TECHNISCHE UNIVERSITÄT MÜNCHEN

Lehrstuhl Analytische Lebensmittelchemie

Machine Learning and Network Analysis using Mathematical Optimisation in Mass Spectrometry Bioinformatics

Dimitrios Tziotis

Vollständiger Ausdruck der von der Fakultät für Wissenschaftszentrum Weihenstephan für Ernährung, Landnutzung und Umwelt der Technischen Universität München zur Erlangung des akademischen Grades eines

Doktors der Naturwissenschaften

genehmigten Dissertation.

Vorsitzender: Univ.-Prof. Dr. E. Grill

Prüfer der Dissertation:

1. apl. Prof. Dr. P. Schmitt-Kopplin
2. Univ.-Prof. Dr. M. Rychlik

Die Dissertation wurde am 17.07.2013 bei der Technischen Universität München eingereicht und durch die Fakultät für Wissenschaftszentrum Weihenstephan für Ernährung, Landnutzung und Umwelt am 10.02.2014 angenommen.

Table of Contents

CHAPTER I

Thesis overview

CHAPTER II

Prologue

CHAPTER III

Structural mass difference networks and the Netcalc method

CHAPTER IV

Unsupervised learning and cluster analysis on mass spectrometric data

CHAPTER V

A combinatorial learning framework for sample classification and discriminant signal identification in complex datasets

CHAPTER VI

An adapted combinatorial learning model for the study of discriminant chemical reactions in mass difference networks

CHAPTER VII

Epilogue

CHAPTER I

Thesis overview

In this initial chapter I provide a general overview of my doctoral dissertation as it is presented throughout this manuscript. I talk about the motives behind my research interests and describe how they develop into concise scientific objectives over the course of my work.

1.1 Background and motivation

Today's progress in Natural Science comes with an exponential increase in the production of experimental data. Consequently, there is a growing need to adapt and develop new computational and quantitative techniques to deal with this data in a meaningful manner. Machine Learning is a modern sub-field of Computer Science which combines elements from applicable mathematics, computational statistics, and decision-making. Such a quantitative discipline with powerful computational techniques and the ability to deal with large amounts of digital information is an ideal tool for the “Big Data” era that Life Sciences have entered in the recent years. Biogeochemical studies and metabolomics are no exception to the Big Data phenomenon, and along with continuous progress in mass spectrometry instrumentation comes a steady increase in the production of complex data of high computational demands. High-field Fourier transform ion cyclotron mass spectrometry (ICR-FT-MS) is a modern analytical technique with applications on important areas of biogeochemical research, such as natural organic matter (NOM) and

metabolomics. One critical bottleneck in ICR-FT-MS analyses concerns the quantitative processing and meaningful visual display of extremely large datasets. In the case of natural organic matter, ICR-FT-MS is the key technique to deduce molecular formulae from different terrestrial environments such as soils, sediments, fresh and marine waters, atmospheric aerosols as well as extraterrestrial NOM. Ultra-high resolution and excellent mass accuracy are two characteristics of ICR-FT-MS that materialise the distinction of more than tens of thousands of ions and several thousands of assigned molecular compositions directly out of complex mixtures. A precise molecular description of NOM, based on their carbon, hydrogen, oxygen and heteroatom (e.g. nitrogen, sulphur and phosphorus) -bearing formulae, facilitates the understanding of environmental biogeochemical processes. The elementary formula annotation of those exact masses is one of the most challenging tasks at hand, albeit efficient computational means are yet to be discovered. In addition, information-rich, structure-dependent visualisation schemes are indispensable for any significant mass-spectrometric analysis of NOM and other complex organic mixtures. In the case of metabolomics, ICR-FT-MS produces datasets comparable in size and complexity to those of DNA microarrays. The bottleneck for quantitative analysis of such vast datasets lies on the efficient classification of samples into regions of varying biological significance and the identification of masses discriminant to different metabolic states. The scopes of the two biochemical orientations (bio- and geo-) are inevitably overlapping. Efficient elementary formula calculation algorithms can be used for annotating metabolite masses, while biological classification and machine learning techniques can be applied on NOM samples. Mass spectrometry has only recently entered the world of -omics and, consequently, bioinformatics research on the topic is still at an early stage, albeit the lack of specialised computational and quantitative techniques is particularly evident in the case of ICR-FT-MS data analytics.

My academic background is in Operational Research and Machine Learning; therefore, it has been my objective in this thesis to combine elements from these two disciplines in order to produce quantitative methods and tools adapted for mass spectrometry data mining. In this work, I propose a novel computational methodology to address the requirements of ICR-FT-MS data analysis more effectively. My methodology involves

mathematical models and algorithms adapted for the quantitative needs of two distinct yet related scopes of biogeochemical research: natural organic matter and metabolomics. I treat the two scopes individually, first by developing the theoretical frameworks and producing experimental results for each of them separately, then by deriving a unified framework through the merge of scopes and models together. The first scenario involves a graph-theoretical treatment of ICR-FT-MS data for the elementary formula calculation of exact masses by implementing a biochemical network reconstruction that offers significant advantages over conventional probabilistic annotation approaches. The second scenario introduces a quantitative framework involving complex combinatorial optimisation problem-solving for the purposes of supervised classification, clustering, and other machine learning techniques contributing to discriminant signal identification.

The objectives of this thesis are the following:

(i) Using the ICR-FT-MS *exact mass* information of a single sampled m/z spectrum: To develop and standardise an efficient inference algorithm for the purpose of elementary formula calculation on an enhanced graph-theoretical model of biochemical network reconstruction. Depending on the nature of the dataset and the parametrisation of the search, results of varying efficiency can be attained. The results of our method are always expected to be superior to those of conventional approaches. We divide our reconstructed structural mass difference graphs into *compositional* and *functional* networks and refer to the elementary formula calculation algorithm as *Netcalc* (chapter III).

(ii) Using the ICR-FT-MS *intensity* information of a set of sampled spectra: To develop a theoretical framework which aims to unify and address the principal questions of non-targeted Metabolomics data analysis. The framework is based on the modelling of a biological scenario into an *Operational Research* problem, which can be treated using discrete mathematical optimisation and solved via metaheuristic search. As the approach mixes combinatorial optimisation with machine learning, I refer to it as *combinatorial learning*. The first application of the framework applies combinatorial learning on metabolite masses and I refer to it as *metabolic optimisation* (chapters IV and V).

(iii) Combining the scopes of (i) and (ii): To propose a novel biological parameter which may additionally characterise a ICR-FT-MS dataset, based on the graph-theoretical model of (i). A new adapted optimisation model is created by merging the graph-

theoretical scope of (i) with the optimisation framework of (ii). This application of the framework of (ii) applies combinatorial learning on the structural mass difference networks of (i) and I refer to it as *mass difference optimisation* (chapter VI).

(iv) To develop the software tools (source code) required to materialise, test, and produce results from the described theoretical models.

1.2 Structural mass difference network and the Netcalc method

In a mass difference network, exact masses are represented by nodes and chemical transformations by the edges existing between those nodes. Breitling et al.[1] introduces and successfully applies the concept of mass difference network reconstruction on unbiased mass spectrometric data. In this thesis we extend the approach by using exact masses of higher precision provided by our 12 Tesla ICR-FT-MS instrumentation. We define the rules for *compositional* and *functional* network reconstruction as well as an enhanced graph visualisation scheme equivalent to Van Krevelen diagrams. I refer to those networks collectively as *structural mass difference networks* or just *structural networks*. In addition, we introduce the *Netcalc* algorithm for the efficient calculation of chemical formulae via network inference. Our results were superior to those of conventional approaches and our findings were published in Tziotis et al. [2].

1.3 Combinatorial learning framework and the Metabolic optimisation model

The bottleneck in the quantitative analysis of such vast datasets lies on the identification of masses discriminant to different metabolic states combined with the efficient classification of samples into regions of varying risk. The conventional “black box” approaches used for these tasks have been criticised for a potential introduction of bias through multiple statistical assumptions and transformations, something which inevitably calls their efficiency into question [3]. To date, very few computational methods have been developed for, or adapted to, ICR-FT-MS metabolomics in order to explore the vast potential of this modern analytical technique. The alleged limitations of the current “standard” techniques and the sparsity of in-depth quantitative research on the field of

Fourier transform mass spectrometry metabolomics have inspired us to propose a combinatorial machine learning approach to the problem of discriminant signal identification and sample classification. The key aspect of this approach is the intuitive and flexible modelling which aims to minimise the statistical bias and biological inconsistency of conventional “black box” approaches. Inspired by Operational Research, I propose a combinatorial optimisation framework that uses metaheuristic search algorithms in order to improve the clustering output of a graph-theoretical model that I call *co-intensity network*. I refer to the application of this framework on exact masses as *metabolic optimisation.* We tested the method on Crohn's disease dataset [4] and received biologically pertinent results in the areas of semi-supervised classification, diagnosis, and prediction. Due to the robustness and flexibility of the approach, I believe that it has the potential of becoming a standard method for quantitative analysis in ICR-FT-MS metabolomics as well as other fields of bioinformatics.

1.4 Mass difference optimisation model

Structural mass difference networks bring new insight into ICR-FT-MS data mining. The mass difference information of ICR-FT-MS spectra is precise to the point that it should be viewed as a significant biological parameter en par to metabolic biomarkers. The chemical transformations yielded by Netcalc can be, therefore, the subject of further analysis in order to ultimately isolate the ones which are more biologically pertinent for varying metabolic states. Just as metabolites of importance can be regarded as biomarkers which characterise a sample, mass differences of importance can be similarly detected and labeled “significant” in respect to the spectrum and biological context in question. Mass difference optimisation combines the theoretical framework of metabolic optimisation with structural mass difference networks in order to apply combinatorial learning on mass difference information and detect the chemical transformations that improve the biological clustering output of the corresponding sample's co-intensity network. The structure of this work is illustrated in figure 1.1.

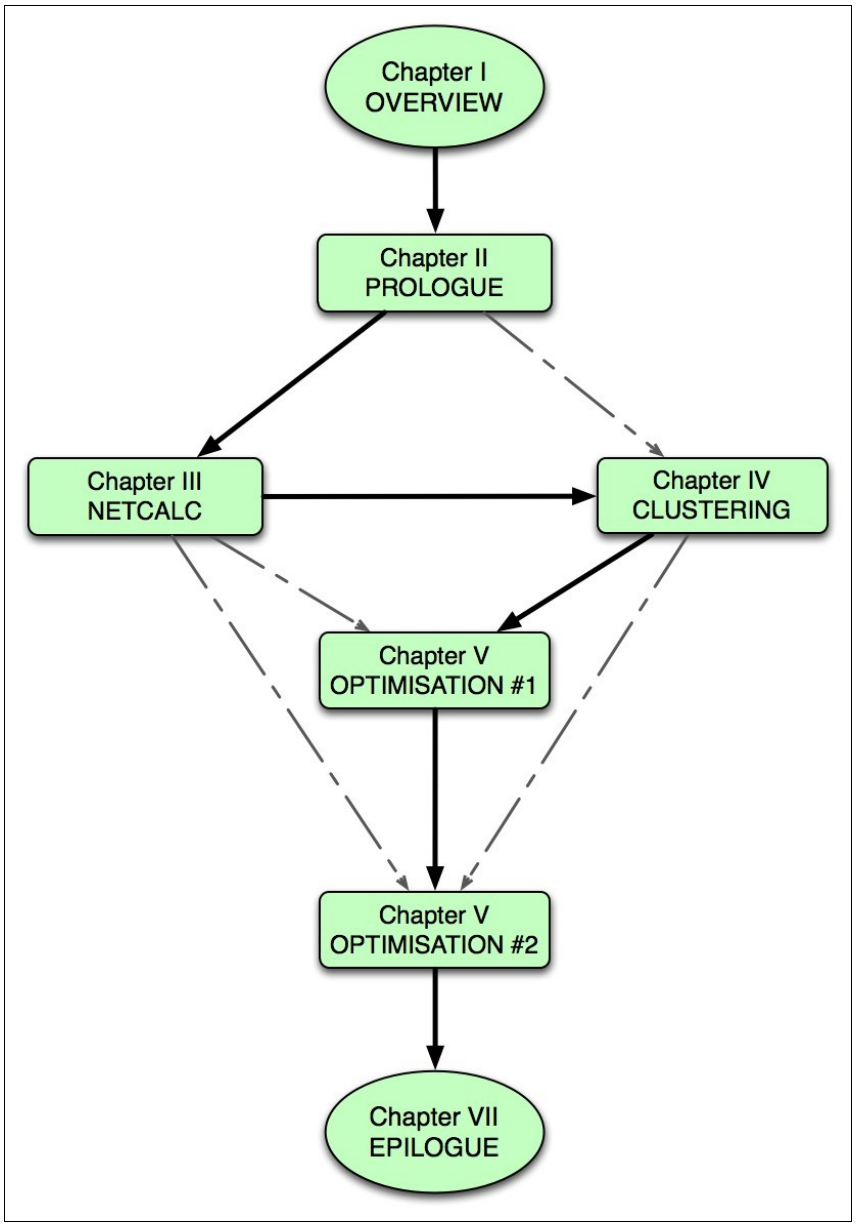

Figure 1.1: Dissertation chapter diagram.

CHAPTER II

Prologue

This chapter serves as an introduction to the multidisciplinary nature of my research interests and objectives. I describe the wider field of *Data Analytics* by tracing its roots to the quantitative disciplines of *Machine Learning*, *Statistics*, and *Operational Research.* I write about the quantitative bottlenecks that are met in Mass Spectrometry Bioinformatics today, as well as the state-of-the-art techniques that are seen as the golden standard to treat them. My main research objective is the combination of elements from Machine Learning and Operational Research in order to produce quantitative techniques specifically adapted to the needs of metabolomics and mass spectrometry data mining in general.

2.1 Computer Science and Bioinformatics

2.1.1 A description of Computer Science

Computer Science is, to date, one of the fastest-growing sectors of scientific and industrial research, serving as the substratum of technological advancement and economic activity worldwide. Contrary to popular belief, this vast field of study is not restricted to the development of software and hardware systems. As a matter fact, applications of computer science, such as computational data analysis, are nowadays deeply rooted in almost every branch of engineering and science, such as physics,

chemistry, biology, medicine, economics, and statistics (in recent years, the merge of computer science and statistics for the purpose of “Big Data Analytics” with industrial applications has been called “*Data Science*”). While the strictly applied scope of computer science is directly related to the development of computer systems, it should be noted that its theoretical basis constitutes a whole distinct branch of mathematical research (figure 2.1). The need to import quantitative techniques from computer science into other engineering and scientific disciplines emerged, perhaps, within Artificial Intelligence research and the ambition of man to create intelligent machines [5][6]. It was then observed that the algorithms and computational methods that had been developed for the purposes of *machine learning* would be easily adaptable and applicable to generic problems of data analysis in other disciplines. A more recent example of such an adaptation would be *pattern recognition*, where the ability of a robot to perceive visual patterns in artificial intelligence was adapted to DNA sequence alignment and protein structure prediction in bioinformatics. The quantitative and computational techniques used and developed within the scope of computer science (collectively known as *data mining*) are today among the most advanced data-analytical tools in existence.

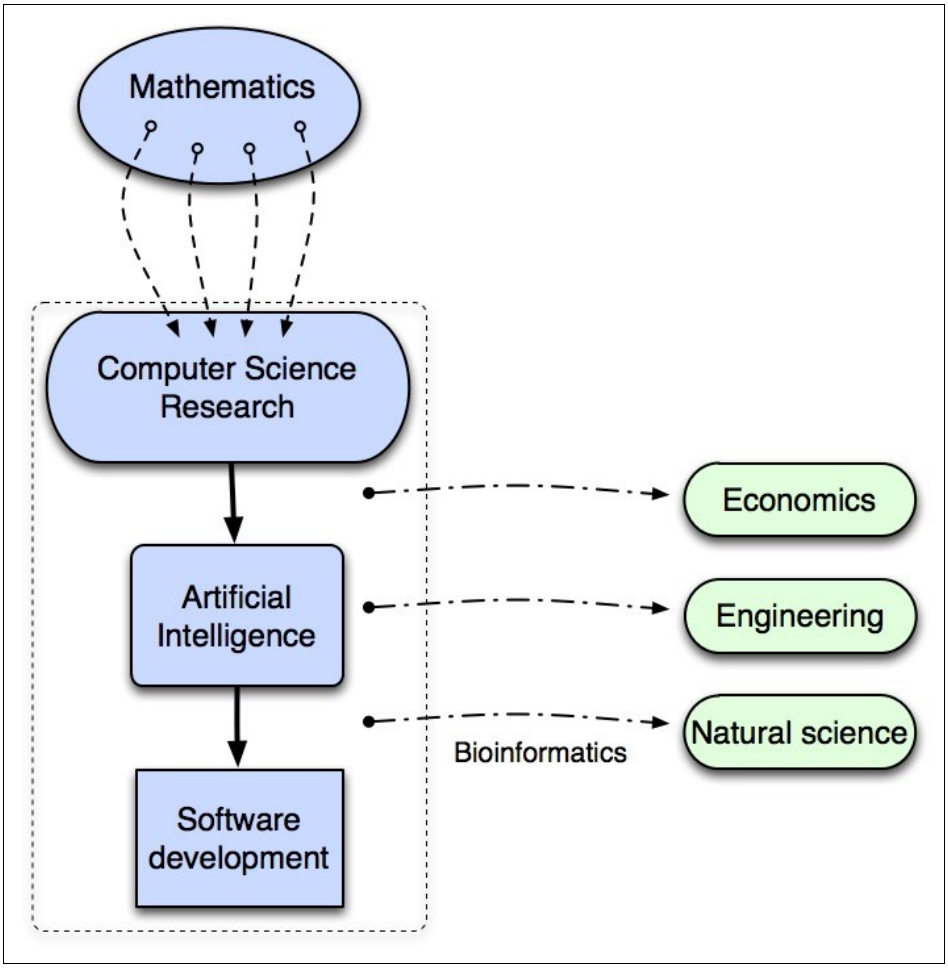

Figure 2.1: Computer science schema.

The applications of computer science outside its scope.

2.1.2 A description of bioinformatics

The complexity of biological data offers a very challenging medium for quantitative research. The adaptation of data mining methods and tools to the needs of biological research is commonly known as *bioinformatics* and, in some cases, *computational biology*. While bioinformatics is a very loosely defined term, it is generally associated with every applied or theoretical aspect borrowed from computer science and adapted to the needs of biology and natural science in general. As a result, bioinformaticians are people coming from various backgrounds (figure 2.2) who do not necessarily share the same scientific expertise and research focus. The applied/engineering side of bioinformatics involves technical aspects from computer engineering, ranging from the installation of databases to the development of specialised software services, while, the theoretical/quantitative side involves techniques from artificial intelligence (mostly machine learning), statistics, and applied mathematics, adapted to requirements of modern biological problems.

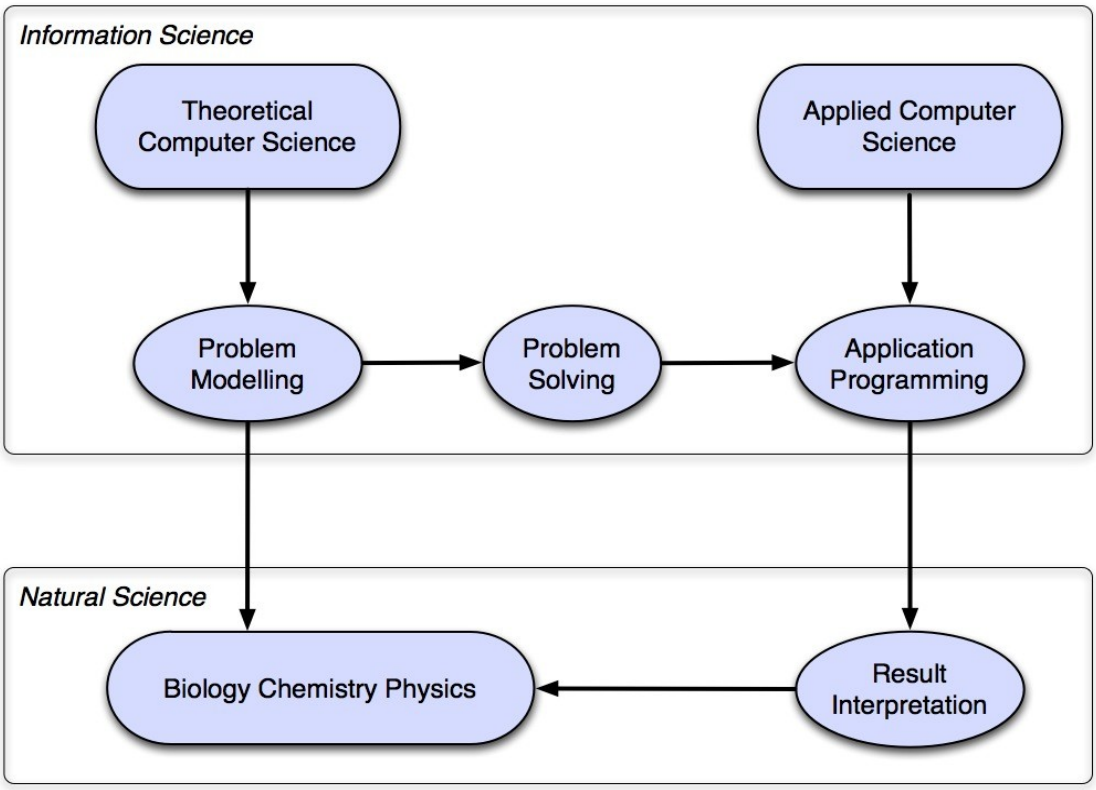

Figure 2.2 : Bioinformatics schema.

The link of computer science to natural science.

2.1.3 Mass spectrometry bioinformatics

Bioinformatics is fundamental for the elaboration of complex mass spectrometric data [7], whilst it has been traditionally associated to the identification and characterisation of proteins. In this thesis, I apply the term *mass spectrometry bioinformatics* to comprise all computational and quantitative methods developed for (or adapted to) any type of mass spectrometry data analysis scenario. I focus on the development of quantitative models and techniques adapted to datasets produced by *Ion Cyclotron Resonance Fourier Transform Mass Spectrometry* (ICR-FT-MS). I make a distinction between *mass-based and intensity-based* techniques, in accordance to the two main output parameters of ICR-FT mass spectrometry. In a later section, I write about the state of the art on the field of *mass spectrometry data analysis* and I describe some of the concepts used in my research, through which I aim to create a quantitative and computational framework for mass spectrometry and the evolution of what I like to call *mass spectrometry bioinformatics*.

2.2 Data production in mass spectrometry bioinformatics

In this section, I describe terms and concepts of analytical chemistry that are associated to my computational and quantitative method development. More specifically, I talk about the mass analysis technique known as *Fourier transform ion cyclotron resonance mass spectrometry* and its applications on *natural organic matter* and *metabolomics*, which results to the production of highly complex datasets.

2.2.1 Fourier transform ion cyclotron resonance mass spectrometry

Fourier transform ion cyclotron resonance mass spectrometry (ICR-FT-MS) is an ultra-high resolution technique, which can be used to determine masses with very high accuracy. This kind of mass spectrometry determines the mass-to-charge ratio (m/z) of ions by measuring their cyclotron frequency in a fixed magnetic field [8]. The continuous

development of mass detectors has improved pumping technologies, while stronger magnetic fields enable ultra-high mass resolution and enhanced mass accuracy in ICR-FT-MS. Mass resolution is very important when dealing with complex samples as it enables the differentiation of closely located signals (m/z - mass per charge) [9]. All datasets used in this thesis have been produced with an ICR-FT-MS (solariXTM, Bruker - Bremen, Germany) equipped with a 12 Tesla superconducting magnet in direct injection experiments. Details about the advantages of ultra-high mass accuracy in mass spectromery bioinformatics are provided in a different chapter.

