tree a regular expression approach is used to identify the previous nodes constituents and add and subtract a number of atomic units for the respective reaction. This operation is done repeatedly until the maximum number of reactions for a given mass range is reached.

For the construction of the initiator trees treelib [2] a python tree implementation has been adapted to store information on the chemical formula, masses and reactions within the nodes of the Initiatortree. For the visualization implementation an adaptation of the d3.js[3] interactive tree diagram[4] has been used. The nodes are the chemical entities and the information made accessible on mouse hover are the masses or vice versa depending on the purpose of the visualization. This means that for the initial construction of the tree we have the nodes and their respective chemical formulas on display. On the other hand in case we search for a path or node in the tree corresponding to mass peaks we will reconstruct the trees from the original .json files switching the information used to represent the nodes. As our aim is to visualize results to facilitate the inspection of certain masses in the mass spectrum, the masses will be displayed as shown in Figure 2.3 to facilitate the users exploration. Hovering over the mass in this exploration step shows the chemical entity corresponding to that mass, while the mass is visible by default as the name of the node.

For the purpose of this thesis we have 6 known repeat units in NOM space listed in Table 1.1 disregarding any nitrogen or sulphur containing ions. The construction of

[2] http://treelib.readthedocs.io/en/latest/

[3] see:https://d3js.org/

[4] see: http://bl.ocks.org/d3noob/8375092

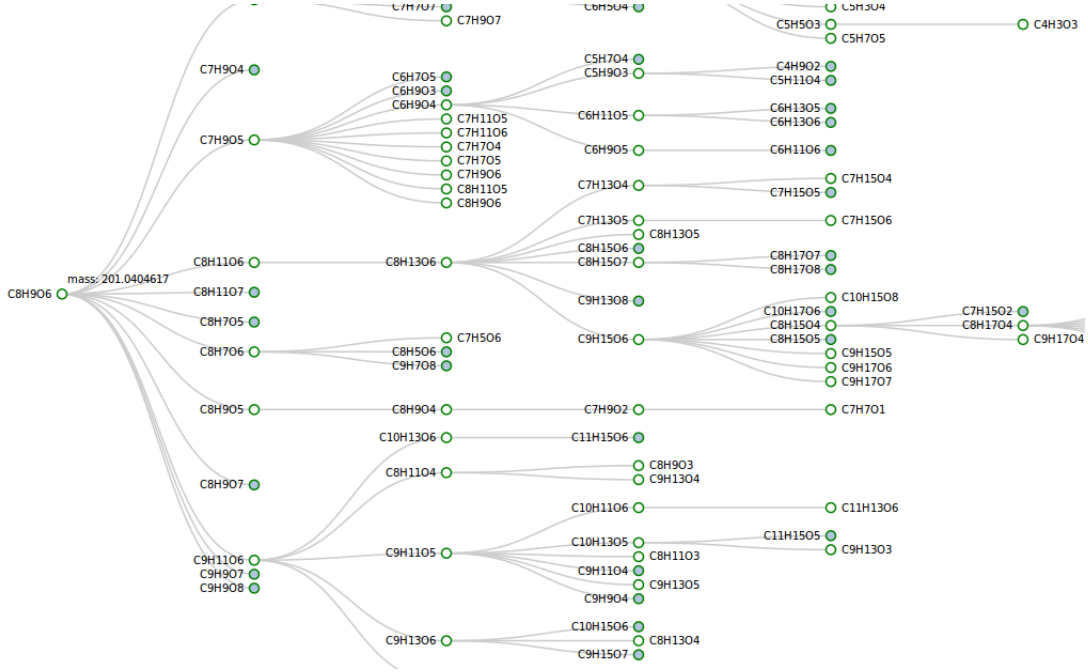

Figure 2.2: shows part of the Initiatortree built from the root node C8H9O6.

such a tree would result in the construction of a $k-ary$ tree which is not feasible in terms of its space and time complexity. After a height of the tree has been reached that corresponds to the maximum number of reactions of the second lightest repeat unit $CH2$ with a weight of 14.01565 within the mass range specified for building the tree, we will only continue to consider additions and subtractions of $H2$ the lightest repeat unit. The implementation thus takes into account that the second most frequent possible additions within a mass range, can occur when doing subtraction and additions using the repeat unit $CH2$. Thus using the height of the tree when having covered the mass range with reactions of $CH2$ automatically includes the maximum possible reactions of all other repeat units because their weight is higher and thus less reactions are possible within the given mass range. Because of this higher weight it is even possible that nodes are created that go beyond the specified mass range. This is a result of the same number of addition and subtraction of the repeat units heavier than $CH2$. However this is should be of no concern as it only provides additional information.
After we reached the level of the tree corresponding to the maximum possible reactions considering $CH2$ covering the whole range of possible reactions within the given mass range at least once, the rest of the additions and subtractions were done adding and subtracting only $H2$ i.e. the lightest repeat unit with a weight of 2.01565. Notably the construction of the tree results in a highly unbalanced tree, however due to space and time complexity this was the best we could do and possibilities for improving on this approach are discussed in Section 3.1.2. Generally this approach brings also several limitations. Firstly due to the limitation of not being able to create duplicate nodes we have a problem if we are looking to trace a characteristic fragmentation path in the tree approach. If any, only few paths are found because a node with a given chemical formula and mass can theoretically be contained multiple times in the tree, however due to the space complexity of the construction of such a tree this is not the case and the

Chemical formula	mass
C8H9O6	201.0404617
C9H5O7	225.0040764
C11H7O7	251.0197264
C14H11O6	275.0561117
C15H9O7	301.0353764
C15H17O8	325.092891
C17H19O8	351.108541
C18H15O9	375.0721556
C19H13O10	401.0514202
C19H21O11	425.1089348
C21H23O11	451.1245848
C22H19O12	475.0881995
C23H17O13	501.0674641
C25H17O13	525.0674641
C27H19O13	551.0831141

Table 2.1: Reference substances used for calibration of mass spectra and as root nodes in initiatortrees.

node will only appear once. If the nodes were contained multiple times in the tree we would be able to find paths more often, instead of only the start node or the target node exclusively. Furthermore having duplicate nodes would enable us to find different reaction paths in the different trees that can result in the same chemical entity. Imagine we find a mass difference in our pattern that corresponds to a multiple of the mass of one of the repeat units and is calculated by taking the difference of mass 1 and mass 2 separated by an arbitrary number of peaks. Now if we had duplicate nodes in our tree we would be guaranteed to find all paths in trees containing both the corresponding nodes for these two masses. We would have effectively found the chemical entities that could be annotated to the masses in the pattern and the sum formula of the ions that caused the peaks at the specific mass to occur with a mass difference between the masses. This could help investigate which repeat units were split from the original parent ion. However because we have no duplicate nodes we will only find the nodes corresponding to both mass 1 or mass 2 at one specific place in the tree, effectively finding at most one path while there could be multiple. Often times we would possibly only find one individual node corresponding to either mass 1 or mass 2 and no path if for example the node during tree construction was already placed in conjunction with another mass. I.e. we will probably be able to find mass 1 but we will fail to find mass 2 in the sequence pathway or vice versa and thus can not complete the pathway. Hence we will not be able analyse the whole pathway, while a tree with duplicate nodes would have given us the correct path causing the peaks to occur at these masses. This means that our initiatortree which can be seen as a database of mass differences is incomplete and often only single nodes will be found. However whenever a starting or ending node of two peaks connected by a mass difference corresponding to one of the repeat units is found we can also conclude that there would be a node corresponding to the particular mass difference found somewhere else in the tree, and manual extrapolation would yield the

corresponding mass. Another drawback of this approach is that theoretically there is still a possibility that nodes are not created that would result in new nodes after addition and subtraction of only $H2$ has commenced. However we assume this possibility to be small, and disregarding their drawbacks the initiatortrees can be considered sufficient for the proof of concept of these trees being helpful with annotation or not. An additional limitation worth noting is that there will be error propagation due to an experimental error that stems from imprecision of the measurements of the reference substance that serves as the mass of the root node in an initiator tree. The consideration to compute the maximum possible error accumulation was that we will not have as many additions for heavier repeat units thus the error accumulation is lower for these repeat units. Hence in order to investigate the limit of the error of such an approach particularly in higher mass ranges, we want to compute the maximum possible error accumulation which would stem from additions and subtractions of $H2$. This error accumulation might often make it impossible to use the complete precision in ppm range that is needed to unequivocally annotate a given mass and that is the benefit of using the high precision measurements of the FTICR. Errors in addition add in quadrature i.e. for a combination of summations and differences the error is defined as $\Delta Q = \sqrt{\Delta a^2 + \Delta b^2}$ where Δa and Δb are errors in the values onto which the arithmetic operation is applied (Hughes and Hase, 2010) (pp. 42). For example if we propagate the error in ppm. The maximum number of additions and thus the maximum error propagation will occur when conducting additions of the repeat unit $H2$. We constructed the all trees with 171 levels. Such that we would cover different mass ranges depending on the reference substance we used to construct the tree. This leaves us with $342/2.01565 = 171$ possible additions or subtractions. Assuming a worst case scenario with an experimental error of 0.000009 ppm which is constant and would be propagated through our arithmetic operations, we could end up with an error of 0.000117 at the last branch of the initiatortree. This error stems merely from the error in the experimental value of the roots monoisotopic mass, as we can assume the mass of the repeat units to be precise. This means that as our tree grows we will have lesser accuracy in the Treen resulting in a loss of accuracy of two to three significant digits at the last level of the tree. This will lead to difficulties when assigning the masses producing meaningful mass differences in the patterns to their corresponding chemical entities.

To work with the initiator trees we construct them first and save them in a .json format for future analysis. Once we discovered meaningful mass differences through our pattern mining algorithm we would like to infer which chemical substances correspond to the peaks with these particular masses. This is done by looking at the masses causing the relevant mass difference and search the tree once for the greater and once for the smaller mass. This search is done with a precision of 6 significant digits in order to not loose any precision within the measurement itself. We have three possible results while searching Initiator trees.

Firstly we can find a node that corresponds to the mass in the pattern, which would enable us to assign the sum formula to the m/z value corresponding to a peak in the mass spectrum. Secondly we can find two or more nodes that correspond to several peaks at distinct m/z values. The nodes found in the initiatortree would trace a characteristic sequence of a fragmentation path indicating the loss or gain of one of the repeat units. As an example Figure 2.3 shows a hydration reaction where $H2O$ is added.

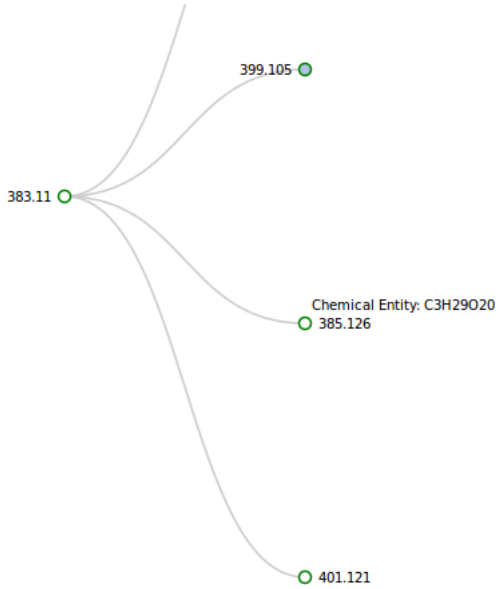

Figure 2.3: Figure showing an example path within an Initiatortree. One can see the transition from 383.110 to 385.126 by adding one unit of H2.

Regardless of the limitations in building a k-ary tree we should be able to find at least some nodes of the patterns in the initiatortrees. However because of the limitations in space complexity while building the tree, this search will presumably very rarely find pathways. Furthermore since the complexity of NOM substances and the processes of degradation can lead to substances that are in no way related to the reference substances it can also be the case that the sum formula can not be found because the extrapolation still depends on the mass of the root of the tree thus a particular reference substance. After a mass node corresponding to a mass in a retrieved pattern is found in one of the initiator trees, a subtree rooted at this node is extracted and can be visualized and inspected by the scientist.

2.3.3 Conceptual Summary of the Algorithm to Mine Frequent Patterns in Mass Spectral Data

It is worth summarizing the different steps included in this pipeline designed to analyse periodicities in mass spectra and find annotations for peaks in these patterns. We mine periodic patterns on the intensity domain which is commonly plotted on the y-axis of the mass spectrum. This approach is a new unsupervised method to explore mass spectral data and has not yet been employed as most algorithms commonly employ data on the mass axis. Since the x-axis can interchangeably be mass or frequency periodicity can be mined in both domains through such an approach. It is worth noting that scientists at Helmholtz Zentrum Munich are particularly interested in the frequency domain for which further analysis would be needed. However the exploration of this algorithm in

the mass space provides a first step towards mining similar patterns in frequency space. Because evaluation is only possible in the mass domain and due to time constraints we will use mass on the x-axis, which can be changed to be frequency if one would like to do so in order to compare relevant frequency differences with relevant mass differences and identify frequency multipliers that correspond to such mass differences. This section aims at providing a short overview over the steps that we elaborated so far.

1. In a first step, the set of given period lengths P is computed by the algorithm of Berberidis et al. (Berberidis et al., 2002a) explained in Section 2.2.1. This algorithm is applied to the data that is represented in a concept hierarchy in the data cubes T-slice containing binary vectors representing each bin and retrieves periodicity hints according to a user specified threshold.

2. Input arguments for the maximum subpattern hitset algorithm are the set of periods of interest P, a minimum confidence threshold and a Time Series S represented in a T-slice of the data cube which contains the discrete representation of the entire time series S. The T-slice of the data cube furthermore includes a hierarchical representation of the time series such as period segments S_i for a given period length p which effectively provides a split representation of the time series into segments of length p. For each of the periodicity hints given in P we apply the maximum subpattern hitset algorithm once. The algorithm retrieves the set of frequent one cycles $F1$ in a first step for each period. This can be seen as retrieving the set of frequently recurring peak heights in the mass spectrum for a given position in the period segment of the data series. In the next step it builds a representation of the maximum periodic pattern $Cmax$ that can be built using these periodically recurring peaks. Subsequently the algorithm looks if any manifestation of this maximum pattern is found as a hit in S_i. In the case that there are no frequent one-cycles found for a given period length the algorithm terminates and continues onwards to the next periodicity hint i.e. the next period length found by the previous algorithm. If there is no further period length it returns the mined patterns containing masses with peaks of periodic intensity.

3. In a post processing step after the patterns are found they are filtered according to their potential interest in the chemical sense by computing the mass differences between masses in these patterns. If a mass difference corresponds to an integer multiple of one of the repeat units in NOM space it is of interest and will be kept for further processing.

4. In a further step we move on to finding corresponding nodes in the initiatortrees with masses found in chemically relevant patterns. We try finding pathways that show the chemical reactions leading to the appearance of these periodic peaks using the initiatortrees. If no path is found, nodes can still be found and help with the annotation of masses with corresponding chemical entities.

The code for this data annotation pipeline is provided on github [5].

[5] see: https://github.com/trummelbummel/MSfingerprinter

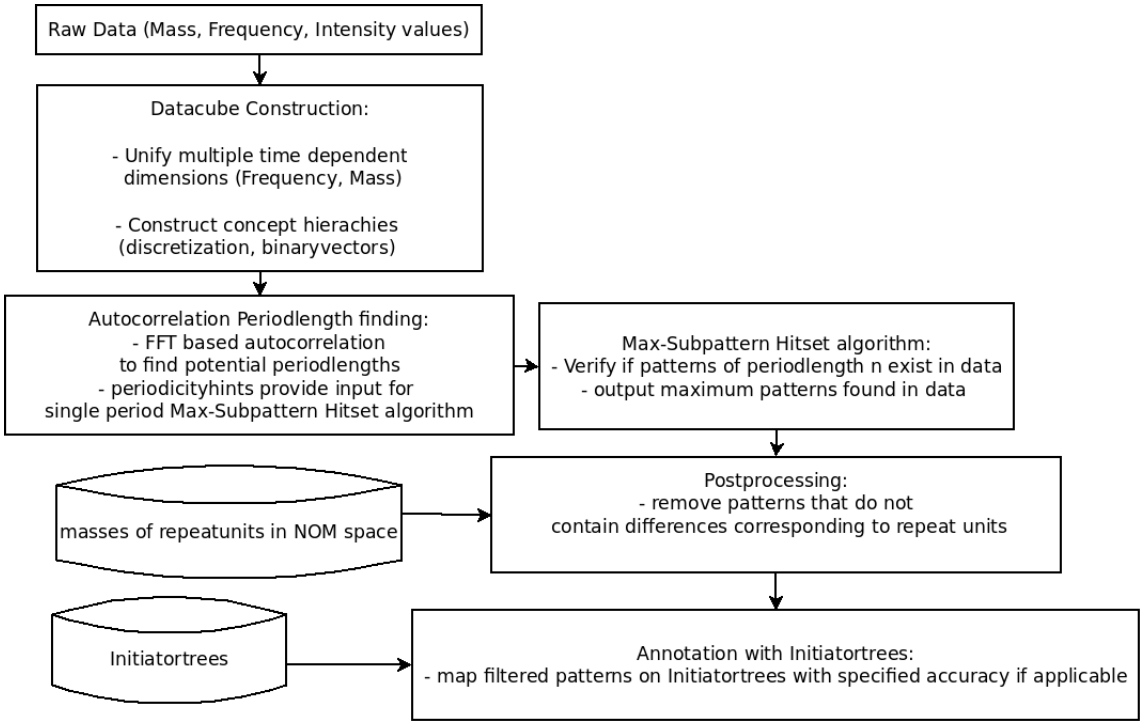

Figure 2.4: This figure shows a conceptual flowchart summary of the algorithms design.

2.3.4 Evaluation of the Results of the Pattern Mining Algorithms

Information retrieval systems are commonly evaluated using the notion of relevant and non relevant retrieved items. Relevance is assessed by judging the users need for a particular type of information which makes it then possible to do binary classification into relevant and irrelevant patterns accordingly (Manning et al., 2008) (pp. 151-176). Once we retrieve patterns that correspond to a number of peaks at m/z ratios we need to make judgement about their chemical relevance. The relevance of our patterns can be inferred by computing the mass differences between masses within a single pattern. Mass differences should correspond to any integer multiple of one of the repeat units in the context of this study. Each pattern that contains such a mass difference will be considered a relevant item, whereas patterns that do not contain any such differences will be considered irrelevant items. This simply translates to a binary classification of our mined patterns as relevant or irrelevant. There is no ground truth for the patterns available, as this is the first application trying to find exact periodic patterns in mass spectra. However for the purpose of this study the closest we could come to a meaningful ground truth with respect to the time constraints of this thesis, was to use any mass differences found between masses of the original mass spectrum that correspond to an integer multiple of the mass of any of the repeat units. This can be done by computing mass differences of any mass with all other masses of the mass spectrum using a double for loop, very similar to the procedure explained in Section 2.3.1. Shifting the data array one by one and computing the mass difference to all other masses in a brute force manner. After the difference was obtained the modulus of this difference and all repeat

units were taken. We assumed an accuracy of ppm for the measured masses i.e. we tolerated an error window of 0.000009 when taking the modulus of the mass differences, in order to not throw out any meaningful mass differences. It is worth noting that is not an optimal procedure but sufficient for the verification of our patterns. After filtering out duplicates a total of 9487 relevant mass differences remained. This means that in an optimal case there would be patterns retrieved that cover all 9487 relevant mass differences. This procedure will likely give a very conservative measure for recall. The intuition behind this statement is, that it is not guaranteed that all the mass differences we record in such a way actually stem from periodic patterns. And it remains up to the results of this thesis if it is likely that periodicity explains chemically relevant mass differences. There is still a possibility that such mass differences simply do not stem from any periodicity that can be observed on the intensity dimension. Thus it is possible that the periodic pattern mining algorithm "rightfully" ignores these mass differences. However it will give us a true estimate of how many mass differences that are chemically relevant exist in the mass spectrum used for mining periodicity

Information retrieval systems are evaluated using two basic evaluation criteria according to Manning et al. (2008):

1. Precision which is defined as the fraction of retrieved patterns that are relevant, thus contain a chemically meaningful mass difference.

$$Precision = \frac{\#relevantitemsretrieved}{\#allitemsretrieved} \tag{2.1}$$

2. Recall is the fraction of relevant mass differences that are retrieved from the raw data using the periodic pattern mining approach.

$$Recall = \frac{\#relevantitemsretrieved}{\#relevantitemsexistinginthedata} \tag{2.2}$$

   Both these measures commonly trade off against each other. You could achieve recall of 1.0 by retrieving all possible patterns, however this might come at a price of lower precision. And usually precision is the measure that if high implies relevance of the patterns for the user and thus in our case would imply that periodicity on the intensity dimension indeed could explain the occurrence of peaks that are chemically relevant (Manning et al., 2008) (pp. 151-176). Furthermore precision can be computed using the data retrieved by the algorithm itself, because we are able to efficiently group patterns into relevant and irrelevant using mass difference computation on the patterns retrieved. On the opposite hand recall needs the ground truth in order to be evaluated and hence it appears to be the more difficult measure to evaluate for our dataset. This is why we resort to a recall measure that takes mass differences instead of periodic patterns into account. Once the patterns that included chemically relevant mass differences have been extracted. Recall can be computed using both the ground truth and the unique mass differences occurring in the patterns we retrieved and found to be relevant.

2.4 Exploring Mass and Frequency Space - Sonification and Beyond

2.4.1 Entropy in Mass and Frequency Space

In order to look at differences between the mass and frequency space it would be interesting to have an objective measure of their quality for discrimination before involving sonification. According to Rényi et al. (1961) (pp. 547) Shannon's measure of information entropy defines the "amount of uncertainty of a probability distribution describing our data or a part of our data, that is, the amount of uncertainty we have about the outcome of an experiment." Hence it is a measure that has its origin from information theory and describes the states a system can take on (Rényi et al., 1961) . It could serve as a baseline knowledge about the ability to use the raw data in frequency space as opposed to the raw data in mass space in discrimination tasks such as for example clustering and ultimately also retrieval based on the spectrogram. The information contained in a mass spectrum are individual peaks with their respective intensities and they present the features that contribute to clustering results. The investigation of which features contribute to clustering mass spectra is important to understand which dimensions of the data are relevant for clustering and which could potentially be omitted to reduce the dimensionality of the data.

The outcomes of experiments i.e. our random variable is the realization of our clustering and can be seen as random variables which follow a certain probability distribution according to observed frequencies i.e. probabilities of certain values f_i a clustering result can take on (Dash and Liu, 2000). Even though we have information on the clustering results at present this might not always be the case in mass spectrometry and sample sizes in mass spectrometry are in general rather small such that there are few instances to be clustered. This is often achieved using hierarchical clustering, where we start with all samples being treated as a separate cluster and successively merge these clusters according to rules specific to the type of hierarchical clustering we chose (Corpet, 1988). It follows that a general version of the Shannon entropy is often not completely straightforward because the computation of $log(p(f_1, ..., f_M))$ needs the probability density and thus the realizations of our random variable i.e. the outcome of the clustering.

If we look at a single feature by itself will only contribute to the clustering if it is not uniform distributed, but exhibits values in a range, hence entropy depends on the data distribution. If entropy is at its maximum the uncertainty in this particular features space is high and clustering will be difficult to achieve based on this feature because of a small content of information in it. It follows that after the removal of feature F_1 which contributes more to clustering than F_2, we would get $E - F_1 > E - F_2$, where E is the total entropy before the respective feature was removed (Dash and Liu, 2000). In case of clustering data of red and white wines, we would expect features that contribute to a good clustering result to take on values in at least two separate ranges. Our our feature space is numerical we will use the euclidean distance to subsequently determine the similarity $S_{i1,i2}$

using a kernel function that represent norm-based distances in Hilbert spaces as the generality of the ranking algorithm requires such a transformation (Scholkopf, 2001; Dash and Liu, 2000). Similarity and Distance measures are inversely related. That is when similarity decreases, distance as a dissimilarity measure increases. One way to compute similarity measures that is used in case of the algorithm at hand is to use a kernel function. Kernels specify the inner products between high-dimensional points and are used to describe similarity between objects. They are generalizations of the canonical dot product which, considering its geometrical interpretation, can be seen as a similarity measure as it computes the cosine of the angle between unit vectors (Srebro, 2007; Scholkopf, 2001). Although strictly speaking euclidean space would be sufficient for our purpose, the generality of the ranking algorithm employed calls for a generalization into Hilbert space in order to serve all kinds of features not just numerical features. Now it also becomes clear why we represent the kernel in Hilbert space as the data has to exist in some dot product space that we will map our feature space on (Arbib, 2003) (pp. 1120). Kernel measures then can be used for generalizations of dissimilarity measures such as distances and enables us to turn them into similarity which is done using the maximum entropy state of a Gaussian similarity function (Scholkopf, 2001).

After computing the similarity $S_{i1,i2} \in (0, 1)$ using the Kernel trick equation 2.3 can be applied to compute the entropy according to Dash and Liu (2000):

$$E = -\sum_{i=1}^{N}\sum_{i=1}^{N}(S_{i1,i2} \times \log(S_{i1,i2}) + (1 - S_{i1,i2}) \times \log(1 - S_{i1,i2})) \qquad (2.3)$$

In practice we need to align our data such that corresponding masses are aligned. Because our data has different numbers of total features recorded for each sample we need to re-sample the data. Missing values need to be either interpolated in order for us to compare the functions, or we can simply fill them with the nearest neighbouring data point. While interpolation is an interesting option for filling missing values in our function it could potentially contribute to changed values of entropy biasing the entropy actually present in our data. Although filling missing values with the value of its nearest neighbour will probably reduce the entropy it will provide a conservative measure of entropy for our data (Pluim et al., 1999).

After missing values are filled in and our data is aligned appropriately we need to compute the entropy. Direct computation of the entropy might be difficult from the small sample of datasets often obtained in FTMS studies. This is why it would be advantageous both for applications where cluster labels for samples are unknown a priori and for small datasets to use the feature ranking technique described in (Dash and Liu, 2000), which makes use of similarity between samples. These findings could help reduce the dimensionality of the data which helps as we could reduce the data for uniform distributed features do not contribute to clustering results which would in turn lead to increased computational efficiency through data size reduction. However because our dimensionality is often quite large computing distances resulted in equidistant objects in our high dimensional space. This effect is well known as the curse of dimensionality (Hinneburg and Keim, 1999). Hence

we decided to segment the data and evaluate always five features at once, moving successively through the data evaluating the individual features compared to their nearest neighbours in terms of the sequence of the data. This means we are forced to take sub-samples of our features to estimate entropy while at the same time trying to get an accurate estimate for our features contribution to the entropy. We take the best or worst feature for every five features investigated and add it to a separate entropy distribution respectively in order to compare the best and worst case scenario for our data. We implemented the feature rank algorithm described in detail in Dash and Liu (2000) giving results that will serve as a measure to investigate how mass and frequency space compare with respect to their ability to discriminate mass spectral data. This should serve as baseline knowledge as to if there are differences between these spaces that need to be considered when working before going on to sonify the data. However it is important to note that the original frequency data will not be used for sonification. Rather a further processed version of the frequency data i.e. the relative frequency data will be used for the purpose of sonification as will be explained in the following section.

2.4.2 Sonification of Frequency Data

The raw data given was characterized by mass values, corresponding frequencies and corresponding intensities for each data point. The data used for these experiments were Mass Spectra of different red and white whines a summary of the class labels of the data can be found in Table 2. a more detailed characterization of the data is also given in Section 2.5.

For the sonification of frequency data the raw data needed to be processed in a way that transformed the data into an audible frequency range. The audible frequency range of humans ranges from 20 Hz to 20 000 Hz. Music commonly is represented as a continuous signal, i.e. they audio waves we are able to hear with our ears are continuous (Rosen and Howell, 2011) (pp. 163). Because computers are finite machines these continuous signals need to be digitized, hence represented as a sampled version of the continuous function also known as digital signal. A musical signal is commonly sampled at 44100 Hz according to the Nyquist sampling theorem. The Nyquist sampling theorem states that in order to maintain the signal representation without aliasing we need to sample two data points per period (Smith et al., 1997). In a first preprocessing step noise removal has been conducted. This implied that peaks were removed peaks that had a first significant digit after the comma that was at least 10% of the magnitude of the value before the comma. This corresponded to the removal of peaks with a negative mass defect where the Kendrick mass defect expression according to the mass defect analysis explained in Section 1.2.1 would yield negative values. The data transformation into the audible domain was achieved by forming relative frequencies within one nominal mass i.e. all components of one integer mass. Within each nominal mass the highest observed frequency corresponding to the lowest mass was taken as a baseline and the absolute difference was taken by subtracting this frequency from all other frequencies within this particular nominal

mass. The absolute value of these frequencies were serving as a basis for the engineering of sine waves to construct signals for sonification. Subtracting the highest frequency from itself would have resulted in a frequency of 0 rendering it inaudible. Hence I decided to set any frequencies below 20 Hz value to a default value of 20 Hz. This causes some loss of information in the data as the number of states frequencies can take on describing one nominal mass will decrease, as the number of 20 Hz components increases. This difference approach resulted in a pattern of audible relative frequencies describing each nominal mass. Each audible frequency obtained by the previously explained difference approach was transformed into a sine wave representation specified by the frequency and an amplitude value. The amplitude value for the sine wave was taken to be the original intensity value specified in the raw data at a particular frequency value. Linear combination of frequencies were formed that resulted in a complex digital signal describing one nominal mass. These complex waveforms served as a basis for sonification.

Before sonification can be done, the environment needs to be set up in an appropriate way [6] as sonification demands a lot of resources. Under Linux Ubuntu 14.04 OS [7] installing a low latency Kernel with a 1000Hz timer frequency is necessary. Because creating audio from MIDI audio software needs to run at a higher priority as it is a very memory and CPU intensive task. Only physical working memory and no swap memory may be used for sonification, such that one needs to configure an audio group to handle audio in real-time locking any necessary amount of memory in order to keep the synthesizers soundfont in physical memory. Soundfonts[8] contain recorded audio samples of musical instruments and are mapped by the synthesizer according to the specifications in the MIDI file. Because this is a very memory intensive process usually external hardware such as external sound cards are used in order to speed up the process, this may be necessary for any large scale application of sonification. However for the purpose of these experiments the specification of my Lenovo Y410p machine with 8GB RAM were sufficient .