2.2.2 Natural organic matter

Natural organic matter (NOM), the most abundant fraction of organic carbon in the bio- and geo- sphere, ranges among the most complex mixtures of organic molecules on earth [10]. In contrast to biopolymers with known fundamental building blocks, NOM are non-repetitive complex systems [2]. The formation of organic matter (OM) in space and on earth preceded has terrestrial life. Through the ages, coevolution between prebiotic/abiotic molecules, OM, and primitive and higher forms of life resulted in evolutionary, pre-validated biomolecules eventually deriving from a genetic code, and complex biogeochemical, non-repetitive “natural” organic matter (NOM) being generated, within the general constraints of thermodynamics and kinetics, from molecules of geochemical or, ultimately, biogenic origin. Apart from the man-made, non-natural exploration of the chemical space, through e.g. diversity oriented organic synthesis and combinatorial chemistry, the antipodes of natural organic complexity are represented by biomolecules resulting from abiotic chemical evolution, with extraterrestrial organic matter found in carbonaceous chondrites as a credible end-member. Terrestrial NOM, formed by the combined action of biotic and abiotic reactions as a characteristic of the respective ecosystems, ranges in between. However, the intricacy of terrestrial NOM molecular signatures already approaches the limits defined by the laws of chemical binding [10].

In recent years, high-field Fourier transform ion cyclotron mass spectrometry most convincingly demonstrated the enormous molecular intricacy of NOM: ultrahigh

resolution and excellent mass accuracy characteristic of high field ICR-FT-MS mass spectra enabled the distinction of more than tens of thousands of ions and several thousands of assigned molecular compositions directly out of non-fractionated NOM [11] [12][13].

2.2.4 Metabolomics

Metabolites are compounds of low molecular weights and varying concentrations that participate in metabolic reactions. These compounds are attributed with diverse physicochemical characteristics, ranging from ionic to hydrophobic properties [14]. The cellular pool of all metabolites, known as the “metabolome”, is the product of a permanent chemical transformation in which metabolites are precursors, intermediates, or end products [15][16]. The exact number of metabolites in mammalians, plants and microbes is still unknown, however, estimates vary from several hundred thousand up to one million [17][18]. Groups of metabolites are assembled in pathways in which the product of one reaction serves as a substrate for the next reaction while interconnected pathways are forming highly linked metabolic networks [19]. Our current knowledge of the metabolome covers approximately only 10% of estimated existing metabolites and it is assumed that currently “unknown” compounds might truncate and modulate known pathways as well as point to the existence of new metabolic pathways [1]. Reasonably, the process of metabolic adaptation to an external stimulus (e.g. an infection) can have an impact on the regulation of unknown metabolites. Similarly, intermediates from a novel pathway may influence regulation, activation or inhibition of a known pathway (figure 2.3) and the discovery of such intermediates in an infection or disease would offer new therapeutic prospects.

The functional phenotype of a system is characterised by its metabolites. Changes in metabolite patterns reflect environmental and genetic perturbations [9], thus, the analysis of metabolites is an important step towards the deeper understanding of cellular regulation and adaptation processes. A metabolomics investigation can be regarded as *targeted* or *non-targeted* depending on whether it aspires to verify a pre-established

theory or generate a novel hypothesis. Currently, metabolomics is able to monitor changes in metabolic activity in response to genetic and nutrient perturbations, in addition to decode unknown gene functions [20][21][22] as well as investigate processes in disease, infection and treatment [23]. Moreover, metabolomics can contribute to diagnosis and therapeutics by offering: (a) discovery of (pre-) disease markers inferred by metabolite shifting, (b) easier differentiation between cancerous and healthy tissue, (c) monitoring of drug responses, (d) discovery of new lead structures for novel therapeutic agents, (e) identification of microorganisms via fingerprinting techniques. The integration of metabolomics data into the “omics” disciplines (proteomics, transcriptomics, genomics), along with the combination of all available molecular-biological and phenotypic knowledge, can provide a more comprehensive understanding of biological processes [9][24].

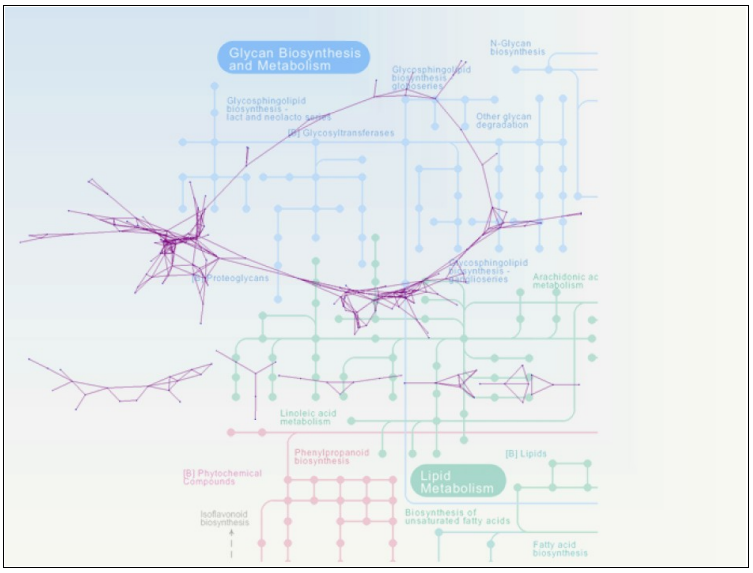

Figure 2.3: A reconstructed metabolic network over a network of KEGG metabolic pathways.

2.4 ICR-FT-MS data analysis: classical intensity-based methods

In this section I present the state of the art in classical “intensity-based” approaches used in ICR-FT-MS data analysis. Much of these approaches have been borrowed from other disciplines and applied on a mass spectrometric context where the intensity information of a m/z spectrum is the most vital parameter in the analysis. This may seem somewhat paradoxical if we consider that the power of this instrument lies on ultra-high mass accuracy rather than sensitivity; nonetheless, this approach becomes inevitable when we are dealing with a multi-dimensional space of variables produced by several measurements over a number of samples. A m/z spectrum of a single sample comes in the form of two lists, one being the exact masses and the other their corresponding intensities, yet in the case of multiple samples (such measurements over a group of individuals) the exact mass information is only used in data preparation by comparing mass values between different samples and creating a *mass-sample intensity matrix*. The information in this matrix is that of intensity values corresponding to masses on rows and samples on columns (as explained in the next section). It has been one of my goals, in this thesis, to integrate the exact mass information in ICR-FT-MS data mining in a more meaningful way (Chapter I, objective (i)).

2.4.1 Multivariate analysis in metabolomics

Non-targeted experiments generate large amounts of data, which can be handled by computational and statistical techniques in order to discover reccurring patterns. The experimental setup involves the mass spectrometric measurement of m samples yielding an equal number of m/z spectra, one for each sample. The exact mass values of all spectra are compared in order to detect a number of n metabolites that were found to be present in all m samples. The intensity information corresponding to those n common metabolites is represented in the form of a $n \times m$ matrix, where the intensity value of mass in row i corresponds to the identity of sampled subject in column j. In conventional approaches, mass information is ignored once this preparatory process is achieved. Standard analysis

procedure involves data pre-treatment, such as statistical transformations and normalisation, the latter being important when the intensities of detected features vary in several orders of magnitudes. The key objectives of the analysis are simple: classification of sampled subjects into biologically pertinent groups and identification of individual metabolites with biological significance in respect to those groups or the dataset as a whole. There are two approaches to the problem of classification via supervised or unsupervised algorithms, depending on whether or not we are willing to consider the experimental knowledge of the biological grouping that we expect to observe on the data. In the unsupervised case, where this prior knowledge is chosen to be ignored, the purpose of the process is to examine how easily the data can form patterns of biological significance. Whether or not these patterns can be formed successfully is an indicator of the data's complexity and biological clusterability. In the supervised scenario, a part of the data is used in order to train a predictor model which is meant to be consequently applied to classify the rest of the data (cross-validation). In theory, this predictor can be also used to classify unknown datasets of the same nature, albeit with questionable efficiency. In both cases, however, we are able to pin down the masses which had the biggest impact on each of the different biological groups. Those metabolites of interest are also referred to as biomarkers. In the rest of this section I describe the two standard methods of *multivariate analysis* used to deal with the above-mentioned problematics in metabolomics: *Principal Component Analysis* and *Partial Least Squares regression*.

2.4.2 Principal Component Analysis (PCA)

Principal Component Analysis (PCA) is currently one of the standard algorithms used for unsupervised feature extraction in metabolomics. The PCA transformation is a simple, non-parametric method which reduces the dimensionality of the data while preserving relevant information by converting highly correlated variables into linearly uncorrelated ones. PCA's dimensionality reduction has deemed it a valuable tool for data compression, with many applications in fields such as image recognition and computer graphics. On the other hand, PCA has limited classification as well as visualisation capabilities, and has been criticised for not being an optimal method for feature extraction [25].

Nonetheless, the approach has gained popularity due to its simplicity and ease of use and has led many data analysts to overlook its downsides [25][26]. In addition, PCA runs under the assumptions that (a) the dimensionality of data can be efficiently reduced by linear transformation and (b) critical information is stored in the directions where input data variance is maximum; conditions which are not always met [27].

2.4.3 Partial Least Squares regression (PLS)

The complexity of metabolomics data usually calls for a supervised technique. The method of preference is known as *partial least squares regression* (PLS), a combination of PCA and *multiple linear regression*. PLS is a multivariate projection-based method used with data that contain correlated predictor variables. The algorithm constructs new predictor variables as linear combinations of the original predictor variables while considering the observed response values, leading to a model with predictive capabilities [28] that can be used for classification/discrimination problems, even though it was not originally designed for this purpose [29]. Unlike PCA, where the variance of a single dataset is maximised, PLS maximises the covariance between two datasets by searching for linear combinations of their variables. These linear combinations are called *latent variables,* while the weight vectors used to compute them are called *loadings* [29]. The variant method known as *partial least squares-discriminative analysis* (PLS-DA) is applied when the response variable is categorical. The main advantage of PLS is that it involves no assumption about the data distribution or scale of measurement, however, it possesses several drawbacks that deem it unsuitable for some scenarios [3]. The limitations of both PCA and PLS along with my motivation for coming up with alternative methods are discussed in a different section.

2.5 ICR-FT-MS data mining: Graph theory and mass-based methods

In this section I describe some of the modern “mass-based” quantitative methods that have been applied on ICR-FT-MS data mining. These are all relatively new techniques,

which are largely based on the discipline of *graph theory*. I provide a brief introduction to graph theory and its generic applications in bioinformatics before I move onto their adaptation on mass spectrometry. The described *network analysis* techniques are, to date, mostly focusing on the *exact mass* information of an m/z spectrum. One of the goals in this thesis was to enhance these graph-based approaches as well as to integrate mass and intensity information in one single quantitative framework (Chapter II, objectives (ii), (iii)).

2.5.1 Graph theory and network analysis in Bioinformatics

Graph theory is widely used in bioinformatics due to its ability to provide efficient means of modelling, visualising, and solving real-world scenarios. Many pragmatic situations can be represented in the form of a diagram consisting of a set of points (nodes) and a set of lines (edges) connecting parts of these points; a mathematical abstraction which yields the concept of a graph [30] (figure 2.4). Such an abstraction can be represented graphically, and through this graphical representation we are able to study some its properties. A graph (also called a network) is in addition associated with a specialised matrix which allows us to store it inside computers and apply mathematical methods to analyse our data more thoroughly; a procedure known as *network analysis* [77].

Network analysis is a sub-field of graph theory which offers a quantifiable description of the networks that characterise various real-world systems. A number of network properties allows us to compare and characterise different types of complex networks. The most elementary property of a node is its degree (or connectivity), *k*, which tells us how many links a node has to other nodes. Based on this property, the degree distribution *P(k)* gives the probability that a selected node has exactly *k* links. The degree distribution allows us to distinguish between different topologies of networks [31]. The majority of biological networks belong to the scale-free topology, which means that their degree distribution approximates a power law [31]. Scale-free networks are highly non-uniform and most of the nodes have very few links. Only a small number of nodes have a very large number of links, these nodes are known as 'hubs' as they hold the network together. The scale-free topology is linked to the growth of the network in which new nodes are

preferentially attached to highly-connected already established nodes [32]. Such an inhomogeneous and modular system displays tolerance to random errors, as well as fragility against the removal of its most connected nodes [33]. As it can be seen in the plots, the degree distribution P(k) of our networks approximates a power law in all samples, a property claimed by complex systems of all types, notably real, large systems with many autonomous and interacting components. Complex phenomena are distributed in a highly skewed manner, rather than following the normal, Gaussian pattern. It has been previously observed in metabolic and biochemical networks that few highly-connected molecules play a central role in mediating interactions among numerous, less connected molecules [34][35].

It is through chemical and physical interactions that molecules influence one another and carry out biological functions. However, some of these influences may have a greater impact than others, therefore a hierarchical characterization of their relative importance can be very useful in exploring their functional architecture [36]. Network analysis plays an important role in the exploration of complex interactional systems. The potential of network analysis lies on the usage of our network's properties in order to (a) provide a hierarchical characterization of masses [37][33], (b) detect meaningful clusters of masses which display biological pertinence (already observed in metabolic networks [37]).

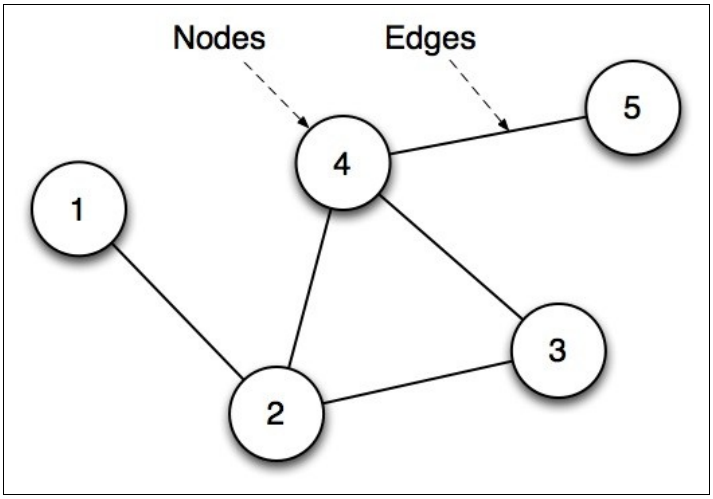

Figure 2.4: A sample graph with five nodes and five edges.

2.5.2 Structural mass difference network reconstruction

Network analysis can be applied on almost every scenario of FTMS spectra in a number of ways. An important approach involves the mass difference networks, in which each node represents an exact experimental mass and each edge represents a selected mass difference either taken from a predefined list of potential transformations, or detected on the fly through mass difference clustering and correlation analysis [1]. Such a network model can be divided into *compositional* and *functional* networks [2]. In the case of structural networks, a list of selected theoretical mass differences is used in order to determine the adjacency relation between the nodes, i.e. detected transformations between the experimental masses. The resulting network can be described as a reconstruction of the real biochemical system which can reflect the structural information expressed in an ICR-FT/MS dataset. Through graph inference we are able to perform an efficient, network-based formula calculation technique known as Netcalc [77].

2.6 Machine learning and prediction

The subfield of *Artificial Intelligence* (in turn a subfield of Computer Science) which studies computational methods for computer reasoning is known as *machine learning*. It can be said that machine learning is a discipline dealing with making predictions from data. Another discipline which also deals with prediction-making is *regression analysis*, a subfield of Statistics. Naturally, machine learning and computational statistics are closely related and often overlap, the main difference being that they were developed within different research scopes. The applications of statistics have been traditionally linked to social science, e.g. predicting how will a certain financial indicator react under certain conditions. Machine learning, on the other hand, is closely related to the Artificial Intelligence hype of the 1950s and the ambition of building conscious and intelligent machines, therefore, prediction-making in this case has to do with how a computer perceives its surroundings, e.g. by means of face or voice recognition.

A *prediction* can be described as the act detecting patterns in an "unknown" dataset, based on your knowledge of observed patterns in a "known" dataset. The assumption is that both datasets should behave similarly. In a temporal context, where all data is associated to sequential points in time, you observe patterns that happened in the past in order in order to use that knowledge and infer what may happen to something similar in the future. This context is known as *time-series forecasting*. What I just described as "past" and "future" is basically expressed in machine learning terms as "known data" (training set) and "unknown data" (prediction set). A "prediction" comes in two forms: *regression* and *classification* – depending on whether the output of the predictor is in continuous or discrete form. The research objectives of this thesis focus more on classification and less on regression, although some classifiers work by discretising the continuous output of a regression analysis.

Classification is a key concept from machine learning which extends to the generic needs of data mining and statistical inference. We mentioned classification in a previous section and described its function in the context of multivariate analysis. In machine learning terms, the two most common subdivisions of classification are supervised and unsupervised learning. There are numerous classification algorithms with wide usage in bioinformatics but there is no clear emerging consensus regarding their performance [38]. Some of those techniques, such as *k-means clustering* and *artificial neural networks,* were developed within machine learning while others, such as *Principal Component Analysis* and *Partial Least Squares regression*, stem from the field of *multivariate statistics*. In this section I describe some key methods from the former category, i.e. the algorithms from Artificial Intelligence. In this thesis I developed various supervised, unsupervised, and semi-supervised learning models, using discrete mathematical modelling.

Artificial Neural Networks (ANN) is a powerful machine learning family of algorithms which was used in this work for both supervised and unsupervised classification. Extensive background work was performed on Artificial Neural Networks in the context of exploring the potential of existing prediction techniques and how they would perform in a discrete optimisation framework. The results produced with the Artificial Neural

Networks known as *Self-Organizing Maps* are presented in chapter IV. It should be noted that I performed considerably more work on Artificial Neural Networks than what is included in this manuscript. I developed custom supervised models, algorithms, and source code, however, I chose not to include such work in this document in order to put the focus on my own methods.

2.6.1 Supervised learning

In a typical unsupervised classification scenario, a dataset of correctly-identified observations is divided between a *training set* and a *test set*, usually at the rates of 30% and 70% respectively. A learning algorithm known as a *classifier* is applied on the training set in order to build a mathematical model that is called a *predictor function* and can be subsequently used to classify the observations of the test set. If the predictor performs on the test set with a satisfactory success rate, it can be then assumed that the model may be applied on a new similar dataset whose true classification is unknown. In other words, the training data is used for model fitting and the test data in model validation; a technique widely known as *cross-validation*. The process of model-fitting, where the target values of the outputs in the fitted data are known, is called *supervised learning* [38]. The rest of this section provides a brief description of the supervised classification algorithms that were used in the course of this work. I developed my own source code for every machine learning method presented in this section (as for most algorithms seen in this work).

Artificial Neural Networks - Linear perceptron :

The linear perceptron is a single-layer *Artificial Neural Network* (ANN). It is a supervised linear classifier which uses a linear predictor function to combine a set of weights with a feature vector and map a single input *x* to an output class *y*. It belongs to the family of algorithms known as *artificial neural networks*; a machine learning model inspired by biological neural networks. As a feed-forward neural network, the linear

perceptron passes its inputs to a single layer of output nodes via a series of weights. This algorithm and its variants form the basis of more complex learning algorithms with wide usage in artificial intelligence, such as the multi-layer perceptron. Outside the scope of computer science, neural networks have found applications in bioinformatics especially within the field of biological sequence analysis [39], though the linear architecture was eventually deemed limited [38]. The extended background research which I performed on Artificial Neural Networks and the Linear Perceptron was not included in this work.

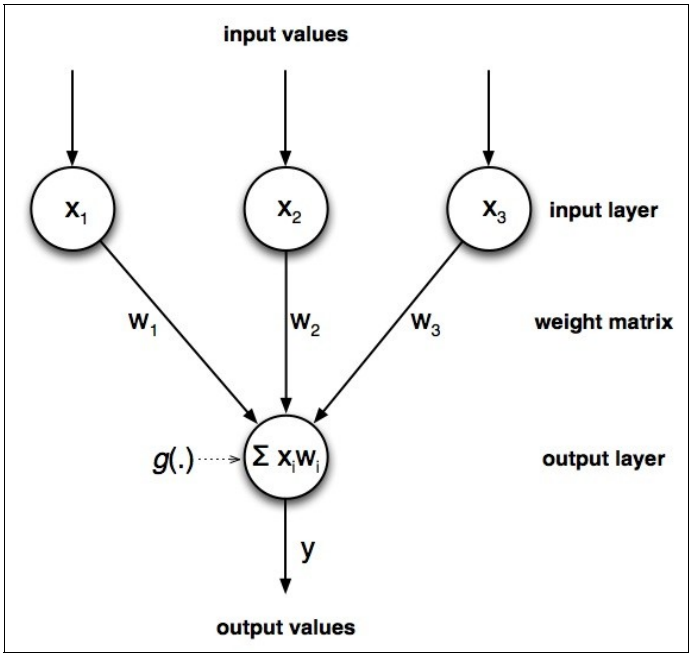

Figure 2.7: The Artificial Neural Network known as the Linear Perceptron. Input layer nodes represent input variables and the output layer node represents an artificial neuron. The Linear Perceptron uses a single artificial neuron in the output layer to perform binary classification via linear combination and a sign activation function g(.).

Artificial Neural Networks - Multi-layer perceptron (MLP) :

This type of Artificial Neural Network can be thought of as an enhanced version of the single-layer perceptron and is represented by a directed graph with several sequentially interconnected layers of nodes, where each of these nodes is a neuron associated to a

nonlinear activation function. The training process is achieved by means of a mathematical optimisation algorithm known as backpropagation. A multi-layer perceptron is able to map several sets of input data to a set of outputs and, unlike the single-layer perceptron, classify data that is not linearly separable. This algorithm has found popularity in artificial intelligence fields such as pattern recognition and, more recently, in bioinformatics for prediction of protein secondary structure [38][40]. The extended background research which I performed on Artificial Neural Networks and the Multi-layer Perceptron was not included in this work.

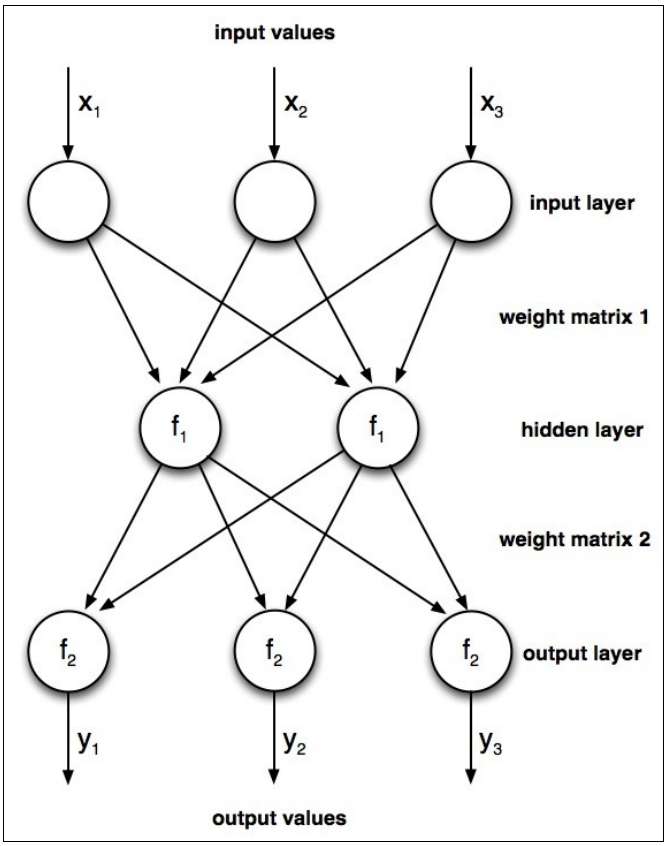

Figure 2.8: The Artificial Neural Network known as the Multi-layer Perceptron. Input layer nodes represent input variables while hidden and output layer nodes represent artificial neurons. The Multi-layer Perceptron is composed out of multiple inter-connected Linear Perceptrons and uses one or more hidden layers in order to approximate virtually any continuous function and perform nonlinear classification and regression.

k-Nearest Neighbours (*k*-NN) :

The *k-nearest neighbours* is a nonparametric unsupervised learning algorithm used for classifying objects based on the closeness of training samples (supervised information) in the feature space. In the training step of the algorithm all training samples are represented as vectors in a multidimensional feature space holding their known class labels. In the classification step, each object from the test set is sequentially introduced as a vector in the feature space and classified according to the most frequent label among its *k* nearest neighbouring training samples. The nearest neighbour class of estimators is one of the simplest machine learning algorithms and a type of *instance-based learning* [38]. The source code I developed for the method was not included in this manuscript.

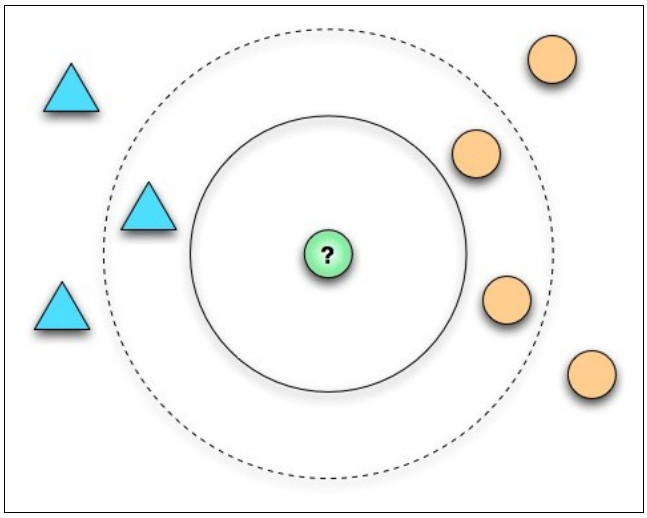

Figure 2.5: A graphical representation of k-NN classification. The green circular object in the middle will be classified along with circular objects on the right as there are two such objects in close proximity.

Naive Bayes classifier :

The *naive Bayes classifier* is a probabilistic classification algorithm which applies the Bayes theorem under loose assumptions of independence (naiveness). The classifier uses a conditional probabilistic model which can be trained efficiently in a supervised learning

setting where parameter estimation is usually achieved via the method of *maximum likelihood*. Despite its simplicity, the algorithm treats real-world Artificial Intelligence problems with substantial efficiency [38]. The source code I developed for the method was not included in this manuscript.

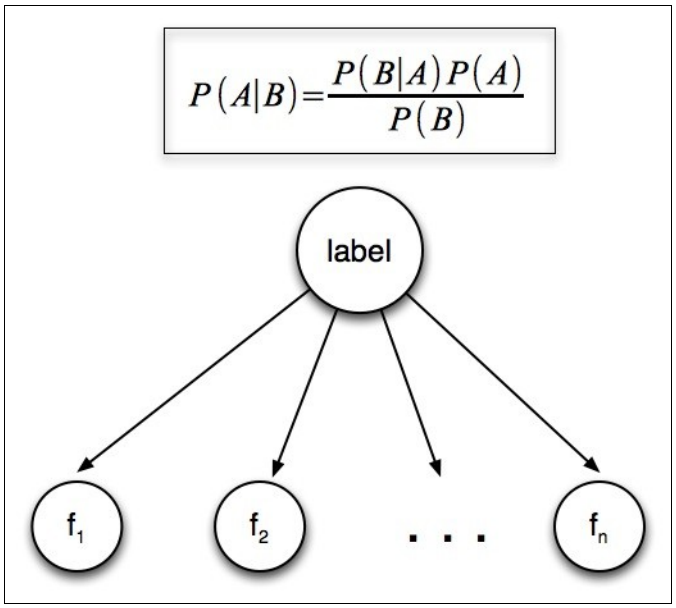

Figure 2.6: A graphical representation of the naive Bayes classifier. The Bayes rule is used to calculate the probability that an unknown object belongs to a certain class label. The parameters of the Bayes rule are estimated from the training sample, where object class labels are known.

Hidden Markov Models (HMM) :

A Bayesian network is a probabilistic directed acyclic graph that represents a set of random variables and their conditional dependencies. A Hidden Markov Model is a *dynamic Bayesian network* in which a system is modelled as a *Markov process* (a stochastic process possessing the Markov property), i.e. the conditional probability distribution of future states depends only on the present state and not on the past. HMM is a stochastic generative model for time series that is defined by a finite set of states, a discrete alphabet of symbols, a probability transition matrix, and a probability emission

matrix. Such a system evolves randomly from one state to another while emitting symbols from the alphabet according to the probabilities defined in the transition and emission matrices, respectively. The random walks between states are hidden but the emissions are observable, while the first-order Markov assumption is that both emissions and transitions depend solely on the current states and not on past ones. The hidden states are represented as hidden/latent variables underlying the observations [38]. HMM can be powerful supervised learning tools with applications in speech recognition, often combined with *reinforcement learning*. In bioinformatics, they have been used in biological sequence applications where the alphabet comprises the twenty-letter amino acids for proteins and the four-letter nucleotides for DNA/RNA problems. In metabolomics, the alphabet can be be composed by a set of chosen metabolites or reaction-equivalent mass differences [38]. The source code I developed for the method was not included in this manuscript.

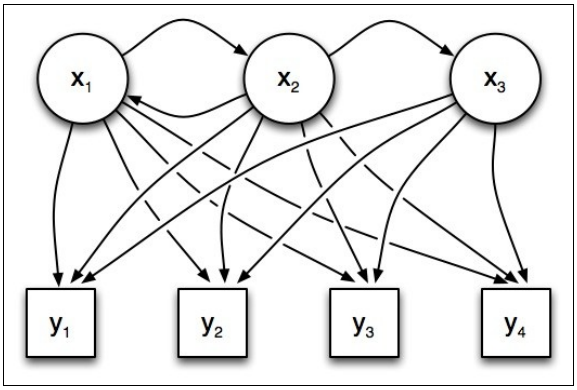

Figure 2.9: A graphical representation of HMM. The edges of the Bayesian network represent the transition and emission probabilities between different states and associated class labels (nodes).

Decision tree learning :

A *tree* is a common recursive data structure in Computer Science. A Decision tree in *symbolic learning* (not to be confused with Bayesian decision-making trees) is a hierarchical network that implements a divide-and-conquer strategy and can be used for

both classification and regression. In supervised learning, a decision tree is used a predictive graph model composed of internal decision nodes and terminal leaves. Every decision node implements a test function with discrete outcomes that labels the branches (edges). The algorithm starts at the root of the tree and iterates recursively until a leaf node is found (the value of that node is yielded as the output) [41]. A variant method is the *random forest classifier*, an algorithm that involves mixing and iterating over several decision trees and has many applications in metabolomics.

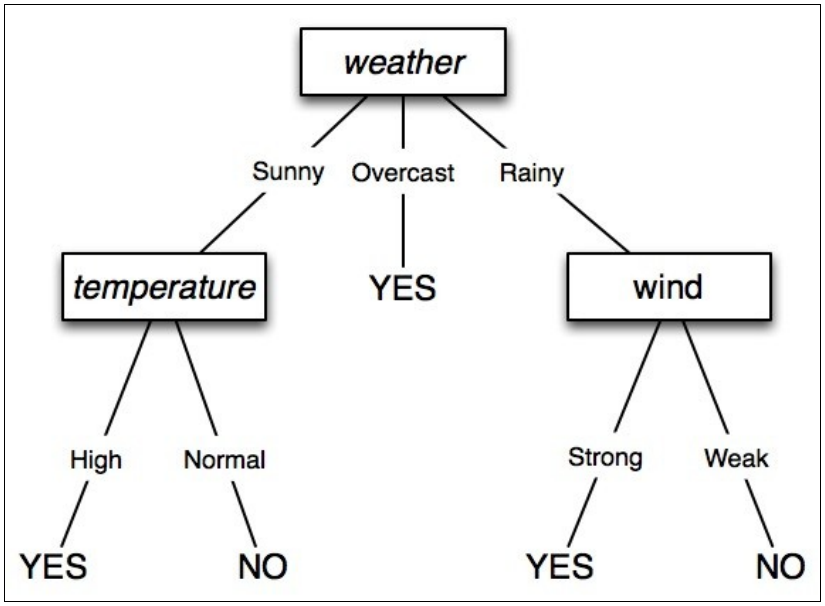

Figure 2.4: A graphical representation of a Decision Tree used for weather prediction. The interior nodes of the tree correspond to input variables with edges linking to children nodes for the possible values of those input variables.

Support Vector Machines (SVM) :

Support vector machines are a group of algorithms within the wider *kernel methods* family, in which a feature space is operated by *kernel functions* that calculate the inner products of the images of all points; an approach which can be computationally lighter than computing the points' coordinates. A basic linear SVM algorithm acts on a set of

input data as a binary linear classifier which predicts two possible classes (figure 2.5). This model can be extended to a non-linear multi-class algorithm via the kernel trick (the application of kernel functions on a multi-dimensional feature space).

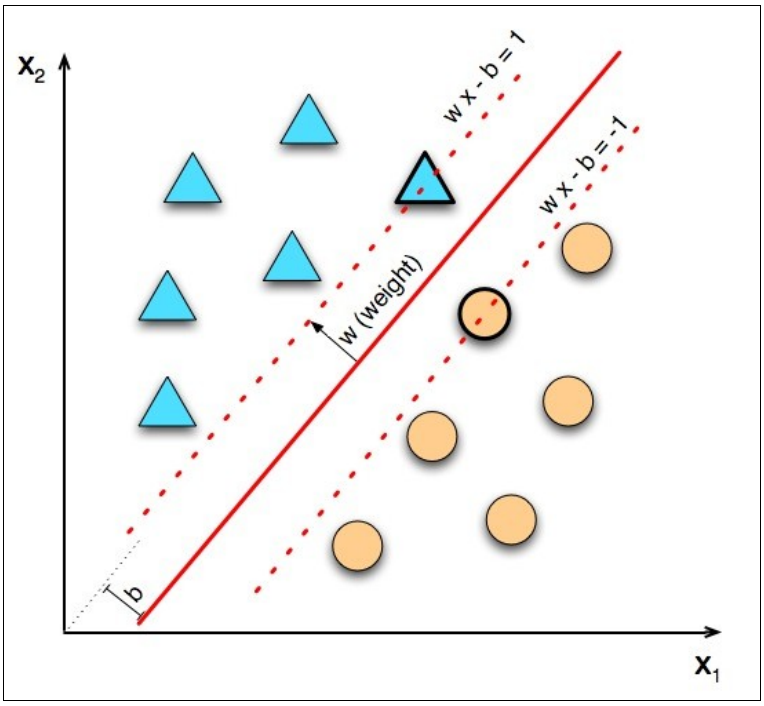

Figure 2.5: A graphical representation of a linear SVM. The dashed lines represent the two support vectors found at the edges of the two classes (triangular and circular shapes). The red line found in the midpoint of the two support vectors is the optimal hyperplane that separates the dataset.

2.6.2 Unsupervised learning (Clustering)

Unsupervised classification, commonly known as *clustering*, is a fundamental technique in data mining aimed at extracting underlying patterns from within the data without the input of some prior knowledge. Data is grouped into classes (or *clusters*) based on some measure of inherent similarity, typically by modelling data points as vectors in a Euclidean space [38]. In this case, learning and classification are performed as a single process on the whole dataset without having to divide it into training and test parts. As the

target values of the outputs in the fitted data are unknown or ignored, the terms *unsupervised* or *self-organisation* are used to describe the classification [38].

The starting point of a typical clustering algorithm is the construction of a matrix of pairwise similarities between the objects to be clustered. The choice of similarity metric is crucial and can have an impact on the algorithm's output. Examples of distance metrics are the Euclidean distance and the Pearson correlation coefficient, which is the dot product of two normalised vectors (or the cosine of their angle). Depending on the situation, every measure of similarity has its own advantages and drawbacks; a correlation metric, for instance, captures similarity in shape but does not place any emphasis on the magnitude of the two series of measurement while it remains sensitive to outliers [38]. A distance metric, though theoretically unit-dependent, can be normalised and expressed in the form of a correlation value in the range of zero and one. I describe my study on the performance of different clustering algorithms combined with varying similarity metrics in a different chapter. The clustering algorithms that were vital to the outcome of this work are briefly described below.

Hierarchical clustering :

Hierarchical clustering is an unsupervised classification algorithm that seeks to build up clusters through a hierarchical branching process. The method groups data over a similarity matrix by creating a cluster tree or *dendrogram*, which is not a single set of clusters but rather a multilevel hierarchy in which clusters at one level are linked as clusters at the next level [42]. As the output of the algorithm is a tree and not a set of clusters, it is not always so obvious how to define clusters from the dendrogram as they are derived by cutting branches at arbitrary points [38]. There are two strategies for the generation of a clusters from the branches of a hierarchical tree: agglomerative and divisive. In the agglomerative approach, each cluster starts at every individual branch at the bottom of the tree and pairs of clusters are joined together by moving up in the hierarchy. The divisive approach is a top-down strategy, in which a single cluster is recursively split up by moving down the tree hierarchy. Alternative strategies include mathematical optimisation and heuristic algorithms.

The agglomerative hierarchical clustering algorithm follows the following general steps [42]:

1. Initialisation: Find the similarity or dissimilarity between all pairs of objects in the data set according to a similarity metric that has been chosen. A similarity matrix is created from this step.

2. Linkage: Group the objects into a binary hierarchical cluster tree. Objects in close proximity are linked together via the distance information in the similarity matrix until a hierarchical tree is constructed.

3. Clustering: Determine where to cut the hierarchical tree into clusters. In the agglomerative strategy, branches are pruned off the bottom of the hierarchical tree and assign all objects below each cut to a single cluster. The data is partitioned by detecting natural groupings or by cutting off the hierarchical tree at an arbitrary point.

Outside the scope of Artificial Intelligence, hierarchical clustering has found many applications in bioinformatics for sequence analysis, phylogenetic trees, and average-linkage cluster analysis.

k-means clustering :

The *k*-means clustering algorithm partitions objects in a data set into *k* clusters, where every object belongs to the cluster of the nearest mean (after a mean value has been calculated for each cluster). Classification is achieved by minimising the sum of squares of distances between data points and the corresponding cluster centroid. In the canonical implementation of *k*-means, the number of clusters is fixed to a value *k* as part of the algorithm's input. A number of *k* representative points, called *centroids*, are selected during the step of initialisation, then at each iteration the algorithm performs the following steps until convergence [38]:

1. Assignment: Each point in the data is assigned to the cluster of the closest centroid.

2. Update: New centroids are computed by aggregating (averaging or taking the centre of gravity) the members of each computed cluster.

In most occasions, *k*-means acts as an online variant of the generalised EM (expectation maximisation) algorithm, where the assignment step is also referred to as *expectation step* and the update step as *maximisation step*.

In computational complexity theory, the described problem of *k*-means clustering is *NP-hard*, i.e. a class of problems which are too complex to be efficiently treated by exact algorithms; however, there exist efficient heuristic methods, such as the *expectation-maximisation algorithm*, that can be employed in order to yield an optimal solution. As a nondeterministic heuristic algorithm, there is no guarantee that it will always converge to a global optimum nor that it will always yield the exact same results on the same dataset (mathematical optimisation theory is explained in the last section of this chapter). Nonetheless, in practice *k*-means is one of the most efficient clustering algorithms in both terms of speed and quality of classification.

Artificial Neural Networks - Self-Organizing Maps :

A *Self-Organizing Map* (SOM) (or a *Kohonen network*) is a type of Artificial Neural Network, in which training is achieved via unsupervised learning in order to produce a low-dimensional representation of the multi-dimensional input space that is called a *map*. This dimensionality reduction is achieved by a data compression technique known as *vector quantisation.* Unlike other clustering algorithms, SOMs follow the practice of supervised learning techniques which involves a training and testing/mapping steps. However, in contrast to other neural networks, the SOM algorithm manages to classify the data without supervision, preserve all topological information of the training set, and offer meaningful visualisation of high-dimensional data. SOMs can deal efficiently with very large datasets and are, to date, regarded as one of the most efficient unsupervised learning methods. Its principal downsides are the difficulty to pinpoint clusters with

precision and the often high computational demands of the training process. My background research on Artificial Neural Networks was not included in this manuscript.

Community structure partition (graph clustering) :

The *community structure partition* of a network consists of finding natural groups of nodes with dense connections within the groups and sparser connections between them (figure 2.5). An optimal community structure partition consists of maximising the number of within-module connections while minimising the number of between-module connections. Therefore, for any given node in such a partition there is a higher probability to have a connection inside the same module rather than outside it. By using a type of activation function that sets a threshold value on a similarity matrix, any given dataset can be represented in the form of a modular network whose optimal community structure partition is a cluster analysis of its input space. As in the case of *k*-means, the problem of finding an optimal community structure partition is NP-hard and can be only solved efficiently via combinatorial optimisation.

There are many computational techniques to extract a community structure partition from a modular network ranging from hierarchical clustering to various optimisation models. One of the most optimal ways to achieve this task (and the one used in this work) is the so-called *modularity optimisation* model. The *modularity* of a network, computed as a real number between zero and one, is a measure of how efficient a given community structure partition of a network may be in terms of inter-modular and and intra-modular connectivities and can be though of as an objective function which measures the quality of a particular division of a network into several clusters. The arising optimisation problem consists of maximising that function.

Due to the lack of specialised software tools, community structure partition is, as of yet, not a standardised method for data clustering. As shown in a different chapter, however, the output of this clustering is similar to, or outperforms, most conventional methods while offering a number of additional advantages. This clustering algorithm was the method of choice for the computational frameworks developed in this work.

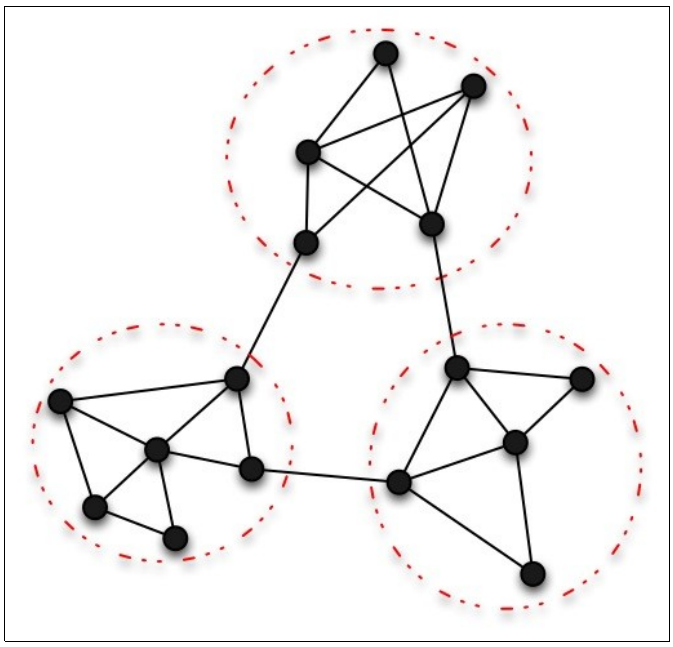

Figure 2.5: A community structure partition of a graph with three modules (highlighted in red dashed circles). The clusters of densely interconnected nodes are called "communities" and reveal natural patterns in the data.

2.7 Mathematical optimisation

So far, we have mentioned the term *optimisation* in a quantitative context many times in this chapter, albeit without having clarified what it stands for. Indeed, *mathematical optimisation* (also known as *mathematical programming*) is a very important concept in machine learning, used by most computational training algorithms. In this section I describe in more detail the background of mathematical optimisation in the context of computational and quantitative research. In a later chapter, I describe how mathematical programming can be used to build up a whole new framework to deal with the quantitative problems of metabolomics and possibly other fields of bioinformatics.

2.7.1 Optimisation in Operational Research

The term *optimisation* refers to the search for an optimal state; a very old and important concept in our world. The process of optimisation is part of life and a fundamental principal in nature, e.g. how atoms try to form bonds in order to minimise the energy of their electrons, how molecules form solid bodies during the process of freezing by assuming energy-optimal crystal structures, or how biological evolution leads to a better adaptation of the species to their environment. In addition, optimisation has pervaded all spheres of human endeavor and extends into daily life where humans try to optimise their resources by minimising their cost while maximising their gain [43]. It has, therefore, become imperative to plan, operate and manage resources in an optimal manner, yet with the advent of computer science it is possible to exploit optimisation theory to its maximum extent [44]. To date, the quantitative discipline which has incorporated and developed the study of mathematical optimisation is known as *Operational Research*.

In mathematics, an optimisation process searches for optimal states of maxima or minima. For complex systems where many decisions need to be made simultaneously, it becomes necessary to take optimal decisions based on prior knowledge and heuristics. Mathematical optimisation theory provides a scientific alternative for decision-making in complex situations where the system can be modelled mathematically [43]. Optimisation algorithms were developed to provide solutions to optimisation problems (known as *mathematical programming problems*) in linear, nonlinear, unconstrained, and constrained domains. Eventually, special strategies were designed for the purpose of seeking *global optima* to those complex problems, where older programming methods would get stuck at a *local optimum* [44]. *Global optimisation* focuses on complex optimisation, where the goal is to find the best possible elements x^* from a set X according to a set of criteria $F = \{ f_1, f_2, \ldots, f_n \}$. These criteria are expressed as mathematical functions, known as *objective* or *evaluation functions*. Formally, an objective function $f : X \rightarrow Y$ with $Y \subseteq R$ is a mathematical function which is subject to optimisation. The range Y of an objective function f must be a subset of the real number set R, while the domain X of f (known as the *problem space*) can represent any type of

object such as numbers, vectors, sets, etc. Objective functions can often be in the form of complex algorithms, significantly more complicated than plain mathematical expressions. Global optimisation involves all algorithms that can be used to find the optimal element x^* in X with respect to the criteria of $f \in F$ [43].

Optimisation can be described as a three-step decision-making process [45] :

(i) *Process modelling*: Represent a real-world system as a mathematical problem and a potential solution to that problem as a mathematical object

(ii) *System evaluation*: Find a measure of system effectiveness via an objective function which evaluates a potential *solution* to the problem

(iii) *Model optimisation*: Apply specialised search algorithms to maximise or minimise the objective function and provide an optimal solution to the problem

Optimisation problems can be divided into *continuous* or *discrete*, depending on whether the variables may take on real or discrete values, respectively [46].

2.7.2 Continuous optimisation

Continuous optimisation deals with the case of mathematical programming where variables can take on any values permitted by the constraints [46]. Depending on the mathematical model involved, continuous optimisation can be *linear* or *nonlinear* in nature. In *linear programming*, an optimisation problem consists of minimising or maximising a linear function under linear constraints, which comes down to a mathematical problem P_0 of the form:

Minimise objective function $f(x)$ under the constraints:

$$P_0 \begin{cases} g_i(x)=0, \ \ i\in I^0 \\ g_i(x)\le 0, \ \ i\in I^- \\ g_i(x)\le 0, \ \ i\in I^+ \\ \\ x=(x_{1,}x_{2,}\dots,x_n)^r\ge 0 \end{cases}$$

where functions f , $g_i\left(i\in I=I^0\cup I^+\cup I^-\right)$ are linear functions of the variables $x_{1,}x_{2,}\dots,x_n$ [47].

A continuous optimisation problem has a feasible solution when its objective function is analytical and differentiable. As this is not the case with the models in this work, further information on continuous optimisation is outside the scope of this manuscript.