Sonification of data can be done in two steps. Firstly the superimposed sine waves, that is a complex signal, have to be transformed into MIDI format. This was done using a python implementation MIDItime [9] which is specifically designed to map numerical time series data to information on pitch, duration and velocity values that can be interpreted by a synthesizer to produce audio.

Secondly the midi data was transformed to audio data and saved to a .wav format by using Fluidsynth[10]. Fluidsynth is a cross platform synthesizer which can be called as a sub-process by other programming applications in order to automatize the sonification. Fluidsynth thereby plays the role of a sampling synthesizer that uses the sound fonts recorded sample and produces sound according to the information stored in the MIDI file by modulating the samples in the sound font appropriately.

[6] see: http://tedfelix.com/linux/linux-midi.htmlpreliminaries)
[7] see: https://www.ubuntu.com/
[8] see: http://www.synthfont.com/sfspec24.pdf
[9] see: https://github.com/cirlabs/miditime
[10] see: http://www.fluidsynth.org/

2.4.3 The Fast Fourier Transform

In order to construct a valid representation of the mass spectral data we would want to make sure that the original functions constituents that were used as a basis for sonification, are represented appropriately by the superimposed sine wave such that all information remains intact when linearly combining the sine waves within one nominal mass. In order to do so the Fourier Transform is a tool to retrieve the components in the frequency domain that constitute a complex signal represented in the time domain.

The Fast Fourier transform (FFT) is a more efficient implementation of the Discrete Fourier transform (DFT), where signals are mapped from the time domain into the frequency domain. This is done by estimating the fourier series' coefficients. It describes the signal in the frequency domain as a sum of cosine and sine waves with different frequencies and amplitudes with the help of Euler's formula. FFT is used in many domains of engineering and sciences to compute the constituent frequency components of a complex signal. The complexity of computing the DFT is $O(n^2)$ stemming from the fact that for n frequencies there are n terms. However using the fact that the exponential terms in the DFT are multiples of one another and reusing them the computational complexity can be reduced to $O(n \log n)$ in FFT implementations (Singleton, 1967).

nominal mass	relative frequency component
93	20
93	65.32
93	67.23
95	20

Table 2.2: Summary of the raw data used for constructing wave forms for sonification.

An example of a simple wave constructed by only one frequency value for the nominal mass 95 (Echezeau) and a more complex wave resulting from a linear combination of sine waves for nominal mass 93 (Echezeau) of a wine sample are given in Figure 2.5 and Figure 2.6. Table 2.2 shows the raw data that went into the construction of these signals. The top of the image shows the constructed wave form resulting from a linear combination of sine waves determined by the relative frequency and the corresponding intensity in the time domain. The bottom of the image shows an FFT that recovers the original frequency and intensity components in the frequency domain to show that the linear combination of the sine waves does not introduce any artefacts. In Figure 2.5 one can observe that the sine wave only contains one single frequency component of 20 Hz indicated by the peak at 20 Hz. Whereas the superimposed signal in Figure 2.6 is composed of three different frequency components. We would expect there to be 3 peaks. This is true even though the two peaks around 65 and 67 seem rather collapsed. The data in the time domain serves as input for the sonification procedure outlined in Section 2.4.2. This also gives us some interesting indication. There is danger

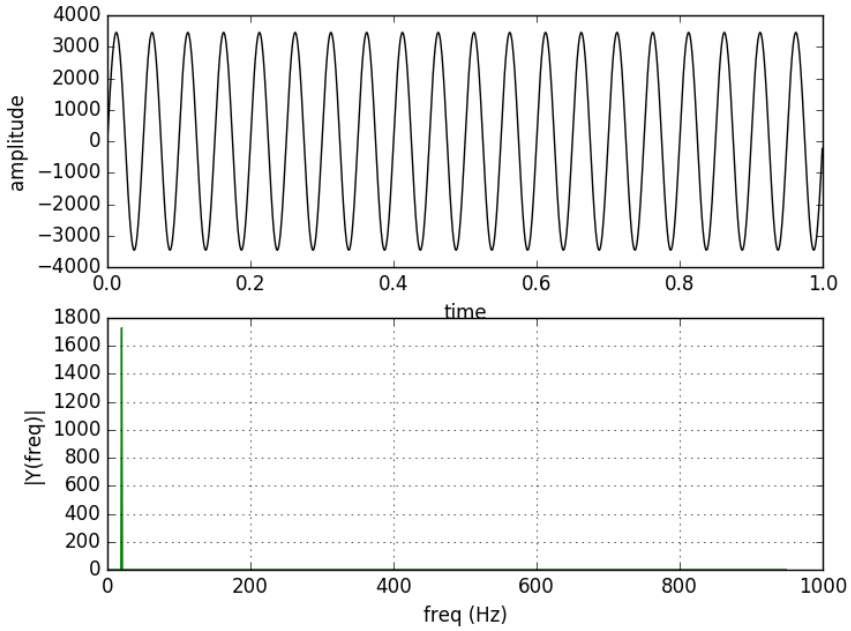

Figure 2.5: Sine wave and FFT of nominal mass 95 with only one 20 Hz frequency component.

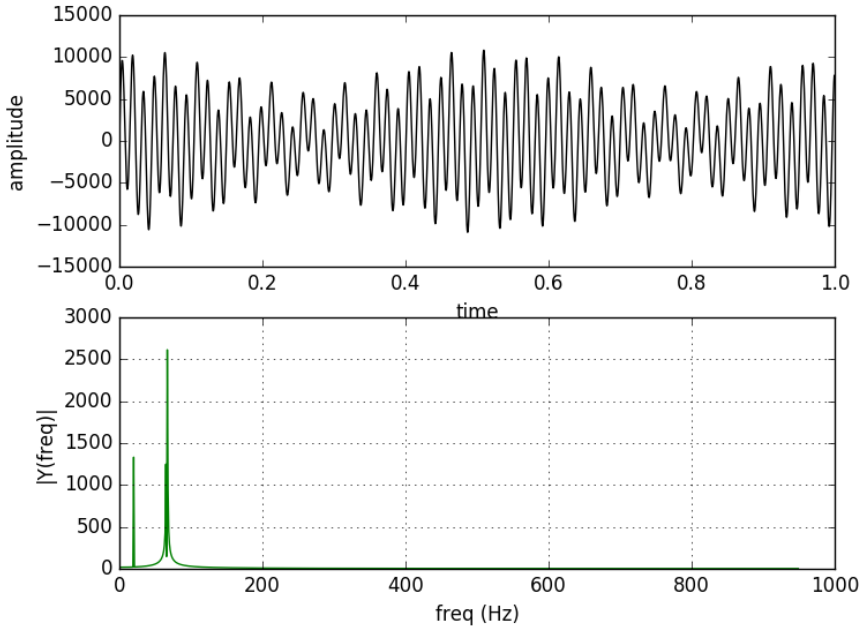

Figure 2.6: This figure shows a sine wave and FFT of nominal mass 93 with multiple distinct frequency components resulting in a complex wave.

of collapsing different frequency components in spectral analysis in the relative frequency space if frequency resolution of the chosen method is poor.

It is worth noting that we already encountered the FFT in this thesis as a more efficient version for computing the CAF as described in Section 2.2.1. We will furthermore need the FFT to compute the spectrogram matrix columns through windowed FFT in the next section.

2.4.4 The Spectrogram

The comparison of sound samples in terms of distance, can be achieved by using the Spectrogram matrix as a basis for the comparison. Furthermore spectrogram matrices serve as a basis for the hashing mechanism used for storage and retrieval of audio data as outlined in Section 2.4.6 in databases. Being the result of a windowed FFT with a window size being a power of 2 in order to obey the FFT's requirement on an input data series. The spectrogram matrix itself combines the information encoded in the time and frequency representation of the signal(Havelock et al., 2008) (pp. 449-481).

Usually the spectrogram matrices are a series of short term FFTs which are computed successively moving along the time axis of the signal in the time domain. The Spectrogram encodes three dimensions, for each window and show how these dimensions change over time. These three dimensions are:

(a) the time dimension, conventionally at the x axis position.

(b) the frequency dimension, conventionally the y axis

(c) the amplitude or energy dimension of the signal is conventionally encoded using a specified color coding, where the energy of each FFT bin is converted into color positioned at the appropriate time and frequency.

(Havelock et al., 2008) (pp. 449-481)

If the spectrogram of the sonified wave is computed using a narrowband differences on the frequency domain are more pronounced. This approach involves computing the FFT over a relatively small window. The spectrogram shows the typical trade-off where a narrowband approach can be used to infer the very fine grained frequency content of the signal, while a broadband approach infers the more fine grained changes over time and thus a trade-off between time and frequency resolution is given. For our task we would like the frequency content to be evaluated carefully while the time domain does not really influence the data as the frequency content of a file remains the same throughout the entire audio file (Havelock et al., 2008) (pp. 449-481).

The spectrogram representation furthermore makes it possible to compute a distance measure for the sonified data and the superimposed sine waves respectively and compare these distances. Distance computation was necessary in order to gauge if the sonification process that were present in the superimposed sine waves would be preserved or attenuated as a result of sonification of these superimposed signals. If this would be the case sonification would render the representation of the data more spurious and subsequent statistical tasks such as for example clustering or the application of algorithms commonly employed in the musical domain

will be more difficult. Because we need to have a frequency resolution below 20 Hz in order to capture frequency content of 20 Hz we chose a FFT size of 4096 which is a broadband approach that especially focuses on changes in the frequency domain rather than the time domain. The frequency resolution can be computed by considering the maximum frequency content in our signal which is 22050 at a sampling rate of 44100 that was used to construct the digital signal. The maximum frequency signal is then computed by taking the number of bins for the analysis window. That is our analysis window is divided into 2048 bins with an FFT size of 4096. Which yields a frequency resolution of 10.76. Spectrograms with this resolution will be serving as a basis for comparison in Section 2.4.5 and 2.4.6.

2.4.5 Dynamic Time Warping

Dynamic Time warping (DTW) is used as a tool in many disciplines ranging from biology, chemistry and data mining to speech and sound recognition as representative sub-fields of Engineering (Turetsky and Ellis, 2003; Kogan and Margoliash, 1998). It's main purpose is to provide a method that enables the computation of distances in time series even if the data can not be completely aligned because for example timing and offset of certain parts might be disparate. As sound is fundamentally one such time transient and DTW is often used to compute the distances in sound files it is the method of choice in this thesis (Turetsky and Ellis, 2003). DTW tries to find the optimal alignment between two time series in order to compute a distance measure. This is made possible by a mechanism that allows for stretching and compressing the signal to align them on the x-axis based on data on the y-axis. This is commonly known as warping and larger warping actions are discouraged as any warping action is linearly penalized (Salvador and Chan, 2007). If both signals would be identical along the time axis the euclidean distance would be sufficient to determine their distance and no warping would be needed (Salvador and Chan, 2007). However in case of distinct signals, alignment is not a trivial task and requires expensive computation that is only solvable via a dynamic programming approach. The dynamic time warping algorithm works as follows. Given two time series X and Y of distinct or similar length $|X|$ and $|Y|$ we want to compute the minimum cost path for the entire data series through a cost matrix D which is established while solving local problems. Because we have a condition on the indices of X and Y to monotonically increase and indices are bound to be between 1 and the length of the signals we will never compute the minimum distance for elements that lie "temporally" past the already computed indices while filling D. The local problems in this dynamic programming approach represent finding the minimum cost path coming from neighbouring positions that are accessible in one step. After D is completed the last element of the matrix contains the minimum cost of all paths between time series X and Y. It is then possible to find the minimum cost path using a backtracking approach (Salvador and Chan, 2007; Dixon, 2005).

In my case the time series two types of spectrogram matrices were under investi-

gation. One type represented as spectrograms from the sonified nominal masses, the other type of spectrograms was produced using the original signal representation as superposition of sine waves. This approach should establish if distances in the original signal representation are maintained after the sonification of the original signal. The alignment score that results from DTW i.e. the minimum cost path through the cost matrix can be used to estimate differences between these representations. DTW effectively gives a distance measure between the two data series, thus can also be used as input for for classification purposes.

DTW is an algorithm that also incorporates some disadvantages. Firstly DTW has quadratic time and space complexity (Salvador and Chan, 2007). However as the purpose of DTW is merely to provide proof for the maintenance of distances between the sonification space and the raw data space, we considered this an acceptable method to compare signals. Secondly DTW would rather score a song with the same mean frequency than shape as similar because it is merely based on the value on the y-axis ignoring trends in the signal even though they are important if one looks at the signal as features with particular shapes rather than mere values of frequency and amplitudes (Keogh and Pazzani, 2001).

To prove the effectiveness of the sonification approach for our data we tried to first simulate data that resembles our dataset and test if differences in the raw data and the sonified data are detected by DTW and if they are detected what would be the functional difference between systematically varying the frequency or Amplitude of the sine waves and sound data underlying the spectrograms. All DTW computations were done using matlabs inbuilt function that has been implemented according to (Sakoe and Chiba, 1978). In order to understand the limitations of the DTW algorithm and it's insensitivity to changes in amplitude for both the superimposed sine wave data and the sonified data, we generated a synthetic superimposed sine wave similar to the ones we were constructing from the raw data. For this purpose both the sonified data and the raw superimposed sine wave have been represented as a spectrogram. These experiments showed that DTW is in fact sensitive to changes in the amplitude when considering an artificially created superimposed sine wave with frequency 555 and 100. Linear increase in the Amplitude values resulted in a linear increase in the distance computed by DTW as can be inspected in Figure 2.7. However this picture changes considerably, when the same sine wave with frequency 555 and 100 being transformed through sonification. Although variations in Amplitude were still visible they were much less pronounced as can be seen in Figure 2.8, and no clear linear correspondence could be observed. One can also observe that the magnitude of the distances computed by DTW are reduced by several orders of magnitude when computing DTW on the sonified data. This is what we will expect to see when computing distances on superimposed sine waves and their respective sonified counterpart using the real data. It furthermore implies that we will expect an attenuation of distances through the process of sonification, which possibly can be attributed to the black box transformation to MIDI or by applying the soundfont using the synthesizer.

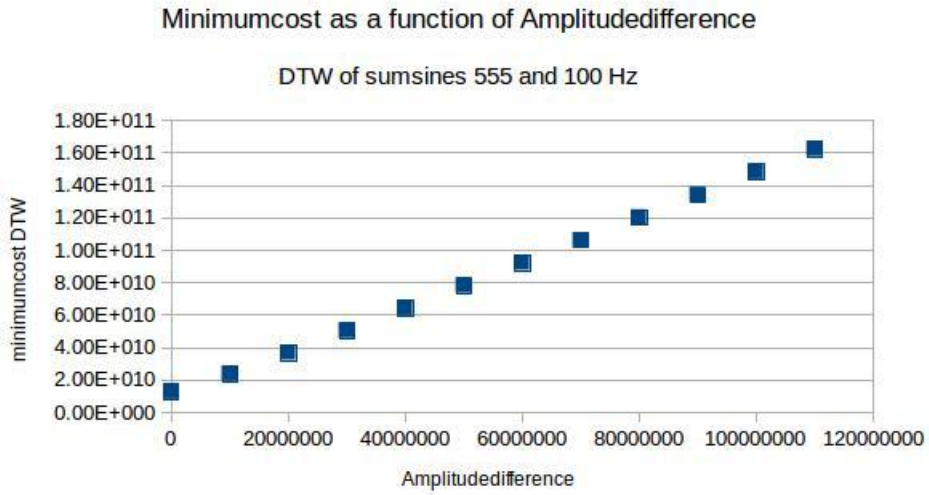

Figure 2.7: Changes in distance computed using DTW in superimposed sine wave Spectrogram representation for varying amplitudes.