2.7.3 Discrete optimisation

In *discrete optimisation*, whether integer or combinatorial, variables may take on discrete (typically integer) values. Those problems are usually computationally *hard* and require the use of smart algorithms to be solved [46].

In this work, we deal mainly with the subset of global optimisation known as *combinatorial optimisation* that consists of finding an optimal object from a finite set of objects using a single criterion $f \in F$. Combinatorial optimisation makes use of discrete models which are based on *combinatorics* (the branch of mathematics dealing with the study of finite or countable discrete structures) rather than analytical differentiable functions as in the case of continuous optimisation. A real-world scenario is modelled as a combinatorial problem whose near-optimal solution needs to be found. Combinatorial problems are classified according to their individual complexity, which is calculated by the time a deterministic/exact search algorithm requires in order to find the one and only optimal solution to that problem; a branch of computer science and mathematics known as *computational complexity theory*. Complexity theory is a central topic in the theoretical foundations of computer science, concerned with the general study of the

intrinsic complexity of computational tasks (algorithms). When the dimensionality of the search is high, it becomes almost impossible to solve a problem deterministically by an exhaustive enumeration of the search space. The complexity of an exact search algorithm would in most cases be too high to ever reach a convergence point, therefore “smarter” stochastic algorithms are considered. A *heuristic* in combinatorial optimisation is a function which makes an algorithm “smarter” by providing some prior information that will guide the search. A *metaheuristic* algorithm is a method for solving general classes of problems by combining heuristics and objective functions in an abstract but efficient way, which significantly reduces search space. This combination is typically performed as a stochastic process by using statistical and probabilistic information from the search space itself, or by imitating the optimisation patterns of a natural phenomenon (e.g. Genetic Algorithms, Ant Colony Optimisation algorithms, etc.).

2.7.4 Approximation and metaheuristic algorithms

Gradient descent :

The basic descent method is a generalisation of the *gradient descent algorithm* for continuous optimisation. In gradient descent optimisation, a differentiable function $\phi_0(x)$ is minimised by moving along the direction of the negative gradient of the objective function at a starting point x_1, i.e. $-\nabla \phi_0(x_1)$. The algorithm iterates from an initial guess until it converges to a maximum iteration count $n_{\max}$:

$$\forall t \in \{1,2,\ldots, n_{max}\}, x_{t+1} \leftarrow x_t - \gamma_t \nabla \phi_0(x_t)$$

The smaller the step size γ_t is, the slower the algorithm will converge but if the step size is too big then the algorithm may fail to converge (divergence). The gradient $\nabla \phi_0$ is the neighbourhood structure operation N in the basic framework. For a non-differentiable function that cannot be minimised analytically, the neighbourhood operation N is implemented by means of an intelligent algorithm, a concept which gives birth to the basic descent search method. The principle of the basic descent method is to start from a solution λ and choose a solution λ' in the neighbourhood of λ, such that $\phi_0(\lambda') \leq \phi_0(\lambda)$.

The step size γ_t is implemented within the neighbourhood structure N. The algorithm usually examines all neighbour solutions and picks out the best one. If the quality of the solution does not increase after a certain number of iterations, then a stochastic neighbourhood structure N_s is employed in order to escape a possible local optimum after a fixed number of iterations without improvement. The source code I developed for Gradient Descend variants was not included in this manuscript.

Simulated annealing :

Simulated annealing is a global optimisation algorithm with many applications in *statistical mechanics*. It is a *Monte Carlo method* that can be applied to arbitrary search and problem spaces [43]. Its name comes from metallurgy and material science, where *annealing* is a heat treatment of matter with the purpose of affecting properties such as hardness [43]. Metals that have been cooled down slowly (annealed) are harder than metals that have been cooled rapidly, therefore the magnitude of macroscopic strength is proportional to the molecular states of lower energy [38]. Heating the metal increases the energy of the ions and their diffusion rates [43]. As in the case of the gradient descent, the algorithm requires a single initial solution as a starting point of the search [43]. Contrary to gradient descent, the closeness of this initial solution to the global optimum of the search space should not theoretically have an impact on the outcome of the optimisation. The simulated annealing algorithm calculates iteratively the energy of the state of a metal during its cooling process in respect to a function known as *annealing schedule*. As the temperature decreases, the energy of every state is calculated by the problem's objective function and the algorithm converges to an optimal solution. It has been shown [48] that for a logarithmic annealing schedule of the form:

$$T^t = \frac{K}{\log(t)}, t \geq 1$$

where T^t is the temperature at time *t*, the algorithm does almost certainly converge to one of the ground states, for some constant value *K*. In practice, a logarithmic annealing schedule is slow and impractical as it suggest that a very large number of all possible

states is visited, making it equivalent to an exhaustive search [38]. In the case of combinatorial optimisation, a metaheuristic is usually applied to a problem which is of NP-hard complexity, where the number of possible solutions increases exponentially with the size of the problem and therefore an exhaustive search needs to be avoided. In order to use simulated annealing in metaheuristic search with positive results, we use a geometric annealing schedule of the form:

$$T^t = \mu T^{t-1}, \quad 0<\mu<1 \quad [38]$$

A positive result in this case would be an approximate solution corresponding to points of low energy and not necessarily a global optimum [38]. However, it has been additionally shown that Simulated Annealing algorithms with appropriate cooling strategies will manage to asymptotically converge to the global optimum [43]. The source code I developed for Simulated Annealing was not included in this manuscript.

Genetic Algorithms :

Genetic Algorithms (GA) are a subclass of *evolutionary algorithms*, a broad family of optimisation algorithms whose source of inspiration is the Darwinian theory of evolution. Evolutionary algorithms try to simulate the inner mechanisms of the evolution of the species as it is understood to date. These algorithms are characterised by the generation of random perturbations (called *mutations*) and the presence of a *fitness function* which evaluates the quality of a given solution. Genetic Algorithms, one of the broadest subclasses of evolutionary algorithms, simulate the evolution of populations of points in fitness space. In Genetic Algorithm terminology, a potential solution to an optimisation problem is called a *chromosome* or *individual*, whereas a set of solutions from which new individuals will have to emerge is called a *population*. An individual is canonically modelled in the form of a binary vector, where every bit is subject to change and is called a *gene*. In addition to gene mutations, individuals are produced by simulating genetic operators and reproduction, such as *crossover*, which involves the recombination of bits from selected solutions [38]. Genetic Algorithms are specifically designed for global optimisation by producing an entire new population (set of candidate solutions) on every

iteration, without the need nor possibility of using a heuristically good solution as a starting point.

The generic process of a canonical Genetic Algorithm applies the following steps:

Step 1: Initialisation – An initial population is created, usually as a random process using a defined fitness function.

Repeat until convergence:

Step 2: Selection/Evaluation – A subset of the fittest individuals is chosen from the current population.

Step 3: Mating/Crossover – A new population is created by the reproduction of the fittest individuals picked out during selection.

Step 4:Mutation – The genetic makeup of each newly-born individual produced in crossover are subject to mutation given a defined mutation probability.

Sample applications of GA in molecular biology biology can be found in [49][50][51]. The source code I developed for Genetic Algorithms was not included in this manuscript.

CHAPTER III

Structural mass difference networks and the Netcalc method

This chapter presents the part of my work mentioned in the Overview (section 1.1) as the *structural network approach*. I describe the enhanced structural mass different network reconstruction scheme and the *Netcalc* algorithm for network-based formula annotation of exact masses. I present my reconstruction and inference algorithms along with our results and applications on real-world data. At the end of the chapter, I present a stand-alone software application which materialises the main points of our approach and allows for a more flexible algorithm parametrisation.

3.1 Method description and application on organic aerosol

3.1.1 Abstract

Here, we propose a novel computational and visual approach for the analysis of high field Fourier transform ion cyclotron resonance mass spectra (ICR-FT-MS) based on successive and multiple atomic and Kendrick-analogous mass difference analyses. *Compositional and functional networks* enable improved assignment options of elemental composition and classification of organic complexity with tunable validation windows. The approach is demonstrated through the analysis of a 12T ICR-FT-MS mass spectrum of an intricate water soluble extract of a secondary organic aerosol with a previously established abundance in CHNOS molecules [2].

3.1.1 Introduction

A precise molecular description of NOM, based on their carbon, hydrogen, oxygen and heteroatom-bearing formulae, facilitates the understanding of environmental biogeochemical processes. One critical bottleneck in the ICR-FT-MS characterisation of NOM concerns the handling and meaningful visual display of large datasets. Information-rich, composition- and structure-dependent visualisation tools are indispensable for any significant mass spectrometric analysis of NOM and other complex organic mixtures [52]. Van Krevelen visualisations [53][54] and Kendrick mass analyses [53][55] were initial methods of choice to depict the chemical diversity in various NOM and to provide unequivocal assignment of CHO, CHNO, CHOS, and CHNOS molecular series [11][56] [57] and functional group equivalent frequencies [1], respectively. A Van Krevelen diagram illustrates the hydrogen/carbon (H/C) ratio of molecules against their oxygen/carbon (O/C) ratio. The initial motivation behind this scheme was to describe the composition of coal, though it is currently in widespread use in the fields of petroleomics and NOM research [58][10]. In addition, it has been reported that the information in Van Krevelen diagrams may reveal patterns of metabolite classes [11][59].

Mass difference analysis allowed assignment of higher m/z peaks in mass spectra of NOM by extrapolating from lower mass numbers within recurring molecular series [60]. In this chapter, I present a new approach of network reconstruction and visualisation of high field ICR-FT-MS mass spectra which offers expedient assignment of elemental formulae with improved coverage. In addition, network analysis of intricate mass spectra offers new unambiguous means to depict relationships between functional group equivalents, transformations and organic molecular complexity in general.

3.1.2 Materials and methods

Samples and mass spectrometric analysis

The negative electrospray 12 T ICR-FT-MS mass spectrum of a secondary organic aerosol (SOA) has been previously acquired as described in Schmitt-Kopplin et al [57]. With a time domain of 4 megawords and a transient of 1.6 sec, it shows a resolution in

excess of 400.000 at m/z 400 in full scan mode, allowing for a decent coverage of reliably assigned elemental compositions after internal sub-100ppb linear calibration using fatty acids in the m/z range 150 to 600. All analysed ions were found singly charged.

Computational analysis

To model the mass spectrum of SOA we used a large undirected graph $G=(V, E)$ in which the set of vertices V represents the obtained exact masses from a m/z list, and the set of edges E represents predefined differences Δm from a transformation list. During network reconstruction, a polynomial-time online algorithm performed an exhaustive search across the m/z list of experimental masses comparing all differences between those masses and the theoretical mass differences Δm found in the transformation list, within an adjustable error margin (set to 100 ppb here). Any match detected between experimental and theoretical Δm established a conditional relationship and stored specific colour-coded values inside an adjacency matrix according to mass fragment specifications. Network reconstruction, statistical analysis, and network-based formula calculation were programmed in Matlab. Visualisation was performed on Mathematica using the algorithm of Hu [61]. The source code I developed for the algorithms used in the method was not included in this manuscript.

3.1.3 Empirical results

Generation of mass difference networks

Our networks are created out of exact mass differences in a way similar to the process described in Breitling et al [1]. The ICR-FT-MS mass spectrum of SOA initially served to produce a mass list by standard good practice [12][13][56][62]. A transformation list of preselected mass differences Δm corresponding to abundant specific small molecular units has been independently proposed. This level of abstraction provides information on the chemical variation in the sample under study by depicting sort (Δm) and site (m) of transformations between different network modules. Two types of mass difference networks were computed via this semi-targeted approach:

(i) *Compositional network* reconstruction was applied using an element-based transformation list (figure 3.1). A mass difference network was constructed on the fly out of the m/z list and a transformation list composed solely of selected elements (i.e. C, H, O, N, S, P) and mass differences between two main isotope pairs (*ΔC*: ^{13}C-^{12}C, and *ΔS*: ^{34}S-^{32}S). The resulting sparse matrix can be used without further analysis to characterise any complex bio(geo)chemical system; this will be exemplified by a mass spectrometric analysis of a secondary organic aerosol in the following section (figure 3.1). The goal of this model is the visual and mathematical evaluation of organic molecular complexity in terms of elemental composition; an approach that permits explicitly defined relationships between different samples.

(ii) *Functional network* reconstruction was applied using a restricted list of selected small molecular units with defined *Δm*, corresponding to common chemical functional group equivalents (which can be commonly inserted in between any C-C and C-H bond, such as: CH_2, H_2, O, CO_2, S, SO_3, NH, PO_3, *ΔC* and *ΔS*) and transformations. Expansion to include any further elements (e.g. B, Cl, Br, Fe, Mn,…), isotopes (e.g. $^{35/37}Cl$, $^{79/81}Br$,…) and combinations thereof is rather straightforward and easily implemented. *Functional network* reconstruction corresponds to a multiple Kendrick analogue mass defect analysis and generates all homologous series according to chosen transformations simultaneously. Specific colours are attributed to any *Δm*; for figure 3.2, blue (CHO compounds), green (CHOS), orange (CHNO) and red (CHNOS) were used.

Network topology and visualisation

Compositional and functional network visualisation in two or three dimensions was enabled by means of a multilevel, force-directed, layout algorithm[61]. Highly connected nodes were arranged near the center and less connected nodes towards the periphery. Our networks display a modular structure with a power-law degree distribution, i.e. node connectivity is high for very few nodes and low for the majority of nodes. This so-called “scale-free” topology has been previously found characteristic of this type of network [1] [31]. The numerous clusters of nodes that can be observed on a visualised network are the result of its community structure partition, i.e. the natural grouping of highly connected nodes. The visual display of functional networks, in which the colours of edges

correspond to specific transformation groups (*Δm*), can be decomposed according to *Δm*. Recognition of additional fragments with given mass differences will gradually add new connections to the system (figure 3.2). Applying this functional network analysis to the water soluble fraction of SOA enabled us to confirm the particular sulfur and nitrogen functionalisation of secondary organic aerosols, which had been previously established from element-edited van Krevelen diagrams [57].

Network-based elementary formula calculation *(Netcalc)*

In the analysis of SOA [53][57], a reconstructed mass difference network enabled the assembly of dense, larger graphs with a wealth of connectivities (figure 3.2C, 3.2E, 3.2F) and dissection down to individual, open subgraphs, which then provided only a very few different types of transformations (*Δm*; figure 3.2A, 3.2D). In general, a network-based elemental formula calculation for any individual (disconnected) subgraph, in which all edges are assigned specific *Δm* (transformations), is feasible if only a single elemental formula is known (figure 3.2D).

Our initial "starting point" formula of SOA analysis was selected randomly from a list of calibrated formulas obtained from the m/z signals in its ICR-FT-MS mass spectrum via classical methods presented in our previous study [57]. About 70% of those m/z were detected in the largest subgraph of our compositional network (figure 3.2F). For validation, we selected those masses of the calibrated list present in the largest subgraph of the network and compared one-by-one the formulas assigned by Netcalc to those of the classical method. 99.57% of the formulas were found to be similar between the classical and network-based approaches at 0.1 ppm. Using carbon (13/12C) and sulfur (34/32S) isotope filtering methods and additional imposed H/C, N/C, S/C and O/C ratio and valency restrictions, we managed to refine the results by eliminating false annotations out of the total formula attributions of the main subgraph, leaving us with a larger count of acceptable formula annotations than previously obtained with the classical approach [57].

Applying this network analysis on the water soluble fraction of SOA enabled us to confirm the particular sulfur and nitrogen functionalisation of secondary organic aerosols (figure 3.2F), which had been previously established from element-edited van Krevelen diagrams [57].

3.1.4 Conclusion

Mass difference network analysis of high field ICR-FT-MS mass spectra offered improved visualisation and evaluation tools for mass spectra of complex organic mixtures, such as NOM. *Compositional* and *functional* networks created out of mass difference network analysis were complementary to any of the methods currently in use, such as van Krevelen diagrams and single fragment Kendrick mass analyses.

Adjacency relationships between nodes in compositional and functional networks allowed to mathematically model and visually depict organic molecular complexity. Compositional networks enabled assignment of elemental formulae out of mass spectra and allowed alignments according to compositional relationships.

Edge-coloured *functional networks* offer not only visually attractive but essentially very informative assessment of relationships between functional groups in complex organic mixtures, which can be adapted to any CHNOPSZ containing functional group (Z: any chemical element or combination thereof) to elucidate, e.g. selective reaction mechanisms or abundance windows according to element composition. Network analysis expands the classical single fragment Kendrick mass analysis and finely complements element-edited van Krevelen diagrams and other mass-dependent visualisation schemes (figure 3.3).

With the example of SOA characterisation, functional network analysis demonstrated that sulfur and nitrogen functionalities were present in two different molecular populations. Furthermore, we were able to confirm and improve our previous results [57] on the annotations of elemental composition.

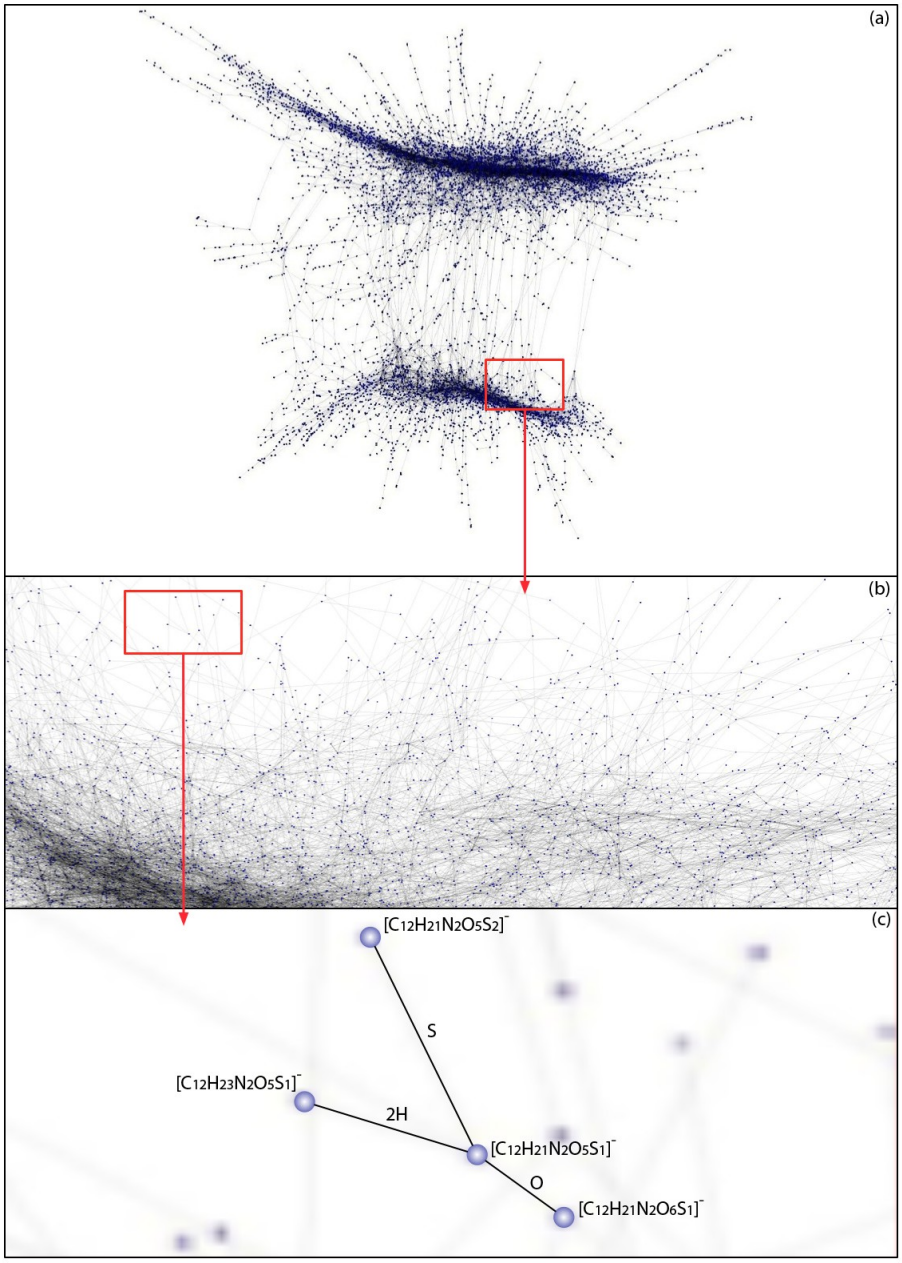

Figure 3.1: (a) Compositional network derived from a negative electrospray 12T ICR-FT-MS mass spectrum of a secondary atmospheric aerosol (SOA) with 9199 interconnected nodes out of 16933 masses and 35123 connections (edges). Given nodes represent assigned compositions, whereas connecting edges represent identified atomic transformations between pairs of masses. (b, c) Successive expansions down to individual compositions and element transformations.

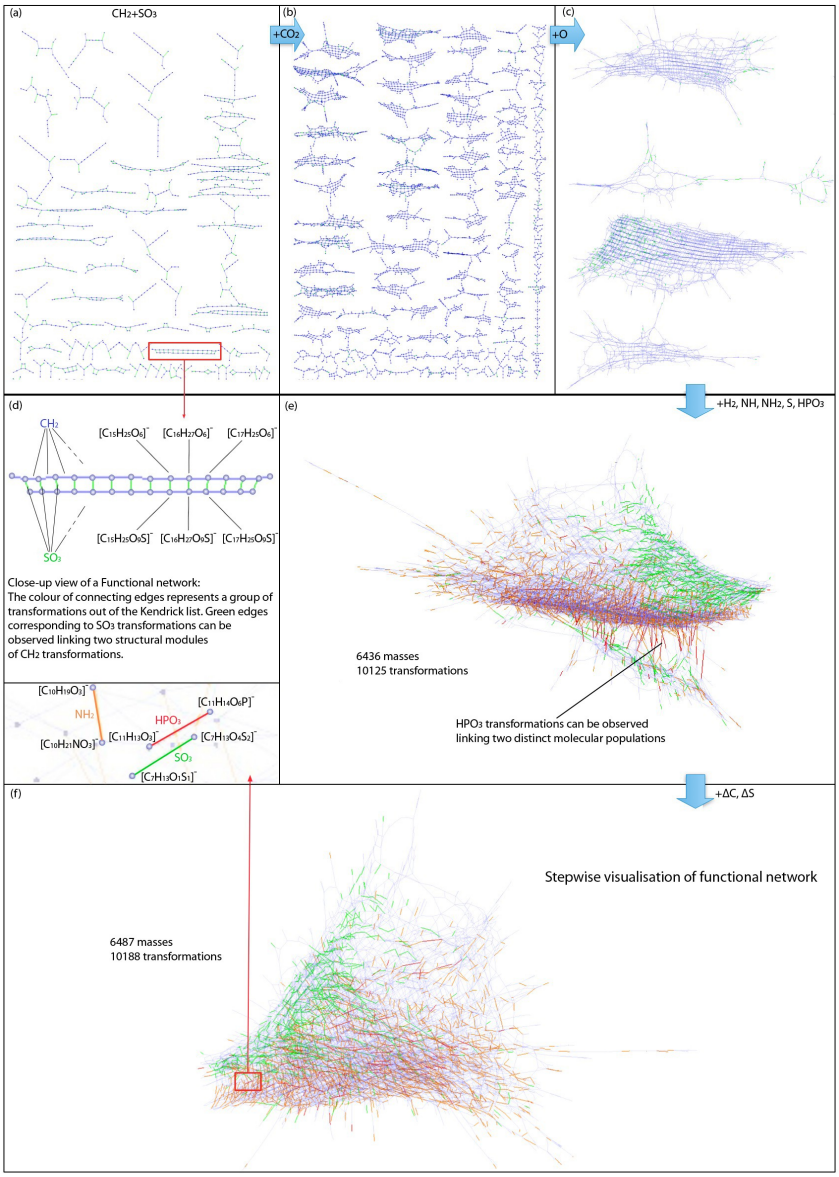

Figure 3.2: Functional network derived from SOA (cf. figure 1): The visualisation can be adapted to depict selected parts of the network, i.e. a range of choice for multiple transformations characterized by specific Δm. (a) functional network solely based CH_2 and SO_3 transformations; (b) inclusion of CO_2 (nominal carboxylation) produces more numerous extended networks; (c) nominal oxygenation produces a few large networks, which are then further integrated and condensed (figure 2e) by recognition of nitrogen, sulfur and phosphorus functionalities; (f) filtering according to isotope pairs $^{13/12}C$ and $^{34/32}S$ (ΔC and ΔS) removes false positive assignments; differential opacity enables in-depth view.