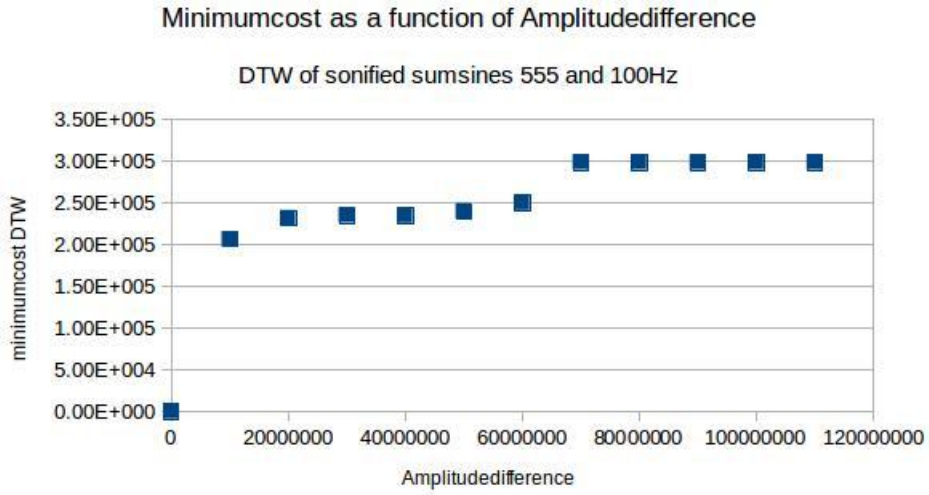

Figure 2.8: Changes in distance computed using DTW in sonified superimposed sine wave Spectrogram representation for varying amplitudes.

2.4.6 Audio Fingerprinting for Sonified Mass Spectral Data

Now that we have gathered the information of the audio files in a spectrogram we can continue onwards to use this representation providing a more condensed and robust description of the spectrograms features. According to similar reasoning as provided in Section 2.4.4 we choose an FFT size of 4096 which yields a frequency resolution of 10.76 thus covering a frequency range that includes the minimum frequency of 20 Hz. This is done by reading in all files from the directory and constructing spectrograms with the specified frequency resolution, which serve as a basis for the fingerprinting mechanism which in turn provides a basis for both,

the storage and retrieval mechanism. The algorithm tries to find robust constellations also referred to as "fingerprints" (Wang et al., 2003). Fingerprints are aimed at providing means for establishing equality between two objects features, using the smaller fingerprints instead of larger objects i.e. the spectrograms themselves. These hashes are generated by using the information of the audio contained in the frequency time domain representation of the sound contained in the spectrogram (Haitsma and Kalker, 2002). To fingerprint each Spectrogram we use a readily available implementation, that can be found on github[11]. It produces a fingerprint of each sonified nominal mass which we hoped to be sufficient to distinguish different sonified nominal masses. This part of our work should serve as a proof of concept that sonified data can be stored in and retrieved from a SQL database using a mechanism successfully employed in the musical domain. "Fingerprinting" is used to produce reproducible hashes, these hashes can be seen as an analogy to common cryptographic hashes where according to Haitsma and Kalker (2002) (pp. 1) "A cryptographic hash function allows comparison of two large objects X and Y , by just comparing their respective hash values H(X) and H(Y). Strict mathematical equality of the latter pair implies equality of the former, with only a very low probability of error." We are interested if "fingerprints" could be an equivalently viable option for storage of frequency based data in the scientific domain. If this approach works, it would presumably be applicable to many kinds of frequency data which can be represented as a spectrogram.

Wang et al. (2003) work illustrates that "spectrogram peaks with high amplitudes are considered robust against noise because they exhibit higher energy content than their neighbours." Furthermore candidate peaks need to be localized in a relatively dense region of the spectrogram containing actual musical content rather than for example a period of silence with one high energy peak compared to its low energy surrounding. This step transforms the spectrogram representation to a more condensed version of itself also referred to as "constellation map", where only high energy peaks in dense regions remain to represent the sonified nominal mass sample (Wang et al., 2003).

Now if a short audio segment is read from disk, recognition is achieved by sliding the constellation map of this shorter audio segment over the parts of the constellation maps saved in the database using hashes until a significant overlap is found between the queries constellation map and the database samples constellation map. However as databases grow, the recognition of such matches would be inefficient and an insufficient number of possible states the hashes can take on would be prohibitive for accurate retrieval. Hence a combinatorial hashing mechanism is employed to guarantee efficient storage and retrieval by increasing the entropy. These hashes are formed using a combinatorial association of time-frequency points from the constellation map (Wang et al., 2003) . To generate such combinatorial associations anchor points are chosen over the entire range of the spectrogram. Anchor points are again prominent peaks relative to their surrounding (Wang, 2006). From these points we define target zones, that is a segment in our constellation map that comes later with respect to the time axis. Then the algorithm requires pairing of frequency component of the anchor point

[11] see: https://github.com/worldveil/dejavu

with multiple frequency components in the target zone as well as recording the time difference between those points (Wang, 2006).

According to Wang (2006) (pp. 4) "The number of hashes per second of audio recording being processed is approximately equal to the density of constellation points per second times the fan-out factor into the target zone." The fan out factor F determines the number of peaks that are to be inspected in the target zone, thus a major factor in the computational complexity but also specificity of the algorithm (Wang, 2006). According to Wang et al. (2003) (pp. 4) "the combinatorial hashing squares the probability of point survival", thus reduces it as a single peak has probability p. This probability per definition will be below 1 and the probability of a pair of points to survive is a lower value of p^2. The probability of at least on hash surviving for a given anchor point is given by $p \times [1-(1-p)^F]$, and large values of the fan out factor will help us approximate the original probability of point survival according to this formula.
To search an unknown sample in the database, the same procedure of hash generation is applied to represent the sample. The hashes from the sample are used to search for matching hashes in the database. For matching hashes found in the database, offset times for the sample and the database files are associated into matching time pairs and distributed into bins. Hence as opposed to DTW that does not assume matching time points, this audio fingerprinting algorithm absolutely assumes such a direct linear correspondence. This implies that bins contain the matching sample and database pairs would be plotted on a diagonal line if the time axis would be plotted against each other (Wang, 2006). Hence after hashes are binned, bins are scanned representing each bin in a scatter-plot between sample and database sound files. If files match the sequence of hashes should be similar, as similar features occur at similar relative offsets from the beginning of the file. If enough relative time offsets match between sample sound file time and database sound file time a diagonal is formed in the scatter-plot representing matching hash locations which can be identified using any regression technique. The slope of this linear regression can be assumed to be 1.0 making identification more simple (Wang et al., 2003).

When reading files from disk there is no noise present in the environment, thus we expect a 100% success rate when recognizing files. The only time we expect there to be false positives is when the linear combination of sine waves that served as a basis for the sonification of the nominal mass would be exactly the same.

2.4.7 Conceptual Summary of the Algorithms used to Explore Frequency Space and Sonification

The aim to explore the frequency space of mass spectral data is two fold. Firstly as methods of analysis have not yet been applied to the frequency space it was important to identify any possible differences between mass and frequency space in terms of their information content. This is done by employing entropy analysis in order to verify that the raw data is equally suitable and provides data describing similar amounts of information.

Notably the frequency space provides the raw data that is subsequently preprocessed and converted into the relative frequency domain in order to provide data that will be used for sonification.
Once this verification is done we explore the difference between the summed sine waves that provide a basis for sonification and their sonified counter parts. For the sonification of mass spectral data a script has been employed for the preprocessing of the data converting the original FTICR frequencies into the audible domain to subsequently sonify it. This script can be found on github [12]. Before and after sonification we explore the mean distance between sonified nominal masses of the data to identify changes that could inadvertently occur through such a data transformation.
Once sonification is done we use the fingerprinting algorithm in order to understand if sonified data can be stored and retrieved efficiently identifying a mechanism that could potentially be used for any kind of frequency data and taps into a previously unused domain of applying algorithms employed in the musical domain to sonified data.

2.5 Data

The data used in this study has been kindly provided by the division of BioGeoChemistry at Helmholtz Zentrum, Munich. Frequency data corresponding to the m/z ratio of the mass spectra under investigation has been obtained using in-house software, effectively reconstructing the frequency data from the original time domain transient generated by the FTICR. For the exploration of frequency and mass space through modes of sonification, entropy evaluation and other methods described in Section 2.4.2 mass spectra describing different wine samples have been used. The data under investigation included Wine samples that have been acquired doing 500 scans on a Bruker(Bruker Daltonics GmbH, Bremen, Germany) solariX Ion Cyclotron Resonance Fourier TransformMass Spectrometer in negative ionisation mode with a time domain of 4 mega-word (Roullier-Gall et al., 2014). We have been provided 10 mass spectra of two easily separable kinds of wines. This included spectra of five red wines and five white wines summarized in Table 2.3. The frequency data of these wines served as the basis for the sonification as outlined in Section 2.4.2. Investigation of the separability of the data was subsequently done in both mass and frequency space using Entropy estimates and hierarchical clustering as outlined in Section 2.4.1.

For the pattern mining task NOM spectra where used as they exhibit periodicity much more than the wine spectra. The presentation of FTICR-MS data is in the form of reconstructed spectra as can be seen in Figure 2.9. Plotting their signal intensity against their m/z ratio, the periodicities occurring in one subset of molecules compared to another subgroup become visible. Mass lists of these spectra were exported with an accuracy of ppm i.e. with 6 significant digits. In order to provide different mass resolutions automatically we will segment the data, in a

[12] see: https://github.com/trummelbummel/sonificationscript

Name	Winetype
Echezeau	red
Chambertin	red
GevrezChambertin	red
ClosVogeout	red
Beaune	red
Lafon20	white
Lafon11	white
SMART	white
CortonC	white
Giardin	white

Table 2.3: Summary of the data used for frequency space exploration.

similar manner a chemist would do when zooming in and out to look at distinct nominal masses. In order to make different resolutions available to the periodic pattern mining algorithm we provide such segments of different lengths as input to the algorithm. It is furthermore important to note that the pattern mining algorithm itself does take a similar approach like the scientist that would have to zoom in and out of the data. This is why one aim of this study is to provide an automated way of extracting periodic events from mass spectral data, and see if these periodicities correspond to chemically relevant peaks in the data. Hence segmentation of the data was varied to see if this would influence the number of patterns and types of patterns retrieved. Segmentation implies that the number of data points that served as input for the periodicity hint algorithm and the periodic pattern mining algorithm respectively varies. I chose a data segmentation of 1000, 500 and 100 to investigate the effect of segmentation on the mining algorithm. Furthermore for comparison a global analysis of the entire 7933 data points has been done. A second parameter that has been varied and had influence on the data was the number of bins used in the discretization step.
The number of peaks recorded per nominal mass varies depending on the noise reduction strategy applied in data preprocessing. There will be less or more data points per nominal mass for more restrictive and less restrictive noise reduction strategies respectively. Noise reduction is usually done by using a peak picking algorithm that only leaves intensity peaks in the data that are beyond a certain threshold. In this case we did not use further noise reduction after extracting the mass-list with a total of 8439 m/z and corresponding intensity values from the calibrated mass spectrum, recorded with a time domain of 4 mega-word. Furthermore the mass range under investigation in the NOM sample ranges from 140 to 1073.

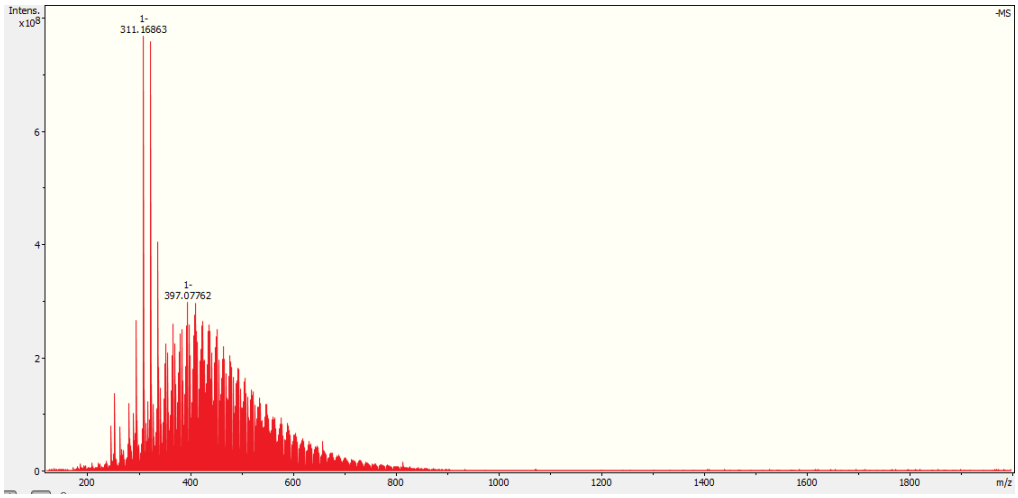

Figure 2.9: shows the Natural organic matter mass spectrum used in the pattern search procedure.

Chapter 3

Results and Discussion

3.1 Results Patternmining

Parametrization of different parts of the algorithms has been discussed in Sections 2.2.1, 2.2.2, 2.2.2 and 2.5. We will note here that runtime was generally dominated by the autocorrelation based algorithm for period length detection, and constructing the maximum pattern was not contributing a significant amount to runtime.

3.1.1 Recall and Precision

I was able to compute a ground truth containing all possible meaningful mass differences as has been explained in Section 2.3.1. This is assumed to be a very conservative estimate because our aim was to investigate if periodicity implied mass differences corresponding to one of the repeat units. In the raw data 9487 mass differences that correspond to an integer multiple of one of the repeat units were found, this was used as a ground truth to compute recall and thus the success of the modelling approach. A total of 125 unique chemically relevant mass differences were found in the patterns after duplicates were removed. The mass differences mined were in a mass range of 181 to 877 which indicates that we could cover most of the relevant mass range. Even though the mass range measured runs from 140 to 1073 there are only 10 data points before nominal mass 181 and 26 data points after nominal mass 877. The observation that periodicity can not be found at these higher and lower ranges in the spectrum is not new and has been observed as a consequence of the low number of peaks and thus a lower number of possibilities to observe periodicity (Yu et al., 2011). This indicates that when using all binning and segmentation scenarios were summarized we could extract periodicity from a large part of the data disregarding only few data points. However compared to the brute force approach where 9487 unique mass differences were found recall is rather low. Recall was 1.3% which might indicate that periodicity is not a good way to model the relevant data in a mass spectrum or that only part

of the relevant mass differences can be explained by periodicity in the intensity domain. Another possible explanation is that the performance of the algorithm was reduced because it was disturbed by noise. Thus an improved noise removal before periodic pattern mining is applied could improve the results. Noise might shift periodically recurring peaks from their ttruepposition in the periodic pattern, thus causing the algorithm to miss out on these patterns, and make it more difficult to for example detect frequent one-cycles.