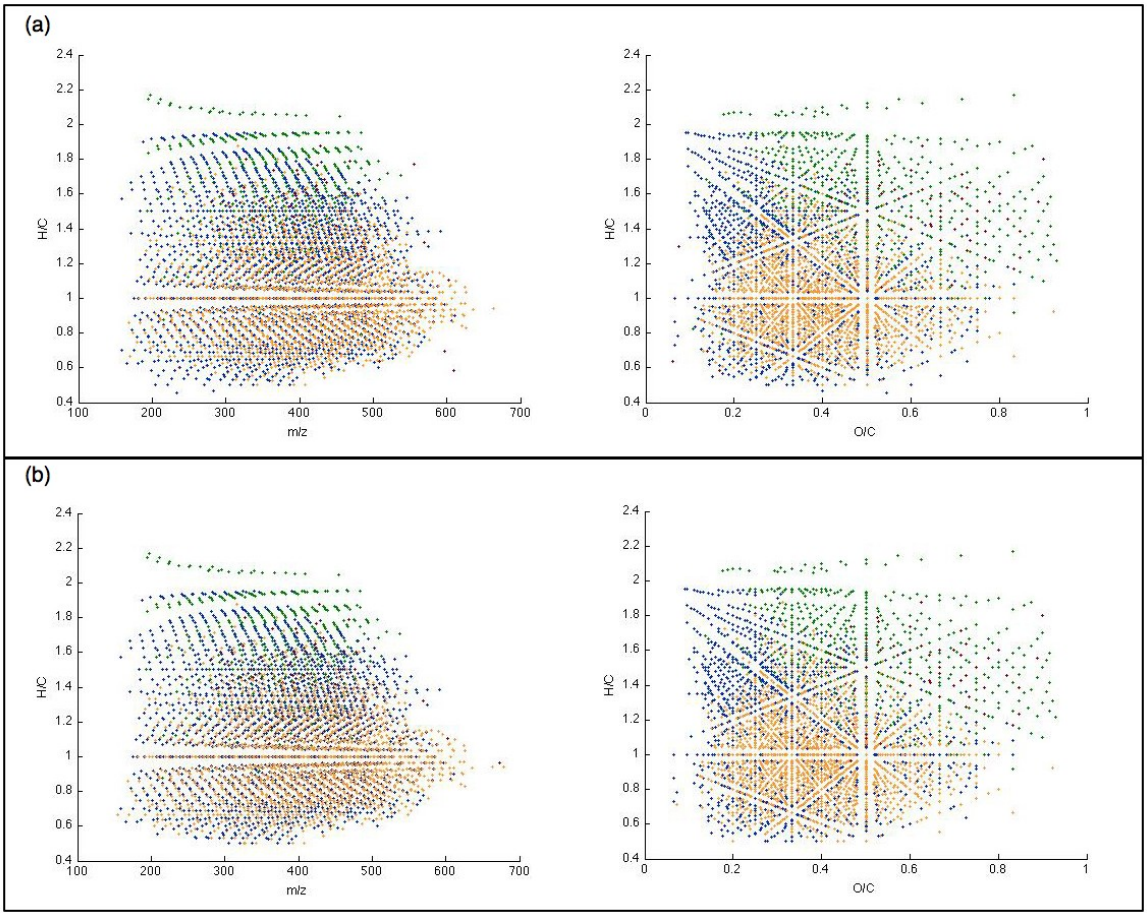

Figure 3.3: Comparison of (a) classical and (b) Netcalc computation of molecular compositions of SOA with mass-edited H/C ratio (left) and element-edited van Krevelen diagrams (right).

3.2 Models and Algorithms

3.2.1 Network-reconstruction algorithm

Structural network reconstruction requires a list of *n* exact masses and a list of *m* known chemical transformations which are supposed to be taking place between those masses. An exhaustive comparison is performed between all possible mass differences of the *n* masses and the *m* transformations in order to detect which transformations are taking place between which pair of masses. In every iteration, the algorithm calculates a mass difference and compares it to all *m* potential transformations within a certain ppm window until a match is detected (normally there should be only one match). The information of detected transformations between pairs of exact masses is stored as non-zero values inside a $n \times n$ sparse matrix. In a sparse matrix only non-zero values are accounted, hence the real amount of memory allocated by this process is far smaller than the maximal value of $n \times n$ which a full adjacency matrix would require. All matrix algebra in my algorithms assumes that calculations are made over a full adjacency matrix *A* of size $n \times n$, noted as $A = [a_{ij}]_{n \times n}$, whereas in reality the memory usage corresponds to that of an adjacency matrix ${}^{S}A$ of size:

$$\sum_{i=1}^{i=n} \sum_{j=1}^{j=n} a_{ij}$$

For a *n* number of exact masses and a *m* number of transformations, The reconstruction algorithm performs a total of $\frac{n^2 \times m}{2}$ iterations.

The complete reconstruction algorithm is provided below :

ALGORITHM 1: *Structural network reconstruction*

Let $X=\{x_{1,}\dots,x_i,\dots,x_n\}$ be a vector representing *n* exact experimental masses.

Let $Y=\{y_{1,}\dots,y_i,\dots,y_m\}$ be a vector representing *m* transformation masses.

Let T_{ppm} be a parts per million tolerance factor between two given masses.

Let f_{sparse} be a sparse matrix constructor function and $\mathbb{Z}$ be the integers set.

Input: exact mass vector [*X*], transformation vector [*Y*], ppm factor [T_{ppm}]

Output: sparse matrix ${}^{S}A=[{}^{S}a_{ij}]_{n\times n}$

$I\leftarrow\{\}$, $J\leftarrow\{\}$, $K\leftarrow\{\}$

$\forall\, i\in\{1,\dots,n\}, i\in\mathbb{Z}$

$\forall\, j\in\{i+1,\dots,n\},\ j\in\mathbb{Z}$

$\Delta m\leftarrow|x_i-x_j|$

$T\leftarrow\dfrac{T_{ppm}\times(x_i-x_j)}{2}\times10^{-6}$

$\forall\, k\in\{1,\dots,m\}, k\in\mathbb{Z}$

if $[\Delta m<y_k+T]\wedge[\Delta m>y_k-T]$

$I\leftarrow\{I\cup i\}$

$J\leftarrow\{J\cup j\}$

$K\leftarrow\{K\cup k\}$

end-if

end-for

end-for

end-for

${}^{S}A=f_{sparse}(I,J,K)$

My source code on various reconstruction algorithms was not included in this manuscript.

3.2.2 Disconnected subgraph clustering algorithm

The sparse matrix created by the network-reconstruction algorithm can be used to either visualise or further analyse the mass difference network of the input spectra. In order to perform network-based formula annotation, the mass difference network needs to be clustered into its all its disconnected graph components (subgraphs). For this purpose, I implemented a subgraph clustering algorithm which breaks down a disconnected graph. The algorithm outputs matrices ${}^{I}C$ and ${}^{J}C$, which hold the indices of all connections of all disconnected subgraphs, such that:

$$ {}^{S}_{q}A = f_{sparse}\left({}^{I}C(q,*), {}^{J}C(q,*)\right) $$

where ${}^{S}_{q}A$ is the sparse matrix of disconnected subgraph number q (all columns of row q in matrices ${}^{I}C$ and ${}^{J}C$).

The number of rows in ${}^{I}C$ and ${}^{J}C$ (which are of equal size) is the number of disconnected subgraphs discovered in ${}^{S}A$, and the number of columns is the number of connections in the largest of these subgraphs. Thanks to the sparsity property, the large unused space in the the matrices of smaller subgraphs is not stored in computer memory.

ALGORITHM 2: *Disconnected graph clustering*

Let I and J vectors be the homonymous output vectors of Algorithm 1, holding the indices of the sparse matrix ${}^{S}A$ representing a mass difference network.

Let ${}^{I}X$ and ${}^{J}X$ be vectors of variable size holding the i-indices and j-indices of a disconnected subgraph, ${}^{t}S$ and ${}^{t-1}S$ being the sizes of that component at time t and t-1, respectively, where time is represented by the iteration counter $c \in Z$.

Let and ${}^{I}C$ and ${}^{J}C$ be two matrices of unknown size holding the i-indices and j-indices of all disconnected subgraphs of a mass difference network, where each row corresponds to a subgraph and each column to a pair of nodes i and j, respectively.

Input: vectors [I], [J] holding the indices of network sparse matrix ${}^{S}A$.

Output: index matrices [${}^{I}C$], [${}^{J}C$], holding the sparse matrix information of all disconnected subgraphs of ${}^{S}A$.

$c \leftarrow 1$

${}^{I}C \leftarrow \varnothing$

${}^{J}C \leftarrow \varnothing$

while $I \neq \varnothing$

$\quad {}^{I}X \leftarrow \varnothing$

$\quad {}^{J}X \leftarrow \varnothing$

$\quad {}^{t}S \leftarrow |{}^{I}X|$

$\quad Q \leftarrow 1$

$\quad$ while $Q = 1$

$\quad\quad \forall k \in \{1, \ldots, |I|\}, k \in \mathbb{Z}$

$\quad\quad\quad$ if $(I_k \neq 0) \wedge ({}^{I}X \neq \varnothing)$

$\quad\quad\quad\quad$ if $(I_k \notin {}^{I}X) \vee (J_k \notin {}^{I}X) \vee (I_k \notin {}^{J}X) \vee (J_k \notin {}^{J}X)$

$^{I}X \leftarrow \{^{I}X, I_k\}$

$^{J}X \leftarrow \{^{J}X, J_k\}$

$I_k \leftarrow 0$

$J_k \leftarrow 0$

end-if

end-if

if $(I_k \neq 0) \wedge (^{I}X = \varnothing)$

$^{I}X \leftarrow I_k$

$^{J}X \leftarrow J_k$

$I_k \leftarrow 0$

$J_k \leftarrow 0$

end-if

end-for

$^{t-1}S \leftarrow {}^{t}S$

$C_t \leftarrow |^{I}X|$

if $^{t}S > {}^{t-1}S$

$Q \leftarrow 1$

end-if

if $^{t}S = {}^{t-1}S$

$Q \leftarrow 0$

end-if

end-while

if $^{I}C \neq \varnothing$

$^{I}C(c, |^{I}X|) \leftarrow 0$

$^{J}C(c,|^{J}X|) \leftarrow 0$

$^{I}C(c,\{1...|^{I}X|\}) \leftarrow {}^{I}X$

$^{J}C(c,\{1...|^{J}X|\}) \leftarrow {}^{J}X$

end-if

if $^{I}C = \emptyset$

$^{I}C(1,*) \leftarrow {}^{I}X$

$^{J}C(1,*) \leftarrow {}^{J}X$

end-if

$c \leftarrow c+1$

end-while

My source code on the disconnected graph clustering algorithm was not included in this manuscript.

3.2.2 Netcalc algorithm

Once all disconnected subgraphs are extracted from the sparse matrix of the mass difference network, the network-based formula calculation algorithm can be applied on each of these graph components in order to annotate the masses represented by graph nodes. This unique annotation technique, which I call *Netcalc*, is an efficient Breadth-First Search (BFS) algorithm that uses a node of known elementary composition as a starting point and, given the known formulae of all chemical reactions on the edges, calculates by inference the compositions of all other nodes within the same graph component. The computational complexity of a typical BFS strategy is, at the worst case, linear in the number of nodes and edges. However, Netcalc's performance is further boosted via its inference algorithm which significantly reduces the number of untreated

nodes at every iteration by a Gaussian trend.

On a list of nodes whose neighbours are known, the Netcalc algorithm applies the following steps:

- Search from the top of the list until you find a mass of known composition.
- Annotate all direct neighbours of that node and mark it as treated.
- Continue the search until the end of the list.
- Repeat steps one to three until all nodes in the list are treated.

Note that in the first iteration of the algorithm the known elementary composition is provided as heuristic knowledge. My source code for the Netcalc algorithm was not included in this manuscript.

3.2.3 Netcalc-filtering

The list of elementary compositions yielded by Netcalc on an experimental mass have a rate of false annotations that can be used as an indicator of the quality of data calibration. In theory, the lower the rate of false annotations and the optimal ppm threshold used by the Netcalc algorithm, the better calibration we get. The optimal ppm threshold can be determined simply by launching several runs; starting from a very low value (such as 0.1) and gradually augmenting it while the rate of false annotations decreases. As a rule of the thumb, we can assume to have reached an optimal ppm at the run prior to the one where the rate of false annotations started increasing. Depending on the dataset, false annotations are detected by applying the filters presented in [63].

Some additional filters used by the Netcalc algorithm can be seen in table 3.1.

H/C > 3
O/C ≥ 1
H/N < 2
S/C > 3
N/C > 1
Nitrogen rule
^{13}C isotope removal
S isotope removal (^{34}S – ^{32}S)
Experimental exact mass comparison

Table 3.1: Core Netcalc filters

For the application of filters, we have developed two different Netcalc-filtering strategies:

- *Static filtering*: The filters are applied on every individual formula after the Netcalc algorithm has converged.
- *Dynamic filtering*: The filters are applied on every individual formula while the Netcalc algorithm is running.

In the case of dynamic filtering, the application of a filter is integrated inside Netcalc and once a node is annotated its formula is checked for validity. If the formula is discarded then the node is not marked as treated, meaning that there is a chance for it to receive an

acceptable annotation during another iteration. The downside of this algorithm is that, in the case of nodes representing artefacts without a possible annotation, some paths in the graph may be blocked and not accessible for annotation. I have observed that the efficiency of each strategy depends on the nature of the dataset in question.

3.2.4 Iterative Netcalc

The Netcalc algorithm uses a *starting mass* of known elementary composition as the heuristic information for inference and annotation of all graph nodes. This mass can be either provided exclusively by the programmer/user or picked out in a targeted fashion out of a list of known theoretical compounds using the lowest error margin as a criterion. The error margin of the starting mass' composition may have an important impact to the algorithm's performance. In order to overcome that issue, the basic Netcalc algorithm can be iterated over several starting points and, using dynamic filtering, maximise the number of correct annotations. The process becomes slower and more targeted in comparison to the standard heuristic Netcalc applied on a single subgraph, however, the output of correct annotations may increase significantly. This variant of Netcalc is still at an experimental stage.

3.2.5 Unsupervised network reconstruction

A strictly non-targeted scenario for network reconstruction is the one applied in [1], where no transformation list is provided to the algorithm. Instead, a local search clusters the most frequently occurring mass differences and constructs a transformation list on the fly. I re-implemented the algorithm of R. Breitling [1] and used it to mine “hidden” transformations between all disconnected subgraphs (yielded by Algorithm 2) in an attempt to unify the entire network structure into a single graph, which would allow us to annotate both newly discovered transformations as well as all disconnected subgraphs in a single run. This method is experimental and its results are to date under evaluation. My source code for hidden link mining and unsupervised network reconstruction algorithms was not included in this manuscript.

3.3 Netcalc standalone application

I developed a user-friendly Graphical User Interface, which can be compiled and used as a standalone application for mass difference network analysis (figure 3.4). The software was written in MATLAB and takes its name after the Netcalc algorithm.

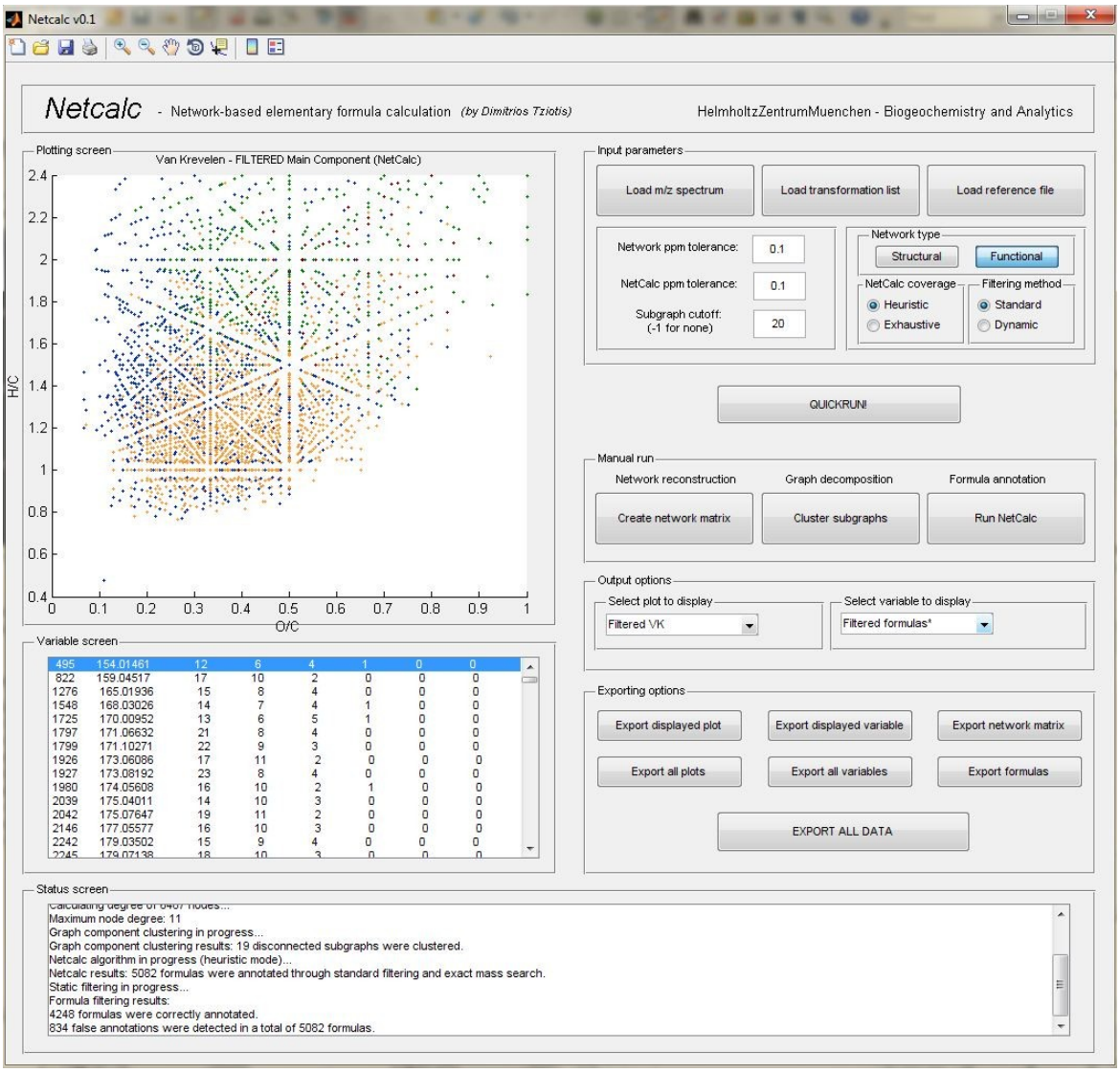

Figure 3.4: Graphical user interface of the Netcalc software showing all screens, input, and output options.

The application is divided in three output screens and four user-interface panels, described below:

The algorithm's input and parametrisation panel. The user loads a mass spectrum in text or .xls format (load m/z spectrum), a transformation list in .xls format (load transformation list), and a reference list from which Netcalc will automatically choose an annotated mass as its starting point (load reference file). The 'network ppm tolerance' field sets the ppm tolerance value used during the network reconstruction process. The 'Netcalc ppm tolerance' sets the ppm tolerance value used by the Netcalc algorithm when reading a starting mass from a reference file containing one or more theoretical compounds (in the case where there are many compounds, the one with the lowest error is chosen). The network type, *structural* or *functional*, basically determines whether a structural user-defined or the integrated functional transformation list will be used. By default, the 'structural' button is pressed, meaning that the user has to use the 'load transformation list' button and indicate a transformation file in the correct format: three columns with description, exact mass, and elemental composition. If the 'functional' button is pressed, then the algorithm will use its built-in, Kendrick-based transformation list. The Netcalc coverage option indicates whether Netcalc will be applied exhaustively on all disconnected graph components or heuristically only on the the largest component. In the former case, Netcalc is launched as many times as the number of disconnected graph components, which can be a slow process. In the default heuristic case, Netcalc is run only once on the largest disconnected network subgraph under the assumption that it contains the vast majority of nodes and all masses of interest. On heuristic mode, the 'subgraph cut-off' field defines the value which will be used as a threshold by the disconnected subgraph clustering algorithm during its searches for the largest subgraph. For example, in the case of the default value of 20, the clustering algorithm will extract only the first 20 subgraphs and will return the largest of them. While the total number of subgraphs is unpredictable, the clustering algorithm usually manages to finds the largest component in its 10 or 20 first iterations. In order for the algorithm to extract all graph components before returning the largest, the parameter needs to be set to the value of -1. The 'filtering method' sub-panel determines the filtering strategy that will be used, as described in a previous section.

The 'quickrun' button takes the current input settings and launches Algorithm 1, Algorithm 2, and filtering, a procedure that can also be done sequentially through the 'manual run' panel. In the output options, the 'select plot to display' and 'select variable to display' menu lists determine what will be displayed in the 'plotting screen' and 'variable screen' output panels, respectively. In the exporting options, all output data from the 'display screen' and 'variable screen' can be exported into .jpg and .xls format either selectively or collectively through the 'export all data' button. The 'status screen' panel at the bottom displays the status of all actions taken at any given point.

3.4 Method applications

3.4.1 Case study: Terrestrial NOM (Suwanee river)

Terrestrial NOM cover a remarkable area of the compositional space and often carry very informative additional biological signatures which are useful to assess the relative contributions of biological and biogeochemical reactions in its formation. Biosignatures in terrestrial and freshwater NOM often appear as intense mass peaks that can be analysed by MassTRIX metabolic pathway annotation, which relates groups of mass peaks to the KEGG metabolome database [64]. The decoding of the time-dependent individual NOM molecular signatures is an initial step toward a conceptual convergence of biogeochemistry and biodiversity with its organism specific metabolites and metabolic networks. Future comprehensive studies of NOM structure and environmental function will utilize the entire toolbox of organic structural spectroscopy, metabolomics, and systems biology. The advanced network analyses presented in the following section demonstrates new opportunities for a qualitative and quantitative evaluation of organic molecular complexity.

3.4.2 Case study: Structural comparison of space, plasma, and oceanic NOM

Primitive meteorites assembled in the early solar system have sampled across a huge variety of compositional, temperature and irradiation regimes which initiated enormous spatial (physical and chemical) heterogeneity on all size scales and a stupendous molecular diversity which rivals and possibly exceeds terrestrial biological complexity. Hence, extraterrestrial NOM fundamentally deviates from almost all terrestrial materials, which display more uniform histories of formation, given the temperate nature of terrestrial ecosystems.

Abiotic, biogeochemical and biological organic molecules occupy vastly different subsets of the chemical space. While the abiotic evolution of NOM often follows entropy-driven trajectories that maximize chemical diversity, carbon based life is confined to a rather restricted biologically relevant chemical space. However, an enormous *structural* diversity of molecules is assembled from a surprisingly small subset of universal precursors and the three-dimensional qualities of these molecules are critical for the sustenance of basic and higher life.