For 7933 datapoints and 10 as well as 7 bins the shortest period length found was 251 and 278 respectively, hence because of limitations of the regular expression approach no periodic patterns could be retrieved when looking at the mass spectrum globally. For 5 bins the global approach yielded 2 patterns of periodlength 90 and both of them contained multiple mass differences that were relevant in a chemical context. Both patterns were predominantly mining the lower mass range from 165 to 305. Because of the fact that twice no patterns were retrieved we did not include the data in the visual representation of the results in Figure 3.1. For mining global periods it would be imperative to extend the approach to a regular expression approach that could handle more than 100 groups at once as the period lengths found by the CAF based algorithm are too long to be handled by our approach.

Precision was computed for different number of bins and depending on the segmentation of the data. The ratio of number of relevant patterns retrieved and irrelevant patterns retrieved for segmentation of 100, 500 and 1000 respectively depends on a complex interaction of the number of bins and the data segmentation this is why parametrization is difficult for our approach. This comes at a time where parameter free algorithms based on Kolmogorov complexity estimates are garnering increasing attention. Not least because of the fact that they provide user independent and often superior results (Keogh et al., 2004). However as per our knowledge periodicity detection in time series still lags behind in developing such algorithms. Hence we are left with the result that parameter choices such as for example the choice of data segmentation had influence on the precision, with 100 data points being the worst consistently throughout trials. This is predominantly caused by a retrieval of many irrelevant patterns as can be inspected in Figure 3.2 and 3.3. This indicates that chemically relevant periodicity in mass spectral data if present should be mined over longer stretches of data. A fragmentation into too many small segments might disrupt their detection as the peaks relevant to cause chemically relevant mass differences can be dispersed over longer stretches of the data especially in the face of noisy data. Furthermore for segmented data we could observe that the number of bins influences the number of relevant patterns retrieved, indicating a more complex dependency. There seems to be an interaction effect between the number of bins and the length of the data segmentation. A more global approach with larger chunks of the data for example using 1000 data points each has yielded the best results with precision being maximum in all but one binning and segmentation scenario when using a threshold of 0.2 or 0.8 for finding $F1$. The overall maximum precision has been yielded by a segmentation of the data into chunks of 1000 data points using a threshold of 0.8 to find $F1$ and 5 bins yielding a value of 40.9%. This indicates that the fraction of irrelevant pat-

terns for threshold 0.2 and 0.8 for retrieving $F1$ respectively changes in favour of relevant patterns with a higher threshold for larger data segments of 1000 points. However for smaller data segmentation of 100 and 500 a higher precision was yielded with the threshold 0.2, indicating that the retrieval of more patterns also implied the retrieval of disproportionally more relevant patterns. Hence it depends on the segmentation of the data as well as the binning which threshold yields better results. This interaction effect makes it difficult to choose optimal parameters. A lower precision in the 100 data points segmentation is predominantly yielded because disproportionally many patterns are irrelevant in a chemical context. Thus it remains to assume that periodicity in the intensity is not necessarily linked to chemical relevance. Further experiments will be needed where noise removal as a preprocessing step of the data is done in order to determine the efficiency of the approach. However due to time constraints this will be left open for future research efforts.

It is also worth noting that contrary to an intuition more bins do not improve the results. Because 10 bins should improve the precision as a wider dynamic range of the time series is covered. It has already been noted elsewhere that the binning of a continuous attribute has often large effect on the detection of periodicity (Parthasarathy et al., 2006). Since equal width histogram binning approach was used because of its simplicity the dynamic range might not have been maximized. Equal width binning is known to be sensitive to outliers (Gama et al., 1998), albeit simple it suffers from drawbacks such that some intervals may contain a disproportionally large number of values (Dash et al., 2011). This sensitivity to outliers might have caused some values to be under represented while others are over represented when using the highest number of bins. This can be true because as we can see clearly that one such outlier is the base peak in mass spectral data, it is often times of much higher intensity than the other peaks as is the case in the NOM spectrum where the base peak stands out as can be seen in Section 2.5. It would be desirable to conduct binning such that there would be an equal frequency of all values before doing mining tasks on time series. This can be achieved by normalizing the time series to have a mean of 0 and a standard deviation of 1. These normalized time series are known to have a Gaussian distribution and can then be divided into segments that contain all values with equal frequency using breakpoints that are specified for Gaussian distributions (Lin et al., 2007). This approach is preferable to equal width binning procedure which might have been a cause for the algorithm to deliver suboptimal results.

In order to identify other indicators for the claim that periodicity on the intensity domain is not necessarily linked to chemical relevance we employed a higher threshold of 0.8 for finding $F1$. Increasing the threshold reduces the total number of patterns retrieved relative to the lower threshold of 0.2 as expected. Generally setting the threshold for finding $F1$ to be higher improved the precision which is expected considering that we impose a stricter threshold on finding periodic one cycles that implies that less such one cycles are found and much less patterns can be formed and thus hit in the segments of the time series. This follows from our previous assumption that low precision is predominantly caused by the retrieval of many irrelevant patterns in the low threshold condition. Precision predominantly

increases because less irrelevant patterns are retrieved alongside relevant ones. It could also be seen as an indicator that if stricter requirements on periodicity are imposed we yield relatively more relevant patterns and thus periodicity is an indicator for peaks bracketing important mass differences. However because the highest value for precision is still low with 40.9% and this value seems to be depending heavily on the parameters chosen. More precisely it does so in 40.9% of the cases in the best case scenario and often in a lower relative number of cases as can be seen in Figure 3.1. It follows that the functional relationship of periodicity in the intensity domain should only be considered a good explanation for some part of the data's chemical relevance. This is also supported by low values for precision throughout all binning and segmentation scenarios even while imposing a stricter threshold for mining $F1$.

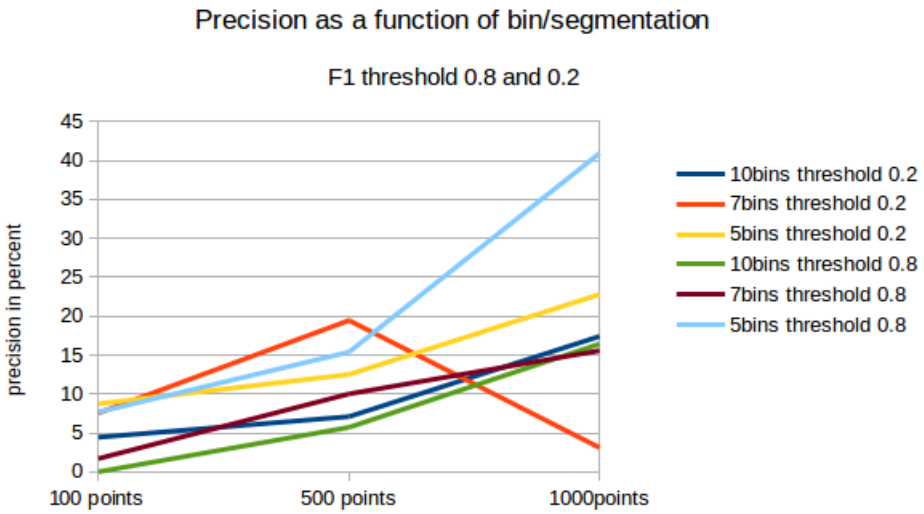

Figure 3.1: Precision as a function of number of bins for different sizes of datasegmentation for $F1$ threshold of 0.2 and 0.8.

Please note that the patterns mined might also contain duplicates in terms of the masses that produce a specific meaningful mass differences because different period lengths are searched on the same data segments according to the period lengths acquired by the CAF based algorithm. These different period lengths can contain partial patterns that are the same and thus the same masses which result in same mass difference are mined. This issue has been taken care of for the computation of the recall as only unique mass differences mined where considered after post processing. For the precision measurements however duplicates have not been eliminated as they were still considered distinct periodic patterns and only a subpattern matched. Even if subpatterns match they still are independently standing periodic patterns of a different length.

Because runtime was not extensive and the algorithm it would be good to run the algorithm in a loop with different parameter choices to mine as many patterns as possible. Even though the different parameter choices sometimes mine the same patterns they also sometimes mine distinct patterns, thus forming the union of

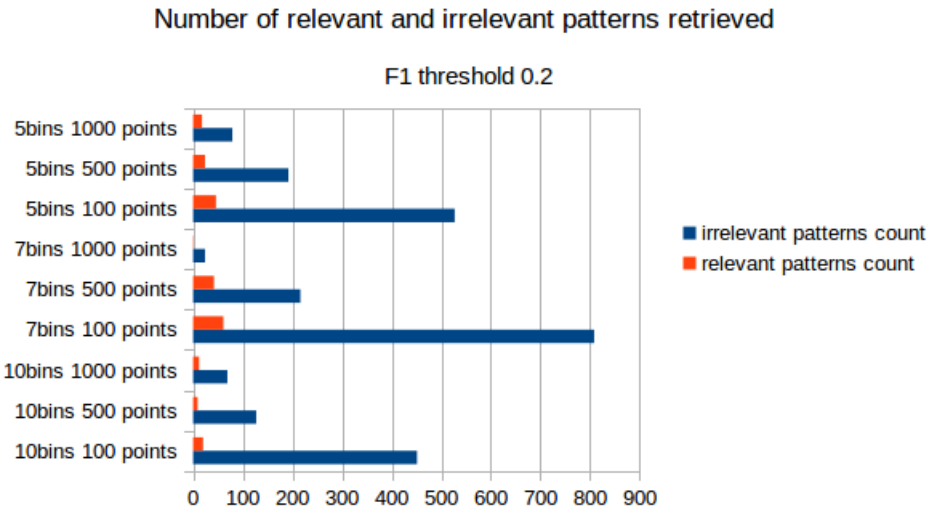

Figure 3.2: Shows the number of relevant vs. irrelevant patterns retrieved depending on the choice of datasegmentation, the number of bins and the threshold 0.2 to find $F1$.

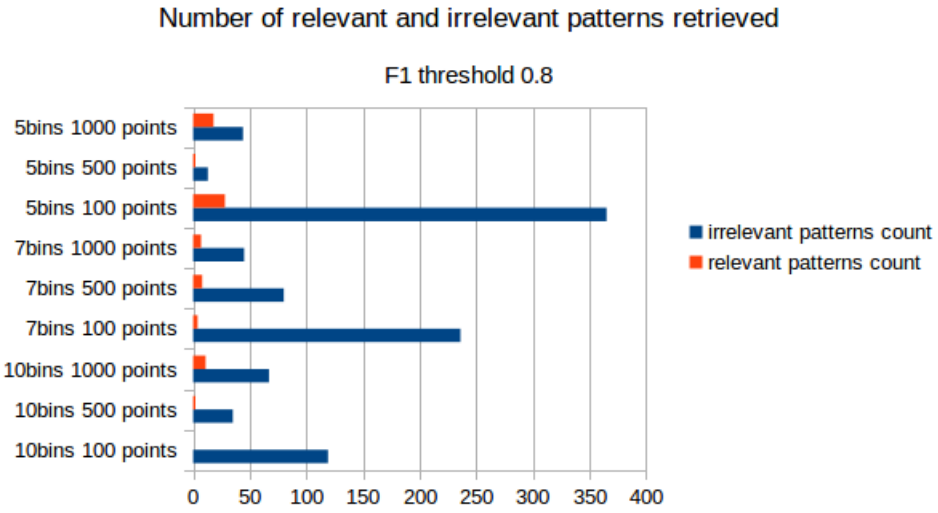

Figure 3.3: Shows the number of relevant vs. irrelevant patterns retrieved depending on the choice of datasegmentation, the number of bins and the threshold 0.8 to find $F1$.

the results will yield the maximum number of patterns possible using the algorithm at hand. Especially because of the difficulties and data dependency of the parametrization this is a more viable option to extract the maximum number of relevant mass differences using a periodic pattern mining approach.

These result leave room for improvement and indicate that the maximum subpattern hitset algorithm without any improvement is not a viable option to mine periodic patterns in mass spectral data. The problem here in my opinion is manifold, which leaves room for improvement and potentially better results. However it might be entirely possible that the use of periodicity is only good for noise removal and periodicity detection in biopolymer spectra (Wallace and Guttman, 2002; Yu

et al., 2011), while the mass differences in NOM spectra might not depend on the periodic appearance of peaks. This is potentially a valid assumption with respect to the results of the mining algorithm. Hence alternative models such as the mass difference statistics algorithm should be considered superior as of now.
We would want to explore some possibilities that might have been detrimental to the performance of the algorithm. Firstly the periodicity hint mining algorithm returns many very large periods which are inaccessible to my approach because the regular expression implementation in python allows only for 100 groups.

Regular expression provided us with an efficient mean to check for alternative events at one position, but comes at the cost of us being only able to detect periodic events with period length of 100. Often times however the periodicity detection algorithm returned many periods that were substantially longer, thus all these periods could not be explored with the current implementation.
Secondly detecting periodicities with the autocorrelation approach might have limited finding of periodic events because of the filter step. Another option would be to exhaustively check all possible period lengths, however again there is a trade off in computational complexity when doing so and this approach is rather inefficient for larger periods we expected to occur in the mass spectral data (Berberidis et al., 2002a). For shorter data segments this could be done as the algorithm runs terminates in linear time scanning the time series only twice for each period.

A binning procedure that automatically maximizes the dynamic range could potentially improve the results. Given a normal distribution of the time series we can produce a equal sized areas under the Gaussian curve, effectively binning the values such that they would have equal probability to occur. More elaborate symbolic representation will approximately contain equally values of the original data series in each bin. This basically equates to maximizing the dynamic range in our discrete representation, and selects an optimal number of bins, and is employed by more sophisticated symbolic representations of time series such as SAX (Lin et al., 2003).

This brings us to another argument where we could see scope for improvement of the algorithm. The segmentation of the dataset was unsupervised, i.e. it did not take any information on the chemical problem into account that could potentially constrain the possible periodicities. That is chemical knowledge only influenced the analysis of the patterns in the post processing steps but not any previous steps in mining these patterns. Hence data segmentation could have led to the splitting of periodic events that would otherwise be detected. I could have approached this problem by shifting the segmentation repeatedly, however this would have led to a combinatorial explosion. We would have to recompute periodicity hints and periodic patterns for each of the splits and this renders this approach infeasible. Furthermore segmentation presumably influenced which values were binned into one bin between segmentations, introducing a bias that is difficult to control for.
Again this was the first time an algorithm to mine exact periodic patterns has been applied to mass spectral data. Applying the maximum subpattern hitset algorithm to chemical data without modification does not yield optimal results, and it remains questionable if periodicity on the intensity dimension explains chemically relevant

data.