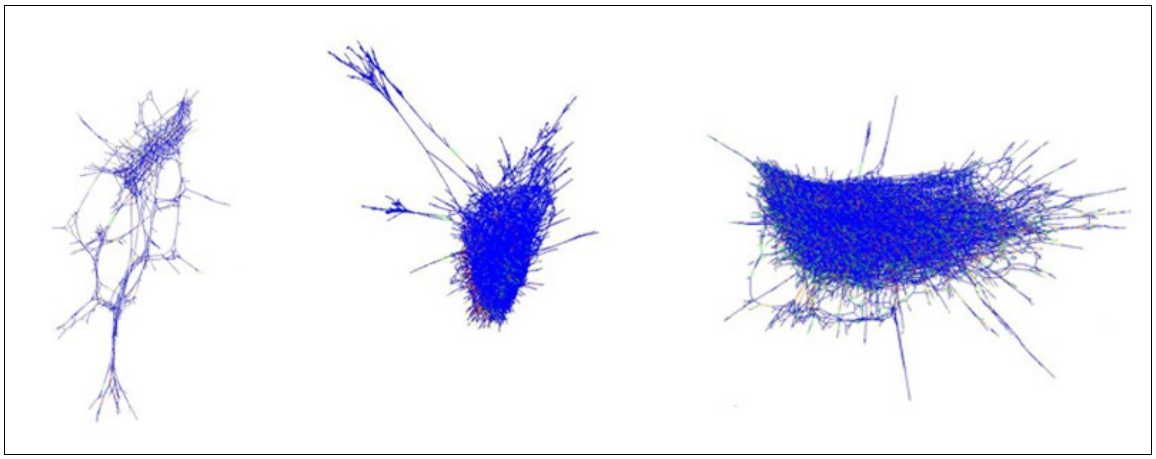

Figure 3.6: Compositional networks of three NOM samples; blood plasma (biomolecular mixture; left), riverine organic matter (biogeochemical mixture; centre), and Murchison meteorite (abiotic mixture; right).

While the emergence of biomolecules from abiotic chemistry remains unsolved, well defined relationships between these classes of complex mixtures can be established by network analysis of ultrahigh resolution mass spectra. These will faithfully depict the compositional space of molecules, which itself represents the isomer-filtered complement of the structural space. It is quantised according to the laws of chemical binding and mass differences when evaluated on basis of atoms, molecular fragments, and within nominal mass clusters.

The *functional network* approach can be used to model such a system in the form of a complex network, i.e. a diagram consisting of a set of points, together with a set of lines. In the application of this section, the visual representation of the network reflects the structural information expressed by ICR-FT-MS complex mass spectra for three samples: biological, biogeochemical, and extraterrestrial (figure 3.7). In functional networks, the set of vertices represents the obtained exact masses, and the set of edges represents a group of selected mass differences from the *Kendrick list*. Through the described model we seek to unravel the dynamic and structural space of NOM depending on various conditions, such as seasonal change, diagenetic evolution, or biogeochemical transformation.

Some properties of the individual networks can be seen in figures 3.6A, 3.6B, and 3.6C. The degree distribution plot reveals what topology a network abides to. As expected, all plots follow a *power-law distribution*, which points to the *scale-free* topology. The power-law degree distribution tells that there exist very few densely connected nodes and many sparsely connected ones. A power-law relationship is of the form:

$$y = kx^n \quad (1)$$

We can, via the properties of logarithms, convert a power-law relationship into a linear one. If we logarithmise equation (1) we get:

$$\log y = \log kx^n$$

$$\log y = \log k + \log x^n$$

$$\log y = \log k + n \log x \quad (2)$$

By defining $\log x \equiv z$ and $\log y \equiv w$, equation (2) becomes:

$$w = nz + \log k \quad (3)$$

Relationship (3) is the equation of a straight line. Therefore, a log-log plot of a power-law distribution should display a linear trend. As depicted, non-linear and linear regression yield in both cases with very high R^2 values.

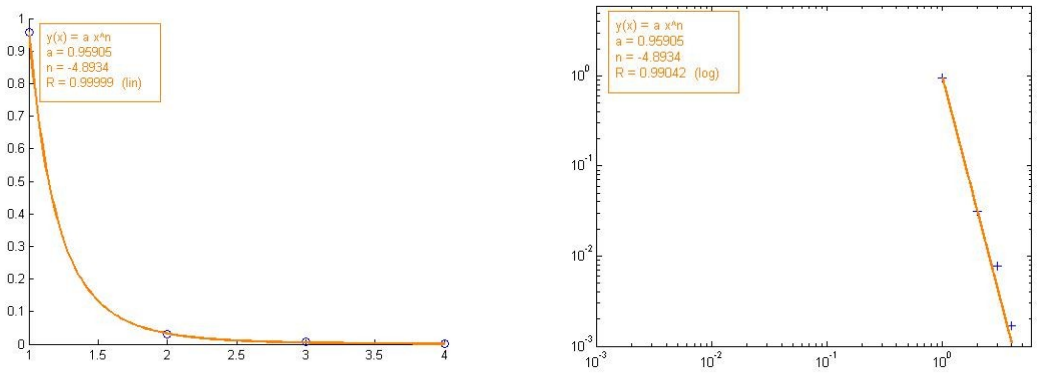

Figure 3.7A: Degree distribution of plasma tulip (biological) network (805 nodes, 1081 edges) showing the expected scale-free architecture with a power law function (left) and a linear function on its log-log counterpart (right).

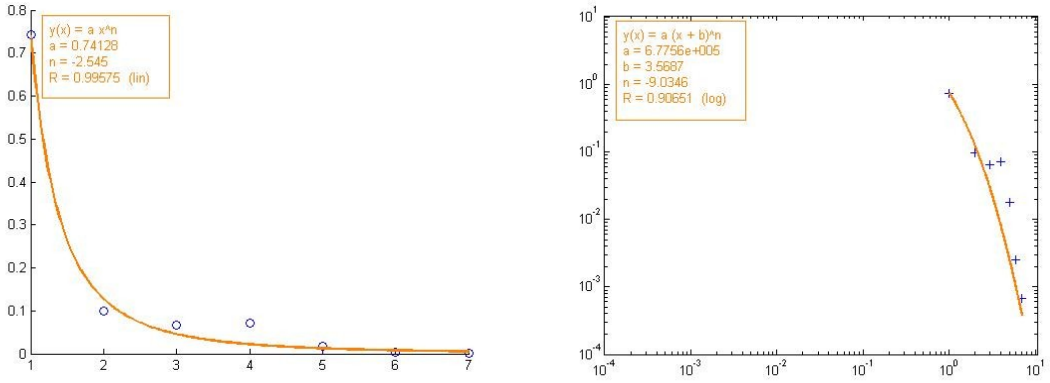

Figure 3.7B: Degree distribution of Suwannee river (aquatic NOM) network (8870 nodes, 19953 edges) showing the expected scale-free architecture with a power law function (left) and a linear function on its log-log counterpart (right).

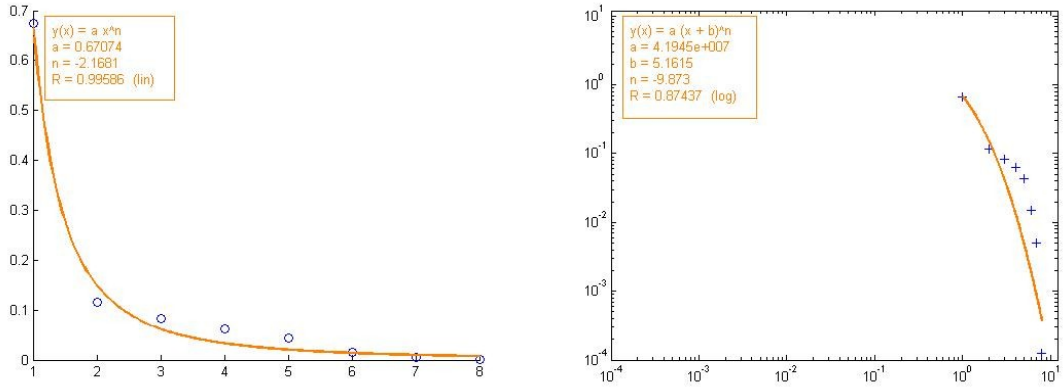

Figure 3.7C: Degree distribution of Murchison meteorite (extraterrestrial NOM) network (15933 nodes, 39616 edges) showing the expected scale-free architecture with a power law function (left) and a linear function on its log-log counterpart (right).

3.4.3 Case study: Aquatic and spatial NOM

A number of structural mass difference networks were reconstructed from aquatic and spatial ICR-FT-MS data, from Suwannee river [65], oceanic Aerosol, and meteorite samples. The reconstruction algorithm creates a network on the fly by comparing exhaustively all mass differences in a m/z spectrum to the entries of a transformation list composed by selected compounds. The resulting network matrix can be directly visualised, without the need of further analysis, in order to provide us with structural information on a given biochemical system (figures 3.8A, 3.8B, 3.8C, 3.9, 3.10). The goal of the specific study was the structural characterisation of the data in terms of elementary composition and functional groups; an approach that permits the relative comparison of samples at a structural level.

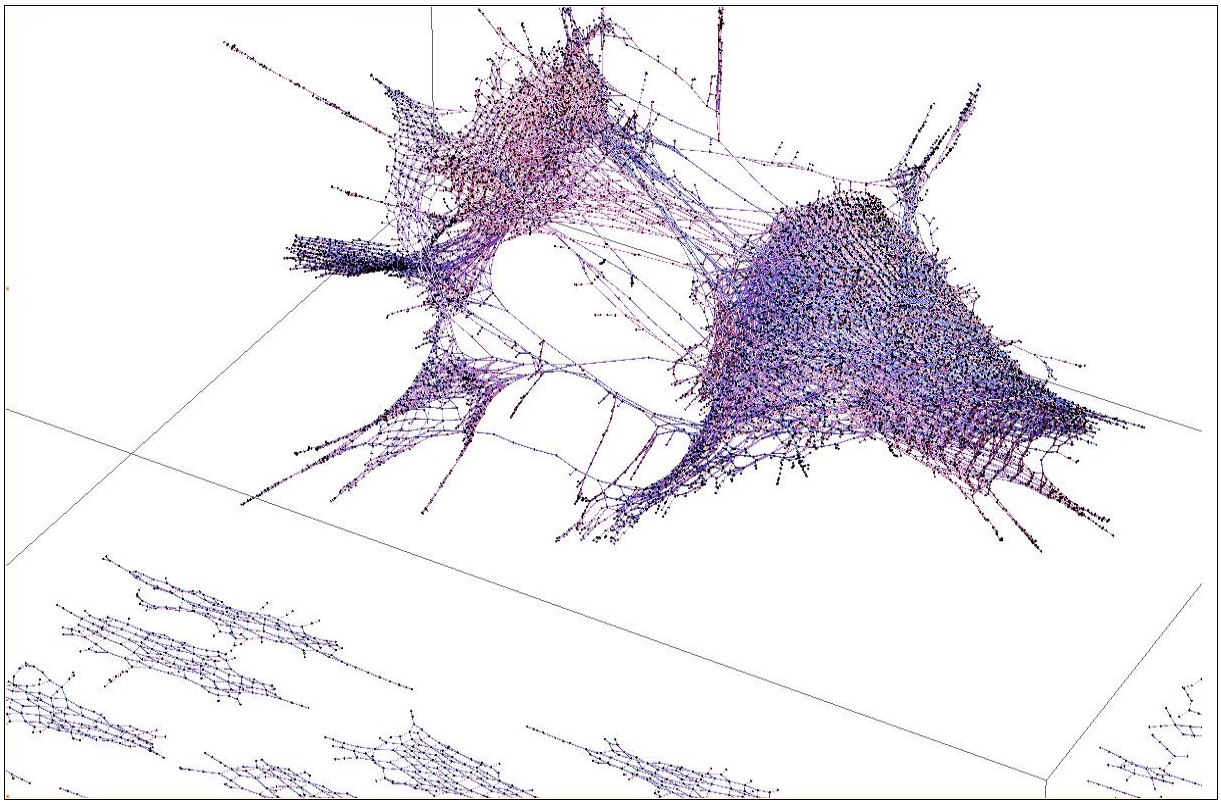

Figure 3.8A: C,H,O-based three dimensional structural network mass analysis of Suwannee River NOM. The main bulk of the network along with its disconnected subgraphs display a strong scale-free architecture. Nodes represent masses and edges represent chemical reactions between masses; colours in this network do not follow a biological meaning.

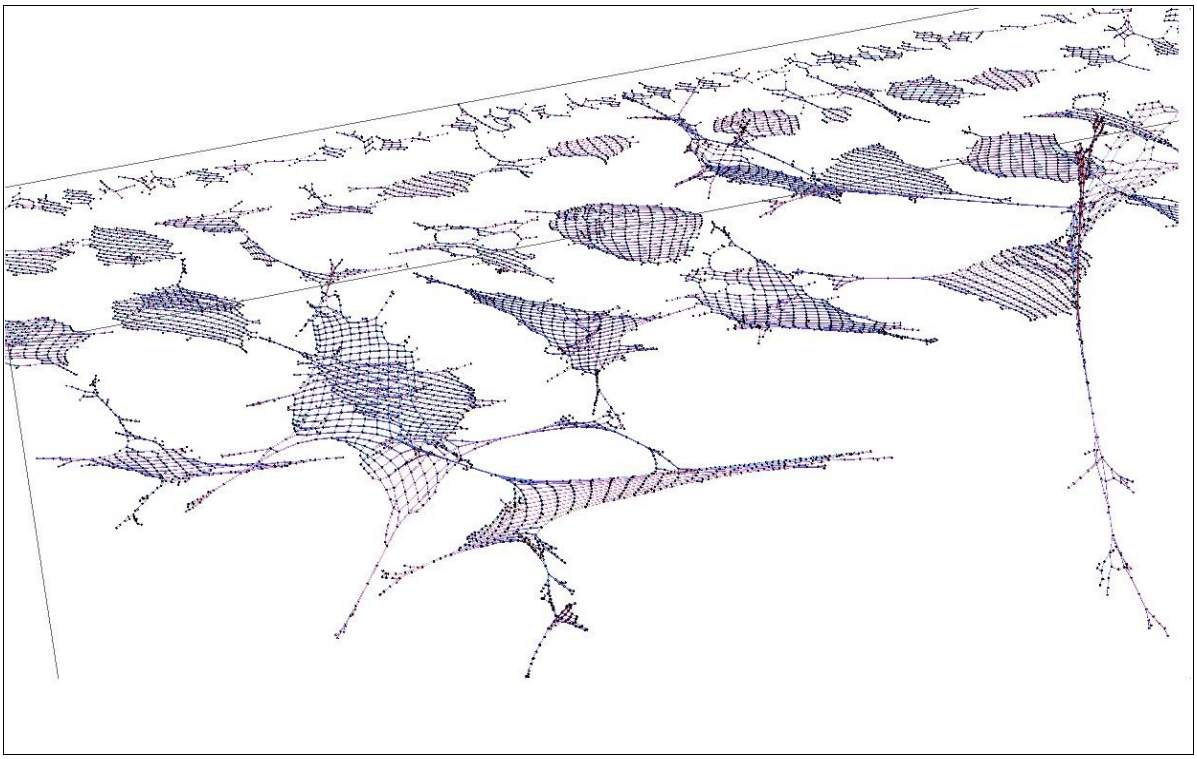

Figure 3.8B: Structural network of Suwannee river NOM (0.5 ppm). At a lower p.p.m. threshold the main bulk of the network is broken into smaller subgraphs of high modularity. Nodes represent masses and edges represent chemical reactions between masses; colours in this network do not follow a biological meaning.

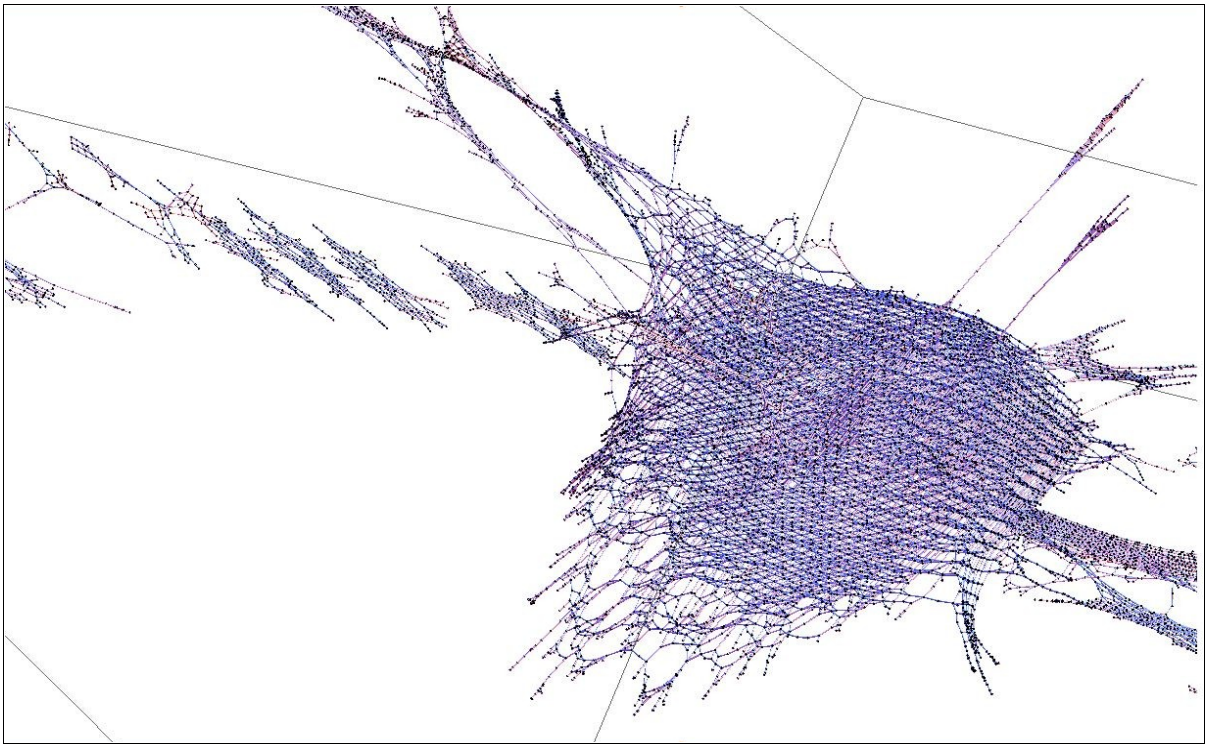

Figure 3.8C: Structural network of Suwannee river NOM (0.5 ppm). A view of the largest network subgraph revealing the strong modularity and scale-free architecture in its form. Nodes represent masses and edges represent chemical reactions between masses; colours in this network do not follow a biological meaning.

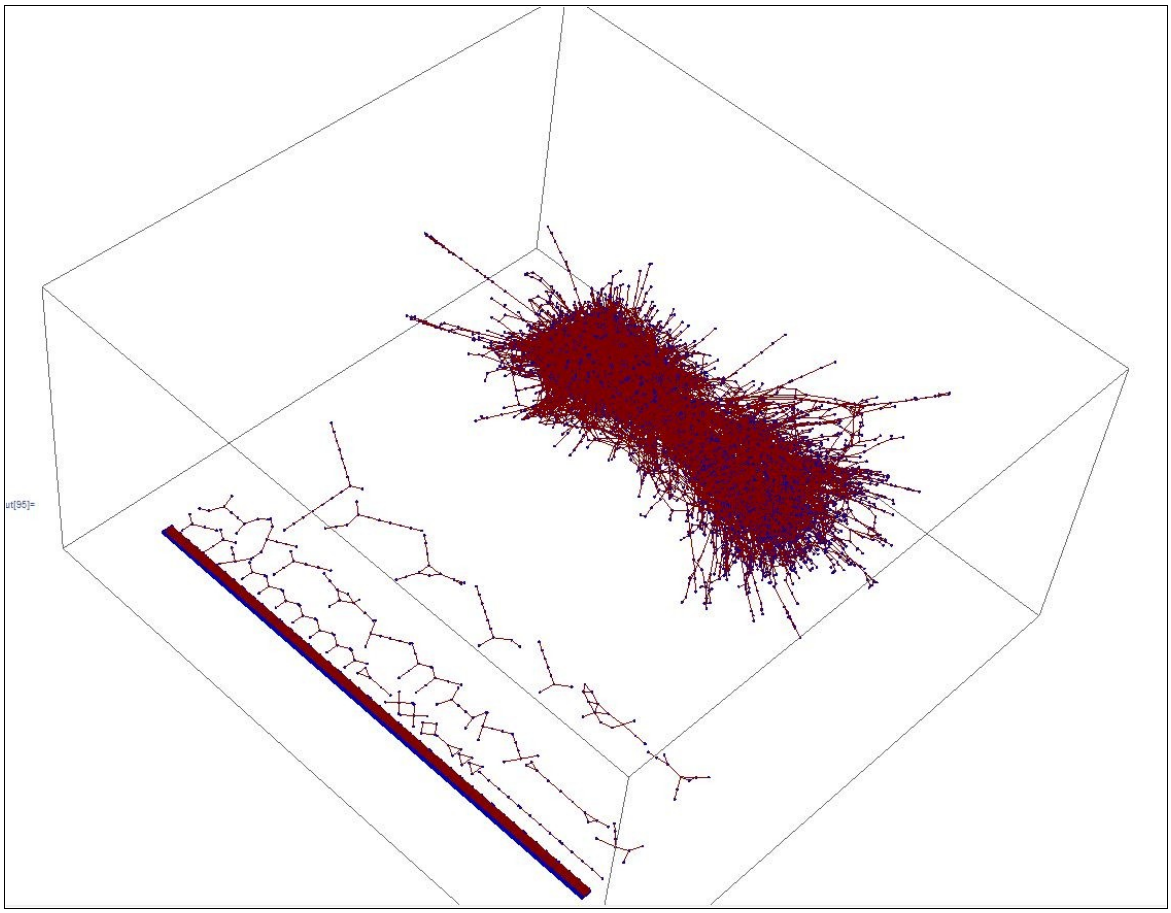

Figure 3.9: Structural network of Maribo meteorite NOM (0.1 ppm). At a minimal p.p.m. threshold the extraterrestrial NOM remains remarkably very solidly connected in a dense subgraph of relatively low modularity. Nodes represent masses and edges represent chemical reactions between masses; colours in this network do not follow a biological meaning.