3.1.2 Annotation of Masses in the Patterns through search in Initiatortrees

A total of 125 unique chemically relevant mass differences were found in the patterns after duplicates were removed. These mass differences mined were potential candidates to be annotated using the initiator trees in this data mining pipeline. After patterns that are relevant are filtered and mass differences were extracted in a post processing step we can use the initiatortrees to search for annotations of the m/z value. When looking for nodes in the initiator trees we would want to find matches that are in the parts ppm range. This level of accuracy implies that we use the benefits of the accuracy of the FTICR in ppm without loss in precision which presumably makes annotation less equivocal. When using an accuracy of 6 significant digits 10 unique sum formulas were found and could be annotated. A summary can be found in Table A.2. These annotations corresponded to the annotations found by Bruker Data Analysis software that is used to inspect mass spectral data (Bruker Daltonics GmbH, Bremen, Germany).

For example the mass 309.061589 has been found in a pattern. When searching for the mass 309.061589 in the initiatortrees it was found to have the respective sum formula C14H13O8. Furthermore the sum formula is generated in the tree rooted at C22H19O12. Part of the resulting subtree extracted based on this node is shown in the Figure 3.4 below, and by hovering over the node one can see which chemical formula i.e. which chemical entity corresponds to the mass of the node.

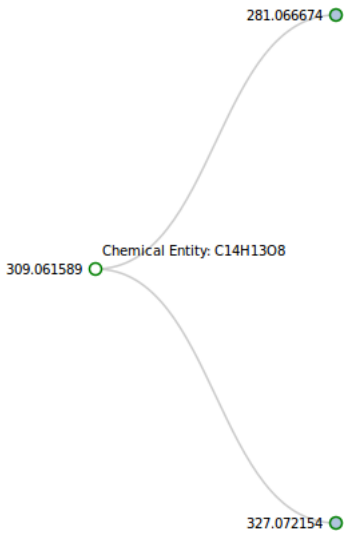

Figure 3.4: A node found using a mass from a pattern that can be annotated with the sum formula C28H29O18.

Generally patterns found by the periodicity mining algorithm and all patterns found 3.1.1 contained relevant mass differences that were within a mass range of 181 to 877. Considering the maximum error propagation of 0.00017 for H2 in the

171 operations used for the construction of the tree. It is entirely possible that some masses could not be annotated due to this error accumulation, considering that only 65 patterns were mass differences in the lower mass range till 400 of the spectrum, and the rest were mass differences found in the remaining mass range. Notably these initiatortrees can only cover part of the search space in NOM namely those depending on C, H and O components that are acquired through successive addition and subtraction of the repeat units containing C, H and O. Heteroatoms such as sulphur and nitrogen atoms where disregarded but do exist in NOM samples. However these heteroatoms should not have played a role as the determination of relevant mass differences already depended on identification based on repeat units only containing C, H and O atoms. However it is indeed possible to not find some mass peaks because they do not stem from a reaction based on the root of the initiatortrees and as such can not be found based on their extrapolation. Hence it is likely that initiatortrees can only help with annotation in specific cases where both the error accumulation is low and the mass can be produced by any of the roots of the trees.

Furthermore because we have 6 known repeat units in NOM space disregarding any nitrogen and sulphur containing compounds, we would have to build an k-ary tree. This is clearly not feasible in terms of its space complexity. Hence it is possible that we could not cover enough of the search space with the reduced version of the tree. Possibly a better approach would be to use context free grammar to generate a framework based on which one can derive the pathway leading from the reference substance to the particular mass by known rewrite rules that could incorporate all known repeat units including sulphur and nitrogen containing compounds and their possible transitions i.e. addition and subtraction. This approach would furthermore make it feasible to include heteroatoms such as sulphur and nitrogen. In a context-free grammar approach we could specify which sort of children are possible by only allowing for productions according to the rewrite rules specified in the grammar. This approach has already been successfully applied to other biological and chemical problems that are similarly intractable or have a very large search space. Examples are RNA secondary structure prediction and chemical substructure matching which usually is a problem of graph isomorphism and known to be NP hard (Dowell and Eddy, 2004; Proschak et al., 2007). It is possible that such an approach could cover more of the search space and would help in generating an automated pipeline that only needs mass differences as an input in order to find chemical substances and the ion responsible for these mass differences. However because of time constraints this issue will be left open to future research efforts.

3.2 Results and Discussion Exploration of Frequency Space

3.2.1 Entropy in Mass and Frequency Space

As mentioned above it is interesting how features in mass and frequency compare with respect to their ability to provide information for subsequent tasks that are based on such information. Because sample size is rather small computation of entropy based feature ranking was done as described in Section 2.4.1 based on computing the entropy for five features at a time we were able to extract a more optimal sub sample for the entropy by taking the minimum entropy value of each five features. However to get a general picture on which mass ranges contribute most to clustering and an estimate for the differences or commonalities of entropy in both mass and frequency space this will be sufficient.

The results are shown in Figures 3.5 and 3.6, . When we look at the distribution of the entropy values we can see that both spaces are characterized by a roughly equivalent entropy, which makes sense as there is an inverse non-linear relationship between mass and frequency space and that is used to map frequency values to the mass over charge ratio. Both distributions show a positive skew with most of the distribution being concentrated at low values of entropy and thus low uncertainty with respect to the classification which indicates a good ability to cluster based on the features in the data. However it is worth noting that Entropy in mass space is slightly better, however this difference is likely not noticeable in clustering applications. For the wine samples in mass space an average entropy value of 0.64 has been observed when using the best of 5 method. The variance of the entropy was 0.045. While in frequency space the mean entropy value was 0.66 and the variance was 0.035. When we change the strategy to choosing the worst of 5 features to define the distribution describing the entropy, the distribution shifts marginally. In mass space we have a mean of 0.79 and a variance of 0.11. While in frequency space we have a mean of 0.8 and a variance of 0.078. This indicates that both spaces are fairly equivalent in their worst and best case entropy estimates. And there should not be much of a difference when using either frequency or mass data for any subsequent statistical tasks on mass spectral data.

When we take a closer look at the entropy values, it becomes apparent that good low entropy features are found at similar positions of the mass spectrum and frequency spectrum. While high entropy values and thus high uncertainty when using these features for clustering are predominantly found in the lowest and highest mass ranges in mass space, the picture is less clear in frequency space as can be inspected in Figures 3.8 and 3.7. Hence when removing features that should not contribute much to statistical tasks, the picture seems to be clearer in mass space than in frequency space. As in mass space you could simply crop data points that are below or above a certain mass range. It is worth noting that indices given in the plot are roughly corresponding to the mass and frequency range as we progress through the data with increasing mass. However from an information point of view both spaces seem to be equivalent and thus roughly provide similar

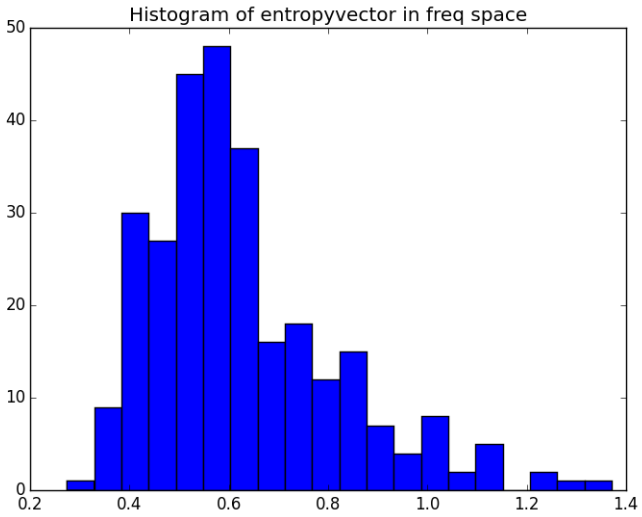

Figure 3.5: Frequency Distribution entropy values in Frequency space.

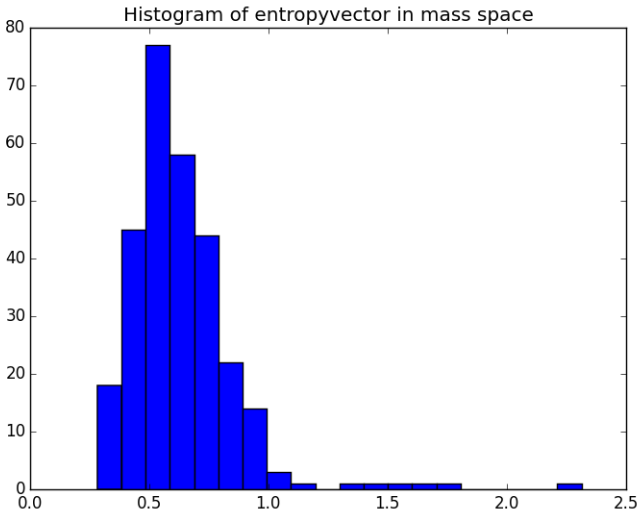

Figure 3.6: Frequency Distribution of entropy values in Mass space.

information overall. This is why we assumed frequency space to provide an equally good basis for sonification and subsequent tasks that use the frequency dimension for discrimination through fingerprinting.

It follows that frequency space and mass space both provide data with a similar information content even though the mapping that exists between these spaces is non linear. Both spaces exhibit an inherent order, the higher freq the lower mass, and this clear inverse relationship makes it possible to use both features mass and frequency space in a presumably equivalent way for subsequent statistical tasks.

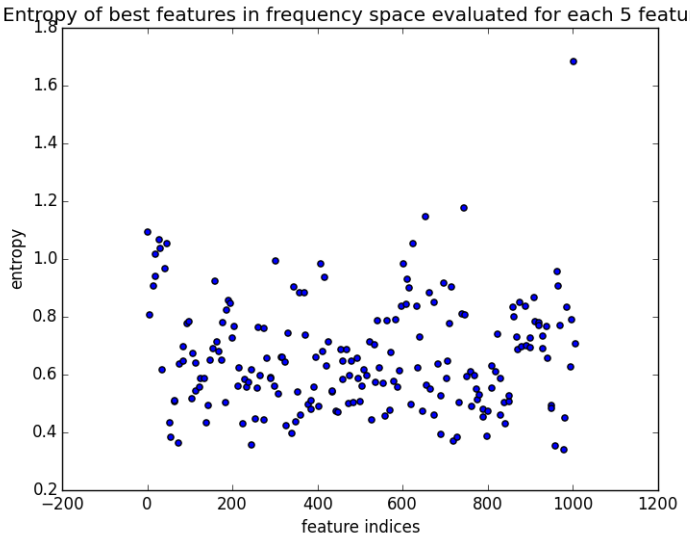

Figure 3.7: Scatterplot showing entropy values in frequency space. Indizes are indicating the progression in the mass range.

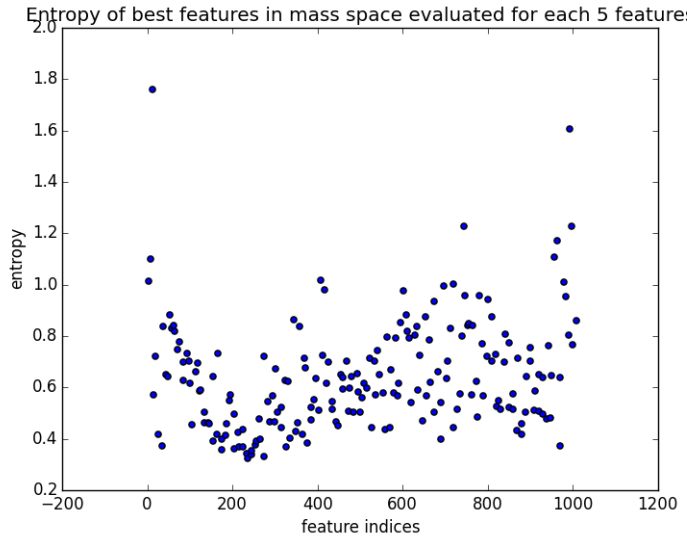

Figure 3.8: Scatterplot showing entropy values in mass space. Indizes are indicating the progression in the mass range.

3.2.2 Preservation of Distance between Raw Data and Sonified Data

As Spectrograms or at least the information contained in the spectrograms form the basis for features on which many music information retrieval algorithm would be operating the maintenance of distance between these spaces was crucial for any subsequent methods employed that would depend on the distances such as for example clustering. In this section we compared the superimposed sine waves and

their sonified counterparts distances using DTW. Importantly we wanted to see if distances are maintained after the transformation of the superimposed sine waves representing the nominal masses to sonified nominal masses. It is important to note that the superimposed sine waves themselves are not based on the original frequency data but on the relative frequency data as explained in 2.4.2. Mean distances between red and red wines and white and white wines i.e. distances between similar types of wines are smaller in both cases than mean distances between red and white wines. However the difference between the mean values for the sonified data are much less pronounced. This implies that it would be more difficult to distinguish the samples based on the sonified data, and there is supposedly no gain in sonifying the data as the distances are reduced by sonification. These results correspond to our expectations after looking at the results of DTW for our synthetic superimposed sine wave data and the sonified data. However distances are still pronounced enough to distinguish between the distinct samples and the similar samples when looking at the data, one can see that distances between similar samples become more homogeneous and all similar type samples show a mean distance of roughly 300.

Comparison	**n**	**mean - sumsine**	**mean - sonified**	**variance - sumsine**	**variance - sonified**
ClosVogeoutEchezeau	704	186.2	325	66964	12113
ClosVogeoutGevrez	682	228.3	363	75215	9152
EchezeauGevrez	683	168.4	322	62207	27659
GiardinLafon	155	463.53	318	121503	3534
ClosVogeoutLafon	254	697	619	96377	34611
EchezeauLafon	175	685	605	94529	26520
EchezeauGiardin	181	673.89	598	114415	28798
GevrezGiardin	175	656.37	609	112134	27924
GevrezLafon	155	678.15	602	88851	26713
ClosVogeoutGiardin	177	675.2	619	120537	34611

Table 3.1: Shows a summary of the statistics on the results of the distance computation using dynamic time warping. Mean/variance sumsine shows the mean/variance distance of the linear superposition that served as basis for the sonification. Mean/variance sonified shows the mean and variance of the distance computed on the sonified data.

The histograms in the appendix Figure A.1 and A.2 show an example of a more detailed version of the above table. One can observe that when comparing the distances computed between the nominal masses of the superimposed sine waves many more values close to zero are observed as opposed to a similar comparison made between a white and a red wine. When comparing ClosVogeout (red wine) and Giardin (white wine) the distribution is shifted to the right indicating higher distance values as computed using DTW on the spectrogram representation of these distinct wines nominal masses as compared to a distribution showing the distances of two red wines. The number of nominal masses that could contribute to these histograms varies according to the presence of the same nominal masses in both samples. Since white and red wine share much less commonality in this example they share only 181 nominal masses less distances can be computed within these samples, thus statistics are less reliable and it would be desirable to repeat

the analysis with samples where distinct samples share more common nominal masses such that a larger sample size could be used. The values computed can be considered lower bounds of the distance as all nominal masses that are not present in one of the wines would contribute a value to the distance that can be regarded as infinite, however this would skew the comparison unnecessarily and for the purpose of comparing the preservation of distance it would not be beneficial to include these values.

One can observe that the trend in distances between the superimposed sine waves represented as numerical data and the sonified superimposed sine waves in .wav format are maintained when computing the distance based on their respective spectrograms. However the distances seem to be attenuated in the sonified data. Similar to our example in Section 2.4.5 distances in the spectrograms based on the superimposed sine waves were more pronounced than in the sonified data. Problematic is also that the variance seems to be reduced in the sonified version of the data which can be detrimental to subsequent statistical tasks. For example in clustering approaches, that could use distances computed by DTW, we try to preserve the variance of the data after clustering primarily in order to not distort the informativeness of the data. However if the informativeness of the data is already reduced before clustering, this would make accurate clustering based on the sonified data all the more difficult. This can be attributed to the fact that usually variables that contribute to data separation are considered variables with large values of variance, as they will also increase the variance within a cluster and contribute to splitting if necessary (Jain et al., 1999).

3.2.3 Results Audio Fingerprinting Storage and Retrieval of Sonified Data

After the fingerprinting database was established based on spectrograms that provided sufficient frequency resolution using the method outlined in Section 2.4.6, we tried recognizing the files saved in the database.

First we tried to identify the files with a fan out value of 15 for the fingerprints. Of 462 files 64 were recognized as a different file with these settings. As a first criterion we looked at the number of frequencies that defined a sample, once that number was distinct the sample immediately qualified as a false positive i.e. the recognition as equivalent was unsubstantiated based on the data. If the number of frequencies in a sample was the same we looked at the cause of the difference. If all frequencies were equal it was in fact a true positive as a false negative that was retrieved because of its objective mathematical equivalence. It follows that if the algorithm would have retrieved all not just one answer to the query we would have also retrieved the correct sample from the database. Another possibility emerged from a query sample having the same number of frequencies factoring into the linear combination of sine waves and largely quite similar frequencies. One Example of such a case is given in Table A.1 in the appendix. Analysis revealed that the samples that were retrieved incorrectly were samples that did not contain the exact same frequency content but had very minor differences that might have

become more spurious during the process of sonification. Often times they were the same sonified nominal masses of for example two distinct redwines, that did contain minor differences. After manual inspection we identified 63 false positives. This meant that the retrieval accuracy of our algorithm is as follows. With a total of 462 queries, 63(13.63%) were wrongly retrieved and 399 samples (86.37%) were retrieved correctly which is below the 100% retrieval rate we expected when reading from disk. Because it often happened that only few frequencies differed in the wrongly retrieved samples we concluded that the specificity of the algorithm i.e. the fan out value chosen might not have been sufficient for correct retrieval. Hence we increased the fan out value which should provide more accuracy. And indeed a fan out value of 20 improved the accuracy to 100%. Only one sample Echezeau nominal mass 95 and Lafon11 nominal mass 429 was retrieved incorrectly, however this sample had the same frequency content of 20Hz, hence the equivalence in the hashes was given by an objective mathematical equivalence of the data. These results were what we would expect an accuracy of 100% from the algorithm when reading from disk. Hence with an a FFT size of 4069 to generate the spectrogram and a fan out value of 20 the results were optimal.

This result indicates that sonification could potentially be used to efficiently store and retrieve data that can be sonified. This would provide an efficient mean using an industrial algorithm from the musical domain. However one has to note that the sonification process increases the size of the data. Hence a more viable approach is to work with spectrograms that are based on the superimposed sine waves rather than the sonified data. The benefit lies not only in the reduction of the data size but first and foremost also in the notion that distances are not attenuated by sonification. In such an approach one could construct spectrograms with each nominal mass being a column in the spectrogram. Hence each column would contain a FFT of a superimposed sine wave describing a nominal mass that does not suffer from the drawbacks of sonification. In such a representation a more broadband approach should be used for spectrogram construction, as the time dimension in the spectrogram then becomes meaningful. The use of each nominal mass as a column of the spectrogram would likely increase the specificity of the representation as there would be an increase in the variance of the data describing one sample and similar content occur with a smaller probability presumably increasing the entropy of the spectrograms features. Since the algorithm works on the basis of the spectrogram retrieving the constellation maps for different samples this approach would open up the possibility of working only with numerical data and as such circumvent the necessity of sonification. The benefit of this approach would go beyond an information theoretic view of the data. As sonification causes a large increases with respect to the space needed for data storage a spectrogram constructed from the numerical data representing the superimposed sine waves would provide a much more succinct representation of the data. It remains to note that we could not exclude conclusively the possibility that sonification introduced artefacts, and thus the specificity of the sonified data's spectrogram might be downscaled by the introduction of noise through the application of MIDI and further digital reduction of the data's information content. However because of time constraints this route was not investigated and will be left open for future

efforts.

Furthermore one drawback of this approach resides on a different level and needs to be noted for mass spectral data specifically. FTICR-MS data depends not only on the cyclotron frequencies but also on their harmonic integer multiples that can differ depending on the specifics of the instrument. One such specific is the ICR cells used to trap ions and measure the mass to charge ratio. The geometries of these cells lead distinct noise patterns also known as harmonic peaks in frequency space (Tolmachev et al., 2008). Hence the method to store and retrieve frequency based data has presumably more potential in applications where the dependency on machine specifics is not that pronounced.

Chapter 4

Outlook

As stated in Section 1.4 this was the first application of the maximum subpattern hitset algorithm to a new problem that was chemical in nature and not a time series in a common sense. Thus improvements can be made to gather more meaningful information from chemical data. To improve the current results of the periodic pattern mining algorithm more information on the chemical problem could be included that could lead to improvements for example through a more informed segmentation procedure. Furthermore improved binning methods that try to find a balance between the dimensionality reduction through the binning procedure and maintaining most the original dynamic range at the same time should be investigated as equal width binning has been used for simplicity but might have delivered suboptimal results.

Unfortunately without sufficient chemical background the evaluation of the periodicity mining approach is difficult as it still depends on the known repeat units in NOM space. Whereas an analytical chemist could potentially find more relevant peaks in the periodic patterns mined by this unsupervised approach, it remains up to the experienced analytical chemist to judge their relevance. Due to time constraints this avenue could not be evaluated and the evaluation of peaks other than the ones corresponding to a mass difference of a known repeat unit will be left open for experts in the field. Because of time constraints however the approach taken in this thesis was to verify if periodicity on the intensity can be used to extract meaningful patterns in the mass domain and the extraction of frequency patterns will be left open for future research. However it is worth noting that a simple change of the x-axis data to mine the frequency data and compare the patterns mined in the frequency domain with corresponding patterns mined in the mass domain. If one analyses the patterns that were chemically relevant in the mass domain and compares them to the patterns in the frequency domain it could lead to insight on frequency multipliers corresponding to known repeat units in mass space. The evaluation of periodicity mining on mass spectra of origin other than NOM remains also an open avenue. Because periodicity has been found to occur in such spectra it might be a viable option to test the periodicity mining algorithm on such samples. Because of the unsupervised nature the implementation of the maximum subpattern hitset algorithm is equally applicable to other

datasets without modification.It remains to be seen if the approach could be applied to other types of mass spectral data such as spectra of biopolymer. However post processing steps would need to be adapted according to the distinct chemical specifics of the problem.

Because of the space complexity that exceeds the capacity of present day machines the Initiatortrees should rather be built with context free grammar. However even using the current version of the initiatortrees some peaks could be annotated and verified. Thus the inherent logic seems to provide a helpful tool for annotation and an implementation in terms of context free grammar could reap the whole benefits of such a logic. Initiatortrees however are a visual extension of already known methods such as the mass difference statistics algorithm (Kunenkov et al., 2009).

Sonification should happen in the original frequency domain to avoid needing to transform the data as relative frequency space was only necessary to provide a basis for audible sonification. Actual sonification of the data is not necessarily the best way to reap the benefits of the audio fingerprinting mechanism as it again causes a large increase in the volume of the data. Since more and more areas of science are looking to store their growing amount of data efficiently, this might be a viable approach to store and retrieve any kind of frequency data that is not dependent on specifics of measurement instruments in databases. The algorithm itself is quite powerful in distinguishing data, and any frequency data can be represented in terms of a spectrogram with an accuracy depending on the users need. It follows that this might be a fruitful research direction for research areas that are looking for a storage and retrieval algorithm of their frequency based data.

Appendix A

Appendix

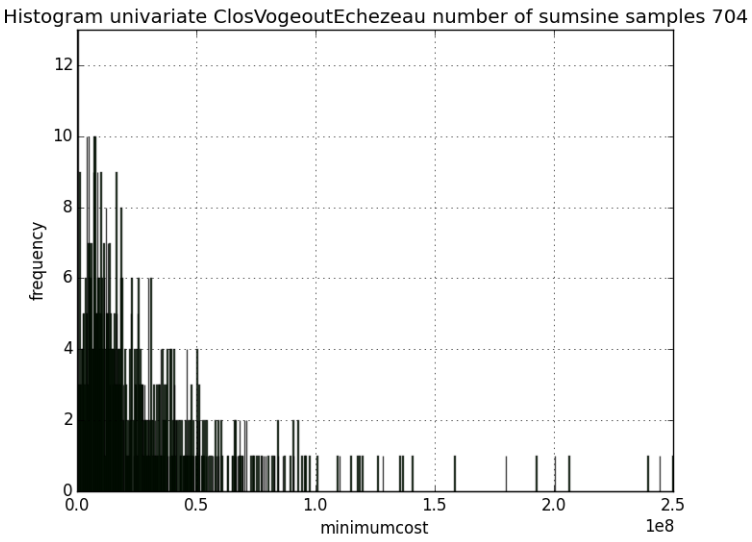

Figure A.1: shows a histogram of the distances between two red wines ClosVogeout and Echezeau.

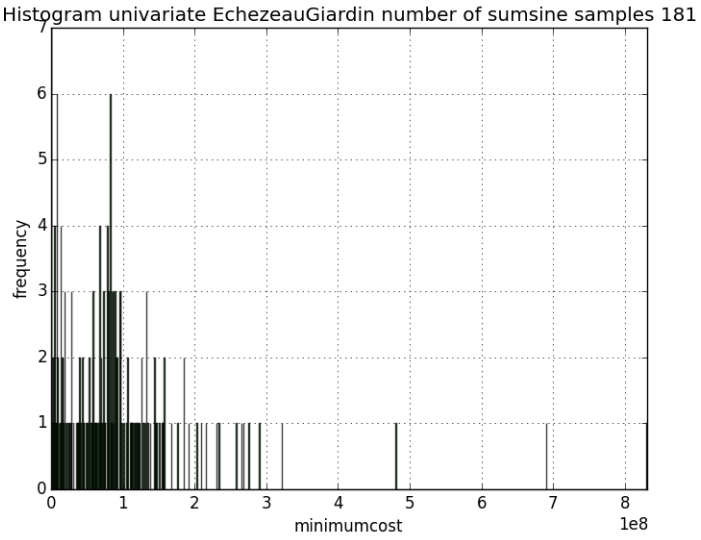

Figure A.2: shows a histogram of the distances between two red wines ClosVogeout and Giardin.

relativefreq sample1	**relativefreq sample2**	**absolute frequency difference**
20	20	0
20	20	0
22.411	22.411	0
26.702	27.179	0.477
30.993	30.993	0
33.378	n/a	33.378
53.405	53.405	0
78.678	78.678	0
113.487	n/a	113.487
126.361	n/a	126.361
134.468	n/a	134.468

Table A.1: Shows an example of a wrongly retrieved nominal mass and the absolute frequency differences between the query and the retrieved nominal mass.

Initiatortree	**Mass**	**Annotation**
C8H9O6	307.045942	C14H11O8
C8H9O6	387.056902	C15H15O12
C8H9O6	653.208712	C30H37O16
C14H11O6	597.088597	C28H21O15
C25H17O13	499.109329	C21H23O14
C22H19O12	307.118709	C16H19O6
C19H13O10	307.009555	C13H7O9
C19H13O10	803.094865	C34H27O23
C22H19O12	309.061589	C14H13O8
C27H19O13	391.030684	C17H11O11

Table A.2: Shows a list of masses of the peaks we could annotate with a sum formula using the Initiatortrees.

Abstract

In mass spectral data analysis mass space which is a projection of the originally recorded data in frequency space has been investigated in much more detail than the frequency space. This is necessitated by the fact that so far annotation of peaks with their corresponding chemical substances commonly happens in mass space. Frequency space should in theory be equally successful in providing information to group data according to its features as there is a known non linear relationship between mass and frequency space. Firstly we would like to investigate how mass and frequency compare with respect to entropy i.e. their information content and thus their ability to serve as features in statistical and data mining tasks. Due to the nature of the Mass spectral data that is available in frequency space and in mass space it would lend itself to sonification which makes it possible to employ algorithms commonly used in music information retrieval. Thus this thesis investigates the benefits and drawbacks that emerge by mapping frequency data into the musical domain for further analysis. This includes an investigation into the maintenance of distance between superimposed sine waves that are derived from the original frequency data as well as their sonified counterparts employing Dynamic time warping. In this thesis we furthermore investigate the use of data mining techniques commonly used in areas of time series analysis for finding periodic patterns. The algorithms employed enable finding partial periodic patterns in mass spectral data in both mass and frequency space. The main contribution is the investigation of the suitability of these algorithms that investigate periodicity in time series analysis on mass spectral data that exhibits periodicity on a mass spectrum coming from a sample of natural organic matter (NOM). The goal of the approach is to find pattern such that they include parts that correspond to mass differences of known CHO containing repeat units in NOM which are known to be chemically relevant. The pattern mining algorithm under investigation terminates in linear time and should focus specifically on retrieving periodic patterns that we hope point towards chemically relevant datapoints. Furthermore I present novel tree based computational approaches for the annotation of the chemical species found in such patterns. These trees can be seen as a visual representation of the well known Kendrick mass defect analysis and the mass difference statistics algorithm. Using reference substances that are used for the calibration of Mass spectra, these initiator trees are build and include subtraction and additions of known repeat units corresponding to the fragmentation patterns commonly observed in NOM CHO space.