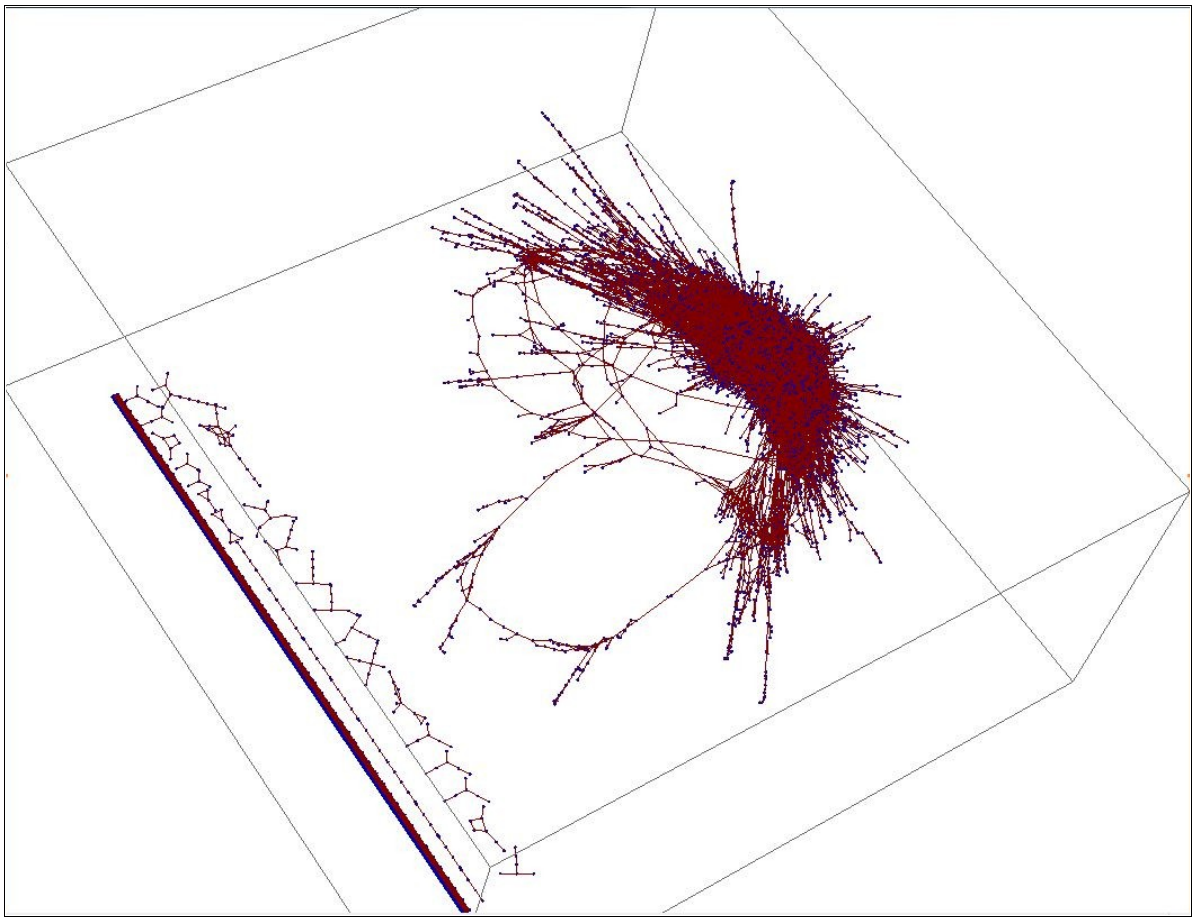

Figure 3.10: Structural network of Murchison meteorite NOM (0.1 ppm). At a minimal p.p.m. threshold the extraterrestrial NOM remains remarkably very solidly connected in a dense modular subgraph. Nodes represent masses and edges represent chemical reactions between masses; colours in this network do not follow a biological meaning.

A functional network (figure 3.11) is similarly constructed out of a m/z list, however in this case, the transformation list (that we refer to as *Kendrick list*) is composed by a fixed number of compounds which (figure 3.12), in turn, comprise a set of chosen transformation groups. The matrix corresponding to this graph is constructed especially in order to store information on the transformation group of every detected transformation. This information is reflected in the colours of edges during network visualisation, where every edge is assigned to the colour of its corresponding transformation group. An application of the method was demonstrated on the Suwanee river NOM sample (figures 3.13, 3.14).

Once reconstruction is achieved, the Netcalc method performs formula calculation. First, a clustering algorithm is applied in order to extract all disconnected graph components from the network. The idea is that, for any given extracted subgraph (which is treated as an individual connected network), knowing one single elementary formula out of all nodes/masses involved in that component should enable us to calculate all elementary formulas in the component by using the known formulas of our reaction compounds that are assigned to the graph's edges (figure 3.15). This inference procedure is achieved by applying an efficient ($O(|E|+|V|)$ Breadth-First Search algorithm on every graph component. Our “starting point” formula is chosen heuristically from a list of calibrated masses, which can be later used in its entirety to validate our results (figures 3.16A, 3.16B).

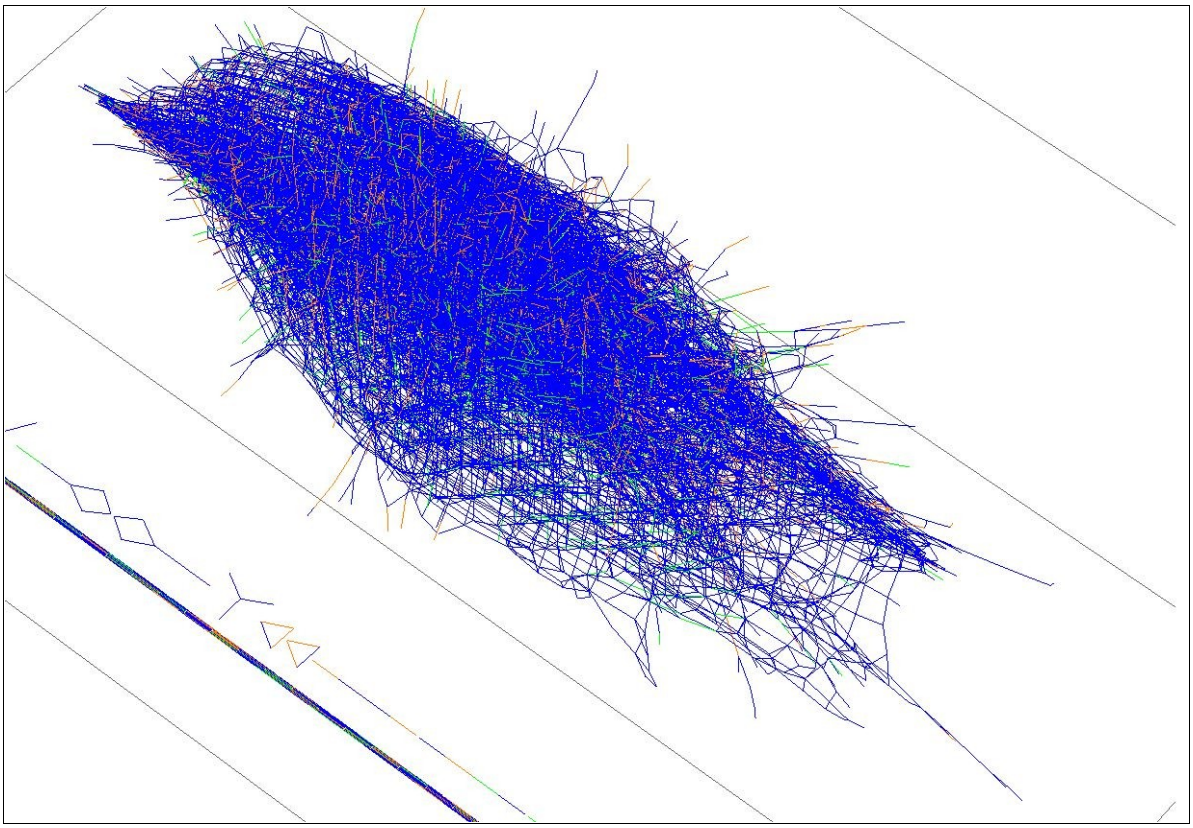

Figure 3.11: Functional network - oceanic aerosol NOM (0.5 ppm): 11092 nodes, 19953 edges, 931 subgraphs, Largest subgraph order (number of nodes): 7404, Largest subgraph size (number of edges): 16873. Nodes representing masses have been rendered invisible while edges represent chemical reactions between masses. The colours of the edges in this network represent the groups that chemical reactions belong to.

methanol (-H2O)	CH2	14.01565007
hydrogenation/dehydrogenation	H2	2.015650074
hydroxylation (-H)	OH	15.99491464
Carboxylation	CO2	43.98982928
Carbon-13	C13	1.003355
tertiary amine	N	14.00307401
secondary amine	NH	15.01089905
primary amine	NH2	16.01872408
Thiol	S	31.972071
sulfate (-H2O)	SO3	79.95681572
Phosphate	HPO3	79.9663324

Figure 3.12: The "Kendrick list" used in functional network reconstruction. The colours of chemical reactions in the list correspond to the colours of edges in compositional networks.

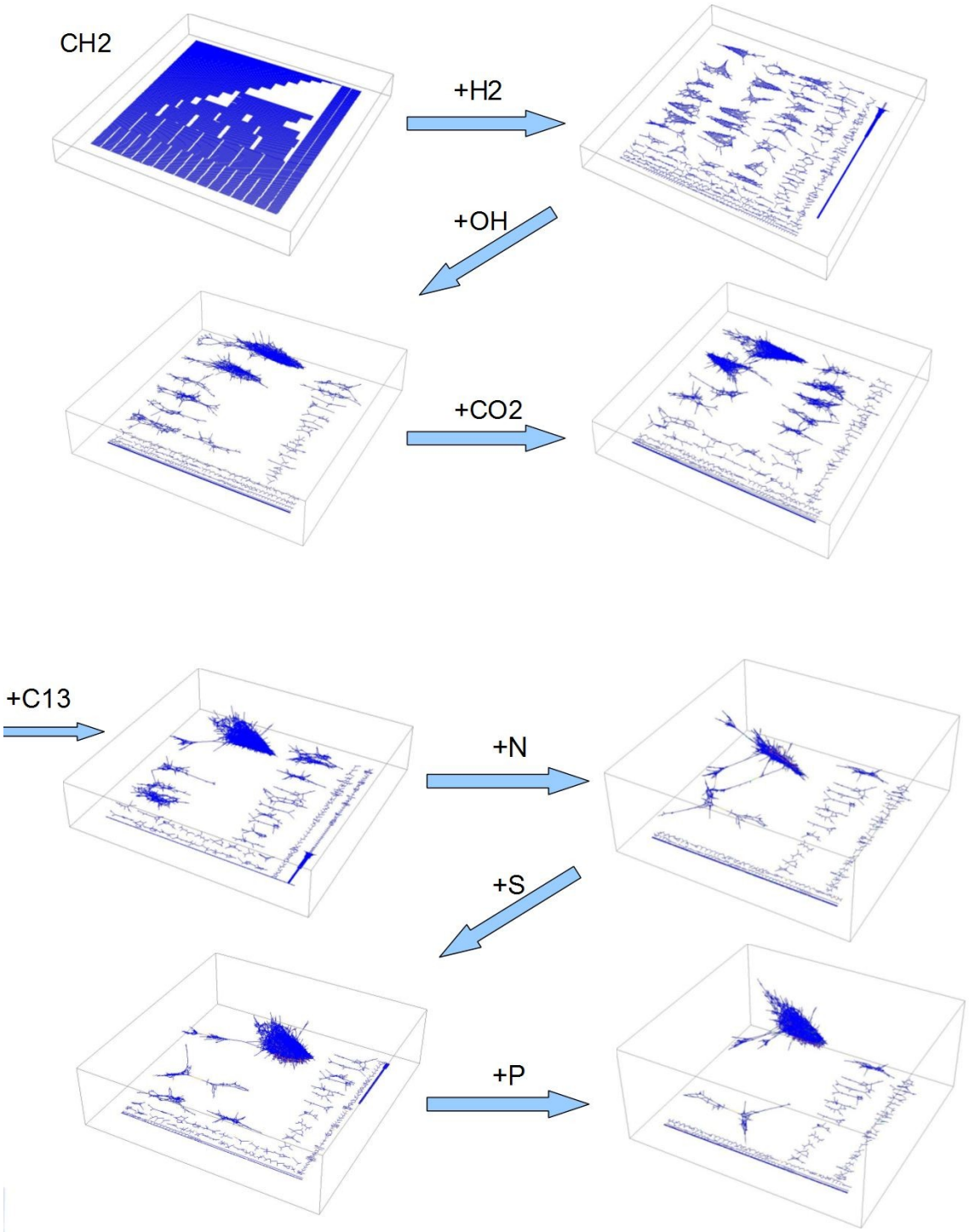

Figure 3.13: Stepwise procedure of functional network creation. Functional networks are visualized in a number of steps, each step corresponding to a type of transformation which gradually adds new nodes to the system. Nodes representing masses have been rendered invisible while edges represent chemical reactions between masses. The colours of the edges in this network represent the groups that chemical reactions belong to.

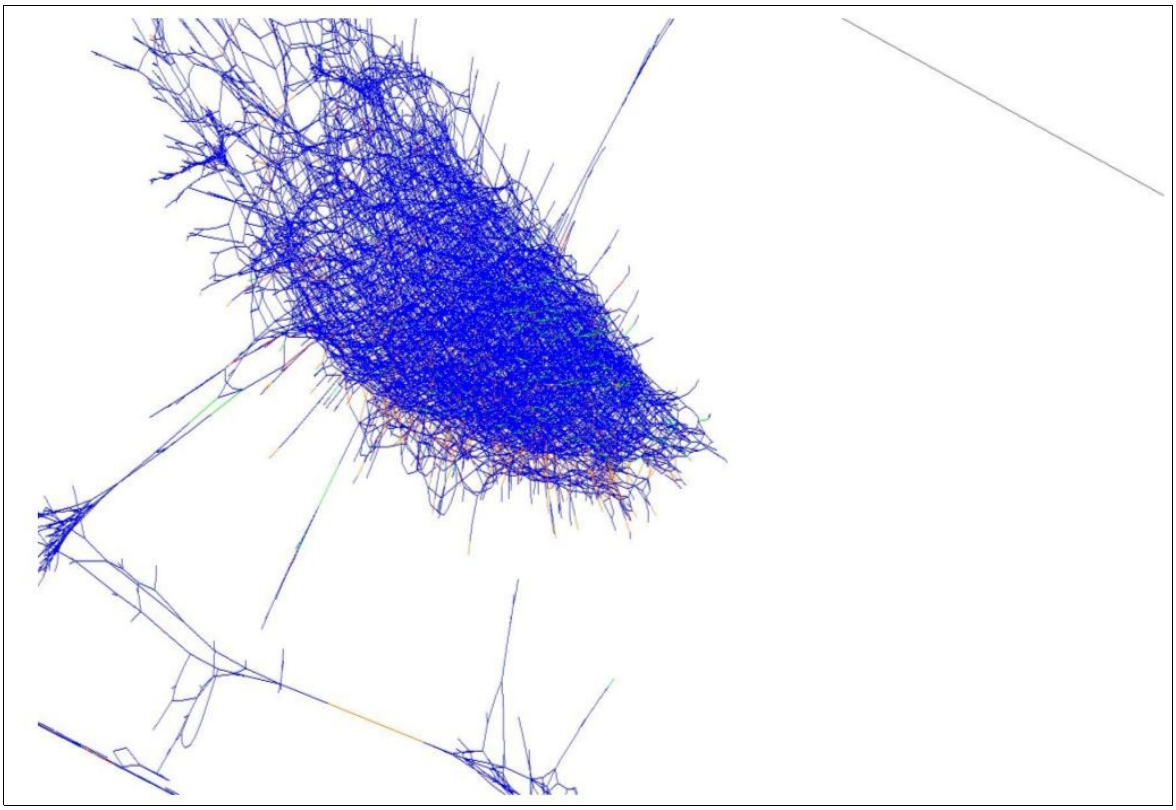

Figure 3.14 : Functional network - Suwannee river NOM (0.1 ppm). Nodes representing masses have been rendered invisible while edges represent chemical reactions between masses. The colours of the edges in this network represent the groups that chemical reactions belong to.

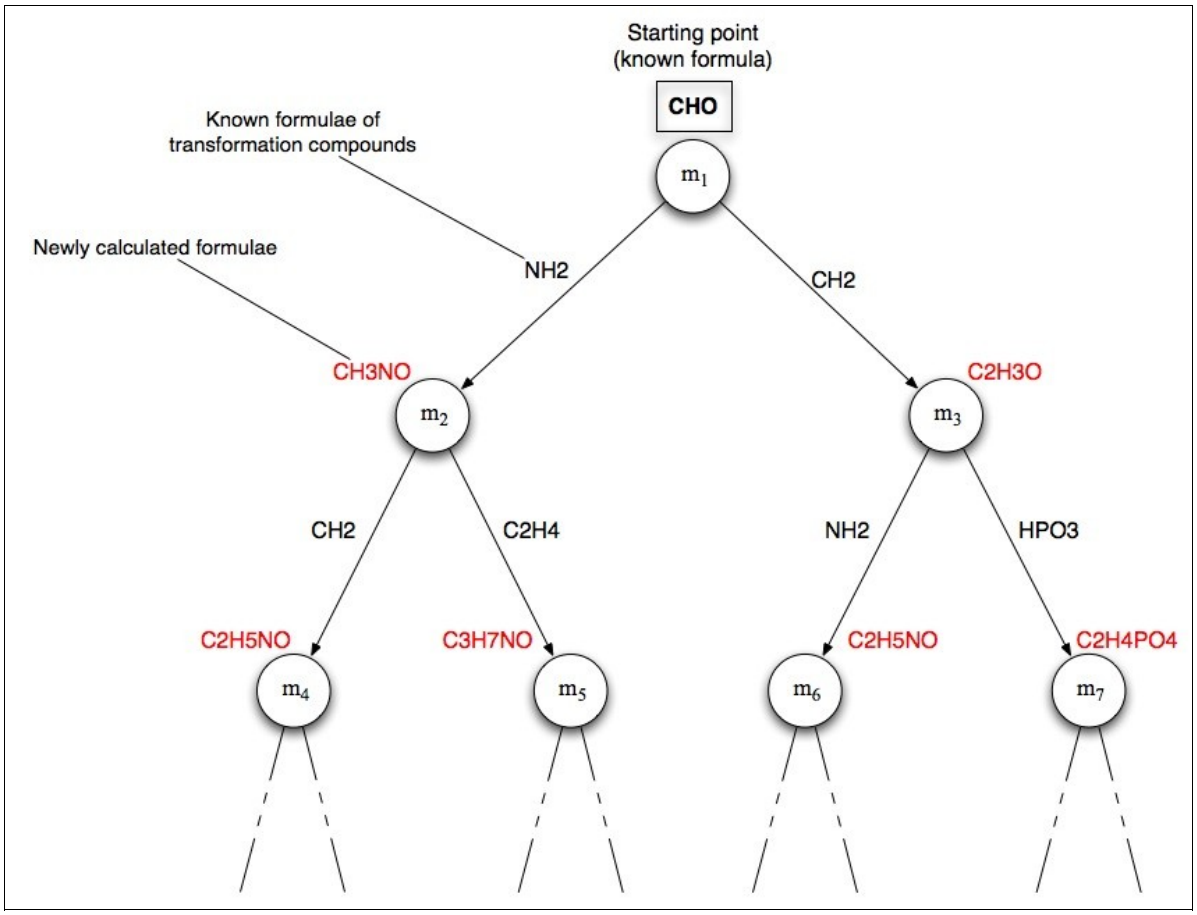

Figure 3.15: Inference of the Netcalc algorithm: A close-up view on the graphical simulation of a mass difference network, showing how nodes represent masses and edges represent chemical reactions. In this hypothetical hierarchical tree structure, the Netcalc algorithm calculates elemental compositions from top to bottom, starting with the CHO molecule whose formula is known.

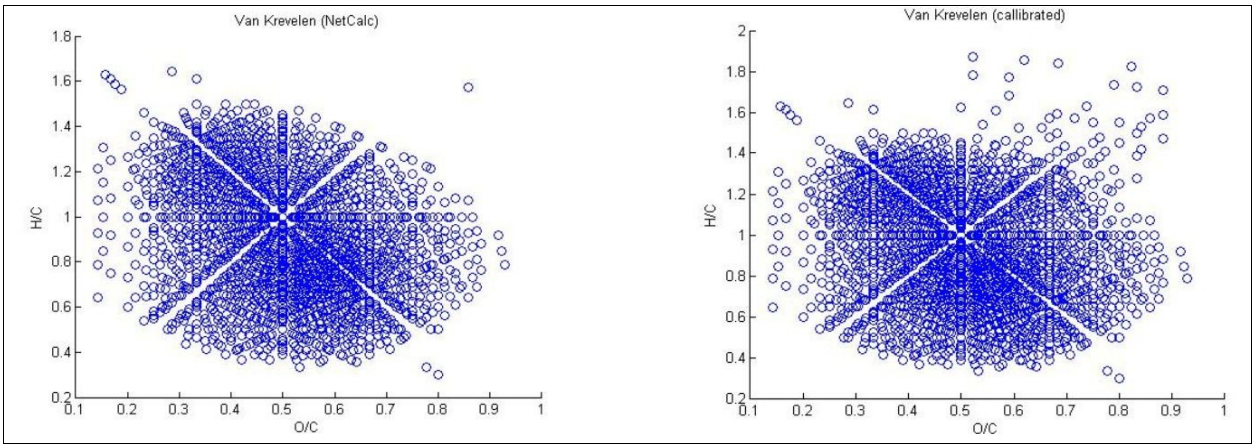

Figure 3.16A

Van Krevelen diagrams of annotated oceanic aerosol samples: Netcalc annotations (left), calibrated mass list annotations (right). A noticeable consistency between the classical method (right) and Netcalc (left) can be seen in the diagrams. Though not directly noticeable, the Netcalc's output Van Krevelen diagram is denser with significantly more masses being annotated.

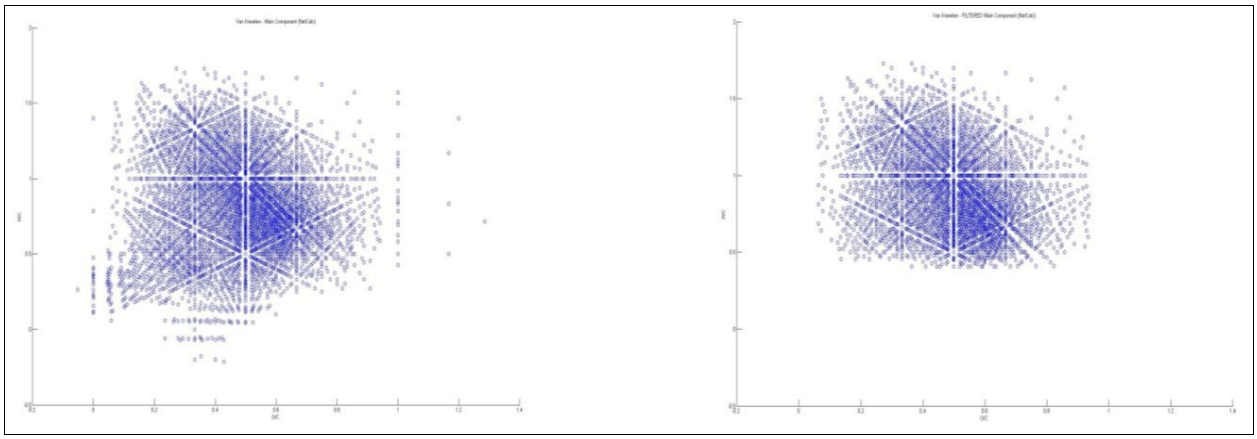

Figure 3.16B

Van Krevelen diagrams of annotated oceanic aerosol samples: Netcalc annotations - before (left) and after (right) filtering. The removal of wrongly annotated masses (filtering) can be observed in the density change from the first to the second diagram.

During the evaluation of Netcalc annotation results on Suwannee river NOM, 1935 out of the 1987 calibrated masses could be found in the largest network component. At 0.1 ppm, only 99 out of those 1935 masses in the largest component had miscalculated formulas. It was observed that those 99 formulas corresponded to very large masses, which logically implies a higher absolute error. Besides those 1935 common masses that could be easily validated, Netcalc assigned formulas to all 7404 masses of the main subgraph. The same procedure could be applied to all subgraphs (~12000 masses), which have at least one mass of known composition that could be used as a starting point.

CHAPTER IV

Unsupervised learning and cluster analysis on mass spectrometric data

In this chapter I present my study on a number of selected machine learning techniques and their application on mass spectrometric data for the purpose of unsupervised classification (clustering). All methods presented in this chapter are well studied algorithms for cluster analysis in machine learning and computational statistics, albeit with basic application on mass spectrometric data analysis (or none at all). I applied those algorithms on a selected mass-spectrometric dataset and studied their performance under varying parametrisation. My comparison of those methods points out the weak points of classical approaches and justifies the selection of tools that I used throughout this work.