Zusammenfassung

Massen Spektrometrische Daten werden herkoemmlicherweise im Frequenzbereich aufgezeichnet. Derzeit finden jedoch die meisten Daten analytischen methoden im Massen Bereich statt, da dies der Merkmalsraum ist in dem die Annotation von chemischen Substanzen mittels Summen Formel Bestimmung stattfindet. Der Frequenzbereich dieser Daten ist daher weit weniger exploriert sollte jedoch theoretischer weise gleich erfolgreich sein Information fuer das gruppieren von Daten bereitstellen da es eine bekannte non lineare funktionale Beziehung zwischen Massen und Frequenzen gibt. In einem ersten Ansatz wollen wir in Erfahrung bringen inwieweit diese Annahme berechtigt ist und Massen sowie Frequenz Merkmalsraum in ihrer Entropie und damit ihrem Informationsgehalt uebereinstimmen. Dies dient dazu die Merkmalsraeume in ihrer Faehigkeit Merkmale fuer weitere statistische Aufgaben bereitzustellen zu beurteilen. Da Daten die im Massen Spetrometer aufgezeichnet werden natuerlicherweise im Frequenz Merkmalsraum bereitstehen wuerden diese Daten exzellent fuer die Sonifizierung geeignet sein. Dies wuerde es widerum erlauben Algorithmen auf Massen Spektren anzuwenden welche herkoemmlicherweise nur in der Musik Domaene angewandt werden. Diese Masterarbeit befasst sich daher auch mit den Vorteilen und Nachteilen einer solchen Datentransformation vom Frequenzbereich in die musikalische Domaene. Dies inkludiert eine Analyse der Erhaltung von Distanzen zwischen ueberlagerten sinuskurven welche als Basis fuer die Sonifizierung dienen, als auch ihrer sonifizierten Gegenstuecke. Diese Analyse wird durch den Einsatz von Dynamic Time Warping durchgefuehrt. Ein weiterer Teil dieser Arbeit beschaeftigt sich mit der Untersuchung von “data miningTTechniken welche normalerweise in der Zeitreihen Analyse angewandt werden um periodische Muster in den Daten zu finden. Die Algorithmen die in dieser Arbeit angewandt werden sollten es ermoeglichen exakte periodische Muster in Massen Spektren sowohl in Massen als auch im Frequenz Merkmalsraum zu finden. Die Untersuchung der Anwendbarkeit solcher Algorithmen auf Massen Spektren von Proben bestehend aus Natuerlicher Organischer Substanzen ist hier das primaere Ziel. Weiters wird versucht die gefunden Muster auf ihre chemische Relevanz hin zu pruefen. Falls diese Pruefung positiv erfolgt sollten die periodischen Muster Massen Differenzen enthalten welche mit bekannten Massen Differenzen uebereinstimmen die bei der Fragmentierung von CHO enthaltenden Einheiten uebereinstimmen. Der hier Untersuchte Algorithmus terminiert in linearem Zeitaufwand und ist darauf ausgelegt periodische Muster zu extrahieren die wie wir hoffen auch chemisch relevante Massen Differenzen enthalten. Weiters wird ein neuer Baum basierter Annotations Vorgang gezeigt, welcher als eine Digitalisierung des in der

Massen Spektrometrie weit verbreiteten Kendrick Massen Defekten gesehen werden kann. Durch die Nutzung von Referenz Substanzen welche fuer die Kalibierung von Massen Spektren verwendet werden als erster Datenknoten und die Addition und Subtraktion von bekannten Fragmenten kann so ein Teil des Suchraums zur Annotation von Natuerlichen Organischen Substanzen abgedeckt werden.

Bibliography

Agrawal, R., Srikant, R., et al. (1994). Fast algorithms for mining association rules. In *Proc. 20th int. conf. very large data bases, VLDB*, volume 1215, pages 487–499.

Amster, I. J. et al. (1996). Fourier transform mass spectrometry. *Journal of mass spectrometry*, 31(12):1325–1337.

Arbib, M. A. (2003). *The handbook of brain theory and neural networks*. MIT press.

Barner-Kowollik, C., Gruendling, T., Falkenhagen, J., and Weidner, S. (2012). *Mass spectrometry in polymer chemistry*. John Wiley & Sons.

Berberidis, C., Aref, W. G., Atallah, M., Vlahavas, I., and Elmagarmid, A. K. (2002a). Multiple and partial periodicity mining in time series databases. In *Proceedings of the 15th European conference on artificial intelligence*, pages 370–374. IOS Press.

Berberidis, C., Vlahavas, I., Aref, W. G., Atallah, M., and Elmagarmid, A. K. (2002b). On the discovery of weak periodicities in large time series. In *European Conference on Principles of Data Mining and Knowledge Discovery*, pages 51–61. Springer.

Cao, H., Cheung, D. W., and Mamoulis, N. (2004). Discovering partial periodic patterns in discrete data sequences. In *Pacific-Asia Conference on Knowledge Discovery and Data Mining*, pages 653–658. Springer.

Chen, J., Gu, B., LeBoeuf, E. J., Pan, H., and Dai, S. (2002). Spectroscopic characterization of the structural and functional properties of natural organic matter fractions. *Chemosphere*, 48(1):59–68.

Cook, R. L. (2004). Coupling nmr to nom. *Analytical and bioanalytical chemistry*, 378(6):1484–1503.

Corpet, F. (1988). Multiple sequence alignment with hierarchical clustering. *Nucleic acids research*, 16(22):10881–10890.

Dash, M. and Liu, H. (2000). Feature selection for clustering. In *Pacific-Asia Conference on knowledge discovery and data mining*, pages 110–121. Springer.

Dash, R., Paramguru, R. L., and Dash, R. (2011). Comparative analysis of supervised and unsupervised discretization techniques. *International Journal of Advances in Science and Technology*, 2(3):29–37.

De Hoffmann, E. and Stroobant, V. (2007). *Mass spectrometry: principles and applications*. John Wiley & Sons.

Dixon, S. (2005). Live tracking of musical performances using on-line time warping. In *Proceedings of the 8th International Conference on Digital Audio Effects*, pages 92–97. Citeseer.

Domon, B. and Aebersold, R. (2006). Mass spectrometry and protein analysis. *science*, 312(5771):212–217.

Dowell, R. D. and Eddy, S. R. (2004). Evaluation of several lightweight stochastic context-free grammars for rna secondary structure prediction. *BMC bioinformatics*, 5(1):71.

Easterling, M. L., Mize, T. H., and Amster, I. J. (1999). Routine part-per-million mass accuracy for high-mass ions: space-charge effects in maldi ft-icr. *Analytical chemistry*, 71(3):624–632.

Eng, J. K., McCormack, A. L., and Yates, J. R. (1994). An approach to correlate tandem mass spectral data of peptides with amino acid sequences in a protein database. *Journal of the American Society for Mass Spectrometry*, 5(11):976–989.

Gama, J., Torgo, L., and Soares, C. (1998). Dynamic discretization of continuous attributes. In *Ibero-American Conference on Artificial Intelligence*, pages 160–169. Springer.

Greiner, D., Lindstrom, P., Heckman, H., Cork, B., and Bieser, F. (1975). Momentum distributions of isotopes produced by fragmentation of relativistic c 12 and o 16 projectiles. *Physical Review Letters*, 35(3):152.

Haitsma, J. and Kalker, T. (2002). A highly robust audio fingerprinting system. In *Ismir*, volume 2002, pages 107–115.

Han, J., Cheng, H., Xin, D., and Yan, X. (2007). Frequent pattern mining: current status and future directions. *Data Mining and Knowledge Discovery*, 15(1):55–86.

Han, J., Dong, G., and Yin, Y. (1999). Efficient mining of partial periodic patterns in time series database. In *Data Engineering, 1999. Proceedings., 15th International Conference on*, pages 106–115. IEEE.

Han, J., Gong, W., and Yin, Y. (1998). Mining segment-wise periodic patterns in time-related databases. In *KDD*, pages 214–218.

Han, J., Pei, J., and Kamber, M. (2011). *Data mining: concepts and techniques*. Elsevier.

Havelock, D., Kuwano, S., and Vorländer, M. (2008). *Handbook of signal processing in acoustics*. Springer Science & Business Media.

Hermann, T. (2008). Taxonomy and definitions for sonification and auditory display. In *Proceedings of the 14th International Conference on Auditory Display (ICAD 2008)*.

Hertkorn, N., Ruecker, C., Meringer, M., Gugisch, R., Frommberger, M., Perdue, E., Witt, M., and Schmitt-Kopplin, P. (2007). High-precision frequency measurements: indispensable tools at the core of the molecular-level analysis of complex systems. *Analytical and bioanalytical chemistry*, 389(5):1311–1327.

Hinneburg, A. and Keim, D. A. (1999). Optimal grid-clustering: Towards breaking the curse of dimensionality in high-dimensional clustering.

Hubler, S. L. and Craciun, G. (2012). Periodic patterns in distributions of peptide masses. *BioSystems*, 109(2):179–185.

Hughes, I. and Hase, T. (2010). *Measurements and their uncertainties: a practical guide to modern error analysis*. Oxford University Press.

Hughey, C. A., Hendrickson, C. L., Rodgers, R. P., Marshall, A. G., and Qian, K. (2001). Kendrick mass defect spectrum: a compact visual analysis for ultrahigh-resolution broadband mass spectra. *Analytical Chemistry*, 73(19):4676–4681.

Jain, A. K., Murty, M. N., and Flynn, P. J. (1999). Data clustering: a review. *ACM computing surveys (CSUR)*, 31(3):264–323.

Kendrick, E. (1963). A mass scale based on ch2= 14.0000 for high resolution mass spectrometry of organic compounds. *Analytical Chemistry*, 35(13):2146–2154.

Keogh, E., Lonardi, S., and Ratanamahatana, C. A. (2004). Towards parameter-free data mining. In *Proceedings of the tenth ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 206–215. ACM.

Keogh, E. J. and Pazzani, M. J. (2001). Derivative dynamic time warping. In *Proceedings of the 2001 SIAM International Conference on Data Mining*, pages 1–11. SIAM.

Kim, S., Kramer, R. W., and Hatcher, P. G. (2003). Graphical method for analysis of ultrahigh-resolution broadband mass spectra of natural organic matter, the van krevelen diagram. *Analytical Chemistry*, 75(20):5336–5344.

Kogan, J. A. and Margoliash, D. (1998). Automated recognition of bird song elements from continuous recordings using dynamic time warping and hidden markov models: A comparative study. *The Journal of the Acoustical Society of America*, 103(4):2185–2196.

Kujawinski, E. B. and Behn, M. D. (2006). Automated analysis of electrospray ionization fourier transform ion cyclotron resonance mass spectra of natural organic matter. *Analytical chemistry*, 78(13):4363–4373.

Kunenkov, E. V., Kononikhin, A. S., Perminova, I. V., Hertkorn, N., Gaspar, A., Schmitt-Kopplin, P., Popov, I. A., Garmash, A. V., and Nikolaev, E. N. (2009). Total mass difference statistics algorithm: a new approach to identification of high-mass building blocks in electrospray ionization fourier transform ion cyclotron mass spectrometry data of natural organic matter. *Analytical chemistry*, 81(24):10106–10115.

Levine, M. A., Marrs, R., Henderson, J., Knapp, D., and Schneider, M. B. (1988). The electron beam ion trap: A new instrument for atomic physics measurements. *Physica Scripta*, 1988(T22):157.

Lin, J., Keogh, E., Lonardi, S., and Chiu, B. (2003). A symbolic representation of time series, with implications for streaming algorithms. In *Proceedings of the 8th ACM SIGMOD workshop on Research issues in data mining and knowledge discovery*, pages 2–11. ACM.

Lin, J., Keogh, E., Wei, L., and Lonardi, S. (2007). Experiencing sax: a novel symbolic representation of time series. *Data Mining and knowledge discovery*, 15(2):107–144.

Manning, C. D., Raghavan, P., Schütze, H., et al. (2008). *Introduction to information retrieval*, volume 1. Cambridge university press Cambridge.

Milczynski, M., Hermann, T., Bovermann, T., and Ritter, H. (2006). A malleable device with applications to sonification-based data exploration. In *Proceedings of the International Conference on Auditory Display*.

Nasir, T. and Roberts, J. C. (2007). Sonification of spatial data. Georgia Institute of Technology.

Paizs, B. and Suhai, S. (2005). Fragmentation pathways of protonated peptides. *Mass spectrometry reviews*, 24(4):508–548.

Parthasarathy, S., Mehta, S., and Srinivasan, S. (2006). Robust periodicity detection algorithms. In *Proceedings of the 15th ACM international conference on Information and knowledge management*, pages 874–875. ACM.

Pluim, J. P., Maintz, J. A., and Viergever, M. A. (1999). Mutual information matching and interpolation artifacts. In *Medical Imaging'99*, pages 56–65. International Society for Optics and Photonics.

Proschak, E., Wegner, J. K., Schüller, A., Schneider, G., and Fechner, U. (2007). Molecular query language (mql) a context-free grammar for substructure matching. *Journal of chemical information and modeling*, 47(2):295–301.

Reemtsma, T. (2009). Determination of molecular formulas of natural organic matter molecules by (ultra-) high-resolution mass spectrometry: status and needs. *Journal of chromatography A*, 1216(18):3687–3701.

Rényi, A. et al. (1961). On measures of entropy and information. In *Proceedings of the fourth Berkeley symposium on mathematical statistics and probability*, volume 1, pages 547–561.

Rosen, S. and Howell, P. (2011). *Signals and systems for speech and hearing*, volume 29. Brill.

Roullier-Gall, C., Boutegrabet, L., Gougeon, R. D., and Schmitt-Kopplin, P. (2014). A grape and wine chemodiversity comparison of different appellations in burgundy: Vintage vs terroir effects. *Food chemistry*, 152:100–107.

Sakoe, H. and Chiba, S. (1978). Dynamic programming algorithm optimization for spoken word recognition. *IEEE transactions on acoustics, speech, and signal processing*, 26(1):43–49.

Salvador, S. and Chan, P. (2007). Toward accurate dynamic time warping in linear time and space. *Intelligent Data Analysis*, 11(5):561–580.

Schlunegger, U. P. (2016). *Advanced mass spectrometry: applications in organic and analytical chemistry*. Elsevier.

Scholkopf, B. (2001). The kernel trick for distances. *Advances in neural information processing systems*, pages 301–307.

Singleton, R. C. (1967). On computing the fast fourier transform. *Communications of the ACM*, 10(10):647–654.

Sleno, L. (2012). The use of mass defect in modern mass spectrometry. *Journal of mass spectrometry*, 47(2):226–236.

Smith, S. W. et al. (1997). The scientist and engineer's guide to digital signal processing.

Srebro, N. (2007). How good is a kernel when used as a similarity measure? In *International Conference on Computational Learning Theory*, pages 323–335. Springer.

Tolmachev, A. V., Robinson, E. W., Wu, S., Kang, H., Lourette, N. M., Paša-Tolić, L., and Smith, R. D. (2008). Trapped-ion cell with improved dc potential harmonicity for ft-icr ms. *Journal of the American Society for Mass Spectrometry*, 19(4):586–597.

Turetsky, R. J. and Ellis, D. P. (2003). Ground-truth transcriptions of real music from force-aligned midi syntheses.

Van Leeuwen, M. (2014). Interactive data exploration using pattern mining. In *Interactive Knowledge Discovery and Data Mining in Biomedical Informatics*, pages 169–182. Springer.

Vlachos, M., Yu, P., and Castelli, V. (2005). On periodicity detection and structural periodic similarity. In *Proceedings of the 2005 SIAM International Conference on Data Mining*, pages 449–460. SIAM.

Wallace, W. E. and Guttman, C. M. (2002). Data analysis methods for synthetic polymer mass spectrometry: Autocorrelation. *Journal of research of the National Institute of Standards and Technology*, 107(1):1.

Wang, A. (2006). The shazam music recognition service. *Communications of the ACM*, 49(8):44–48.

Wang, A. et al. (2003). An industrial strength audio search algorithm. In *ISMIR*, volume 2003, pages 7–13. Washington, DC.

Yu, L., Xiong, Y.-M., and Polfer, N. C. (2011). Periodicity of monoisotopic mass isomers and isobars in proteomics. *Analytical chemistry*, 83(20):8019–8023.

Zou, Q., Chu, W., Johnson, D., and Chiu, H. (2001). A pattern decomposition (pd) algorithm for finding all frequent patterns in large datasets. In *Data Mining, 2001. ICDM 2001, Proceedings IEEE International Conference on*, pages 673–674. IEEE.