4.1 Abstract

For a clustering task, we have the option between several different algorithms, each of which is associated to a similarity metric that maps data objects on a feature space. Every algorithm comes with different complexity and performance, while every similarity metric can have a different impact on an algorithm depending on the dataset in use. The performance of an algorithm on a particular dataset can be generalised for other datasets that are assumed to be similar, however, there are no means to draw a generalised conclusion on the performance of a similarity metric. The only way to evaluate the effectiveness of a certain algorithm-metric combination is to test the dataset in question using different algorithms and metrics. I tried a number of algorithm-metric tests on the

same dataset in an attempt to find the optimal combination. I first used different algorithms with a fixed metric to determine, or rather prove, that *community structure partition clustering* is superior to most mainstream algorithms. I then used the best-performing algorithm with several different similarity metrics. Part of my methods and results are presented in this section.

4.2 Metabolic microdiversity dataset

In order to test the performance of clustering algorithms, I chose a dataset whose biological grouping could be easily revealed through unsupervised means. The Metabolic microdiversity scenario from [66] provides the perfect medium for such a task, with 187 bacterial samples being involved in three classification scenarios.

Salinibacter ruber is an extremely halophilic Bacteroidetes that thrives in crystalliser ponds all around the world. In the article in question [66], two sets of 22 and 35 co-occurring *S. ruber* strains, newly isolated respectively, from 100 microliters water samples were analysed from crystalizer ponds in Santa Pola and Mallorca, located in coastal and inland Mediterranean Spain and 350 Km apart from each other. A set of old strains isolated from the same setting were included in the analysis. Overall the results show a phylogenetically very homogeneous species expressing a very diverse metabolomic pool. The high analytical mass resolution of ICR-FT/MS enabled the description of thousands of putative metabolites from which to date only few can be annotated in databases. Some metabolomic differences, mainly related to lipid metabolism and antibiotic-related compounds, provided enough specificity to delineate different clusters within the co-occurring strains. In addition, metabolomic differences were found between old and new strains isolated from the same ponds that could be related to extended exposure to laboratory conditions.

4.3 Classification modelling scenarios

An arbitrary classification scenario will model biological groups into q classes:

$$C = \{C_1, \ldots, C_k, \ldots, C_q\}$$

The experimental setup of the “microdiversity” dataset involves three classification scenarios that, in this work, I refer to as *cellular*, *regional*, and *temporal*. In the cellular scenario, samples are grouped according to the part of the cell they were extracted from. This is the most “obvious” scenario (in terms of cluster analysis) and the one that most clustering algorithms would easily treat. The regional scenario, whose biological grouping indicates the geographical origin of samples, is the most important one in the article in question [66] as well as the most difficult to detect via computational means. In fact, no algorithm other than community structure partition clustering managed to detect any patterns of this scenario. This was my main reason for choosing community structure as the clustering method of preference in this thesis. Finally, the temporal scenario has to do with the samples' chronological time of extraction and is considered the least important of all.

The cellular biological grouping instantiates the generic model with three classes for *extracellular*, *intracellular*, and *pellet*:

$$C = \{C_1, C_2, C_3\}$$

where each class holds the following index integer values of the sampled observations to be clustered:

C_1= {69...121} (extracellular)

C_2= {1, 121...187} (intracellular)

C_3= {2...68} (pellet)

The regional biological grouping involves the two classes *Mallorca* and *Santa Pola*:

$$C = \{C_1, C_2\}$$

with the index integer values of the sampled observations:

$$C_1 = \{2\text{-}28, 69\text{-}81, 122\text{-}148\}$$

$$C_2 = \{1, 29\text{-}68, 82\text{-}121, 149\text{-}187\}$$

The temporal biological grouping divides samples into *old* and *new*:

$$C = \{C_1, C_2\}$$

with the index integer values:

$$C_1 = \{1, 24\text{-}28, 64\text{-}68, 117\text{-}121, 144\text{-}148\}$$

$$C_2 = \{2\text{-}23, 29\text{-}63, 69\text{-}116, 122\text{-}143, 149\text{-}187\}$$

4.3 Comparison of clustering algorithms

The first step was to test the efficiency of the chosen clustering algorithms by comparing their performance using a fixed similarity metric. I chose the *Pearson distance* as the metric of preference and used it to create a similarity matrix, on which all clustering methods were to be applied. In this section, I describe the performance of the following algorithms: hierarchical clustering, k-means, community structure partition, and principal component analysis.

4.3.1 Hierarchical clustering

In hierarchical clustering, a similarity matrix, containing proximity information between the objects to be clustered, is initially treated by a process called *linkage*. During linkage, this proximity information is used to link pairs of objects into binary clusters. Binary clusters are subsequently re-linked together and with individual objects in order to form bigger clusters, until all objects in the dataset are linked together forming a hierarchical tree structure, which can be visualised in the form of a *dendrogram* [42]. In figure 4.1B we can see the dendrogram of the microdiversity dataset for the cellular classification scenario. The numbers along the horizontal axis are the indices of sampled objects to be clustered and the vertical upside-down U-shaped lines represent the links between objects (samples and clusters of samples). The height of the U-shape represents the distance between objects. The full dendrogram of figure 4.1A with 186 samples and 14193 data points is not very informative. For that reason, the number of objects on the horizontal axis has been scaled down to 30 by representing individual samples and smaller clusters as single objects. Just by visualising this linkage we can make some prediction on cluster creation by noticing how two main patterns are formed, bordered by the objects 30 and 2 somewhere in the middle of the horizontal axis. We can assume that the pattern on the left of object number 30 is most likely a separate cluster (probably the Pellet), however, it is not so obvious what clusters in the right-hand side pattern. Effectively, after applying the clustering process on this linkage, we find that the canonical hierarchical clustering algorithm combined with Pearson distance outputs only two clusters in the cellular scenario and, therefore, fails to deliver a satisfactory performance. Since the algorithm fails to treat the easiest scenario (cellular) efficiently, there is no reason to apply it on the rest. I performed further testing with hierarchical clustering using similarity metrics other than Pearson distance and I discovered that the algorithm does in fact cluster the cellular scenario with Spearman distance (one minus the sample Spearman's rank correlation between observations). This finding would strengthen the assumption that the efficiency of a metric varies arbitrarily over different datasets. The full and scaled dendrogram of hierarchical clustering using Spearman distance are illustrated in figures 4.1C and 4.1D,

respectively. A clustergram (cluster heatmap) of the algorithm's output stands witness to these conclusions by illustrating clearly three main patterns forming along the samples on the horizontal axis (in red colour, figure 4.2). Only 4 out of 186 samples were misclassified by this method on the cellular scenario. Nonetheless, besides those primary cellular clusters, the dendrogram and clustergram of hierarchical clustering offer no further traces of visual patterns that could be linked to the other biological scenarios, and therefore its efficient result are restricted to reflect only the least useful biological grouping (the cellular scenario).

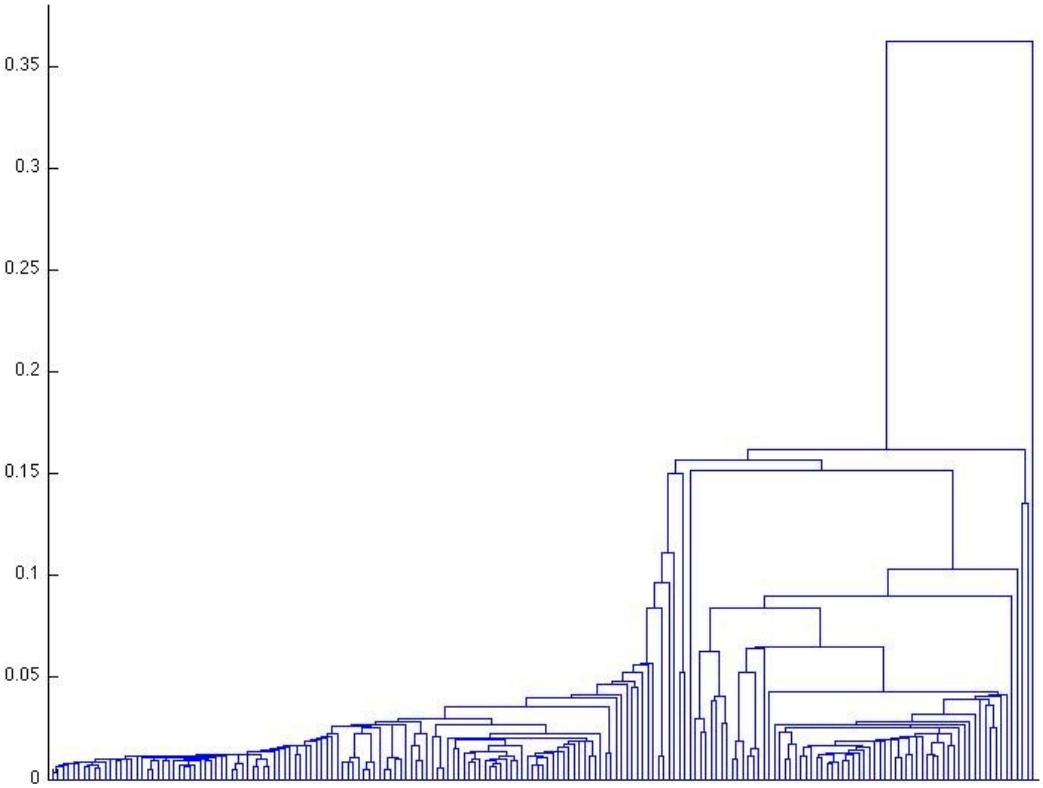

Figure 4.1A: Complete hierarchical clustering dendrogram using Pearson correlation. Numbers along the horizontal axis are the indices of sampled objects while the vertical upside-down U-shaped lines reflect the linkage between objects. The height of the U-shapes represents the distance between linked objects.

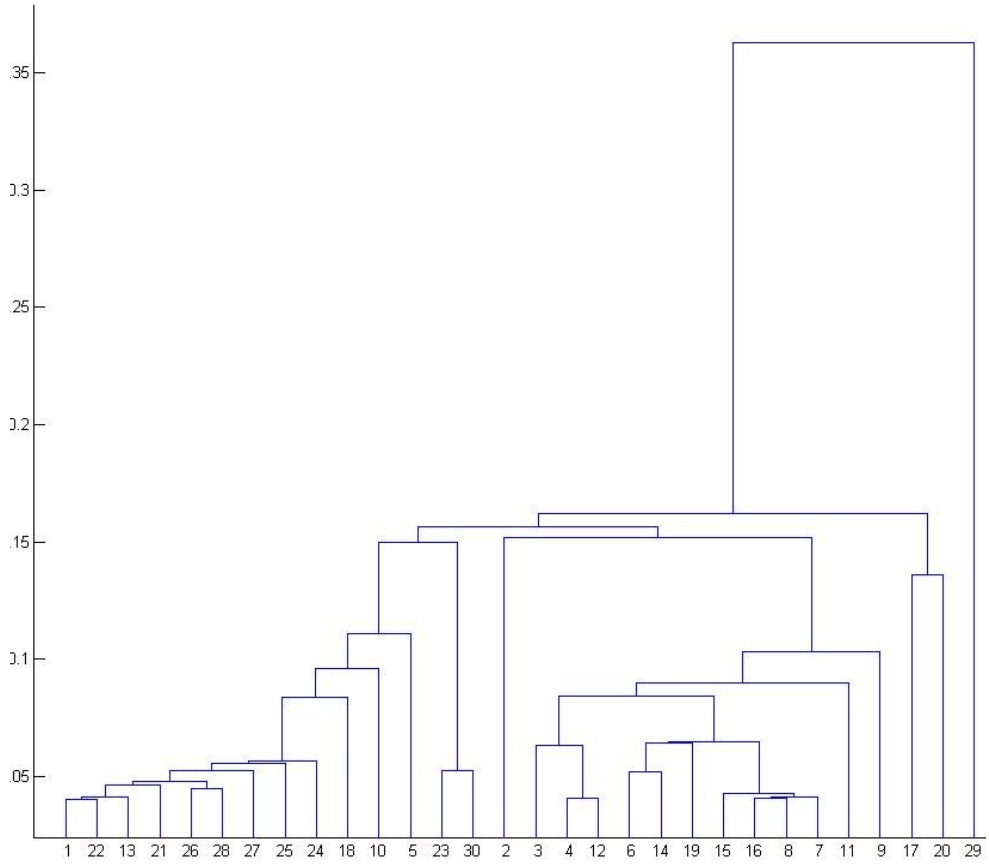

Figure 4.1B: Trimmed hierarchical clustering dendrogram using Pearson correlation. Numbers along the horizontal axis are the indices of sampled objects while the vertical upside-down U-shaped lines reflect the linkage between objects. The height of the U-shapes represents the distance between linked objects. The trimmed dendrogram decreases the number of objects for better readability.

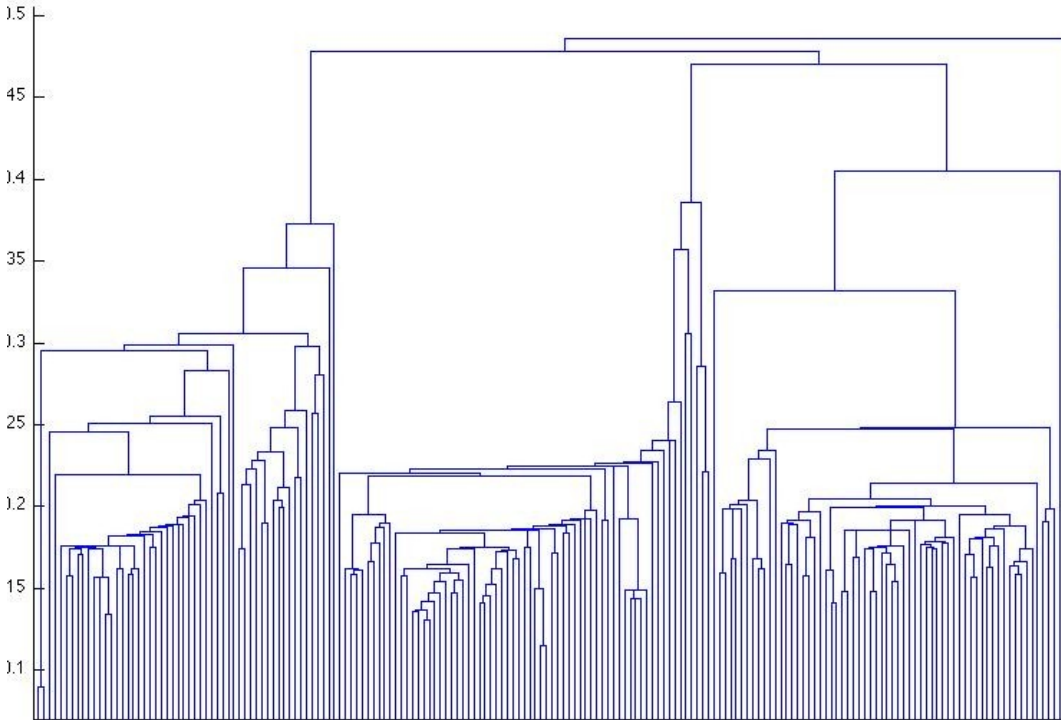

Figure 4.1C: Complete hierarchical clustering dendrogram using Spearman's rank correlation. Numbers along the horizontal axis are the indices of sampled objects while the vertical upside-down U-shaped lines reflect the linkage between objects. The height of the U-shapes represents the distance between linked objects.

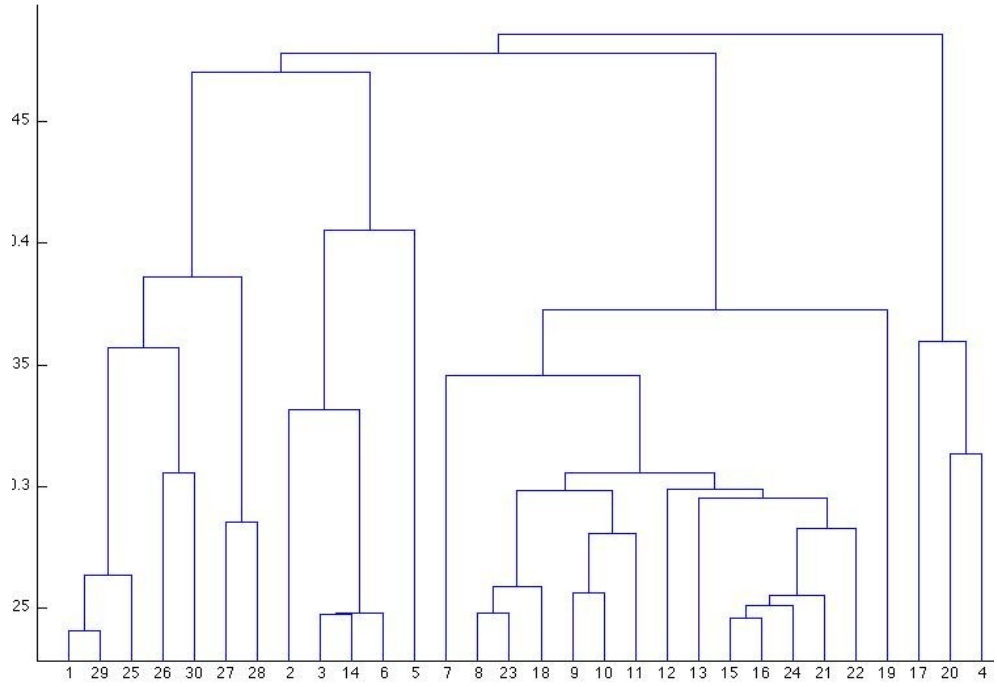

Figure 4.1D: Trimmed hierarchical clustering dendrogram using Spearman's rank correlation. Numbers along the horizontal axis are the indices of sampled objects while the vertical upside-down U-shaped lines reflect the linkage between objects. The height of the U-shapes represents the distance between linked objects. The trimmed dendrogram decreases the number of objects for better readability.

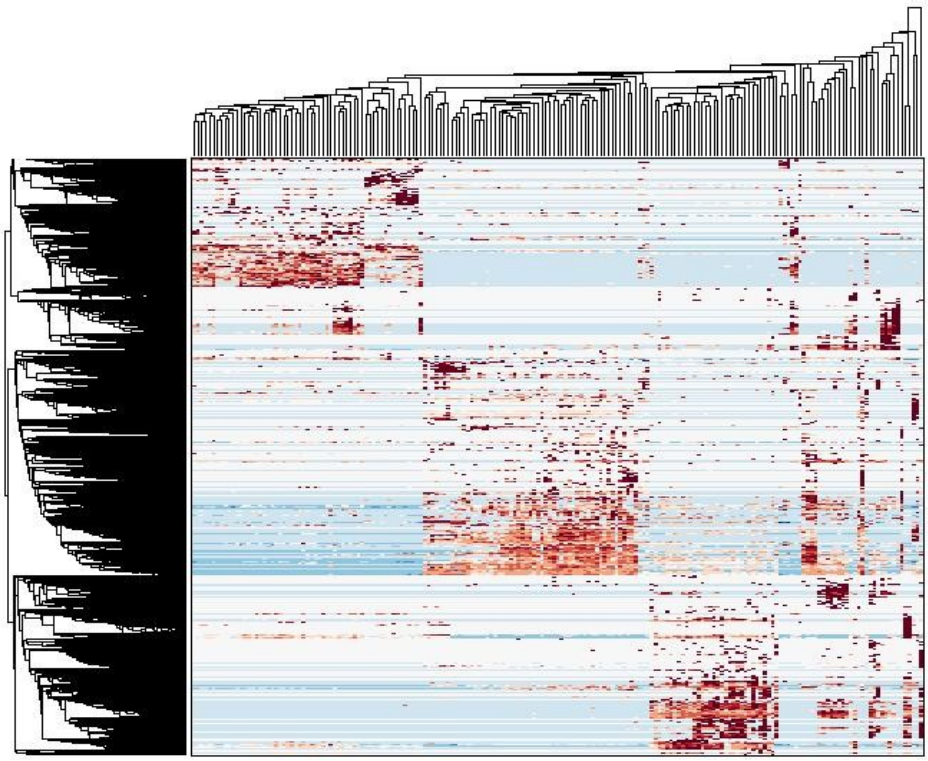

Figure 4.2: Hierarchical clustering clustergram Spearman's rank correlation. Red coloured dots represent closely linked objects that form dense clusters. Three main clusters of predominantly red colour can be observed in the top left, middle centre, and bottom right of the heatmap. Hierarchical clustering can only identify the three cellular groups.

4.3.2 k-means clustering

The canonical version of the k-means algorithm requires that the number of clusters *k* are provided as an input parameter. Using the Pearson distance for proximity, the method was applied on the microdiversity dataset with *k*=3 and managed to cluster all but 8 samples correctly, out of the total 186. Out of these 8 misclassified samples, it should be noted that numbers 42 and 68 had been also misclassified by hierarchical clustering. There is not standard visualisation method for depicting the output of the k-means algorithm for multi-dimensional data. I did, however, try using 2-dimensional plots between the proximity information of two data points corresponding to molecular masses of significantly different magnitudes. The information reflected in these plots is, of course, only approximative and meant only for the sake of demonstration, albeit not suggested as an efficient method for visualising the output of k-means. Figure 4.3A illustrates this 2-dimensional plot, in which the three clusters of samples detected by k-mean are marked with symbols 'o', 'x', and '+'. I added some colouring for the classes of the cellular scenario in order to see how they coincide with the clusters of k-means. The result of this process can be seen in figure 4.3B, where the same symbols represent the k-means clusters and the colours red, purple, and blue represent the biological groups pellet, intracellular, and extracellular, respectively. In figure 4.3C I have isolated the 8 samples that have been misclassified. In figures 4.3D and 4.3E I used two different colouring schemes for the two remaining scenarios, regional and temporal. The purpose of applying colours that correspond to the other two classification scenarios is to observe whether some sub-patterns are formed along the clustered visual output. We already know that the detected clusters correspond to the classes of the cellular scenario, yet we are trying to make sure that there are no other less obvious patterns of biological significance. The result is negative and, as it can be seen in these plots, the colours are distributed randomly in respect to the k-means clusters.

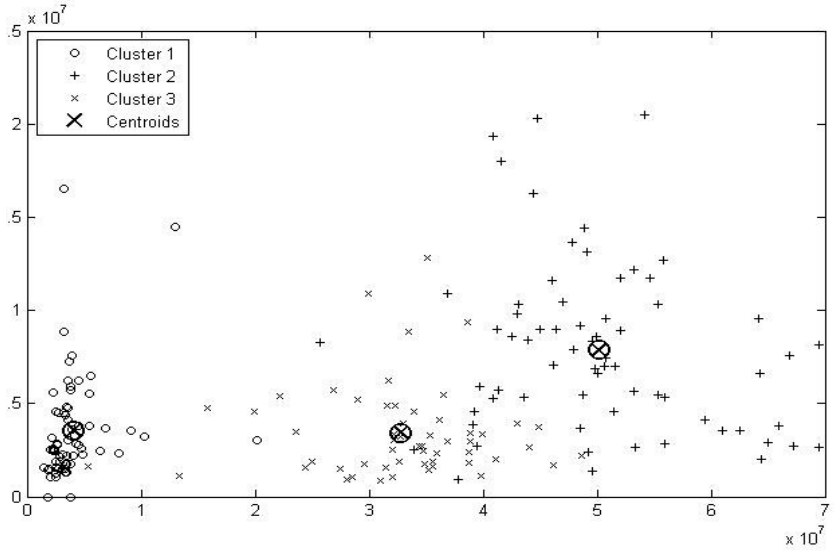

Figure 4.3A: k-means pseudo-plot - two selected molecular masses were chosen to reflect the algorithm's output. The three clusters of samples detected by k-mean are marked with symbols 'o', 'x', and '+'. The monochrome colour of the graph does not have a significance.

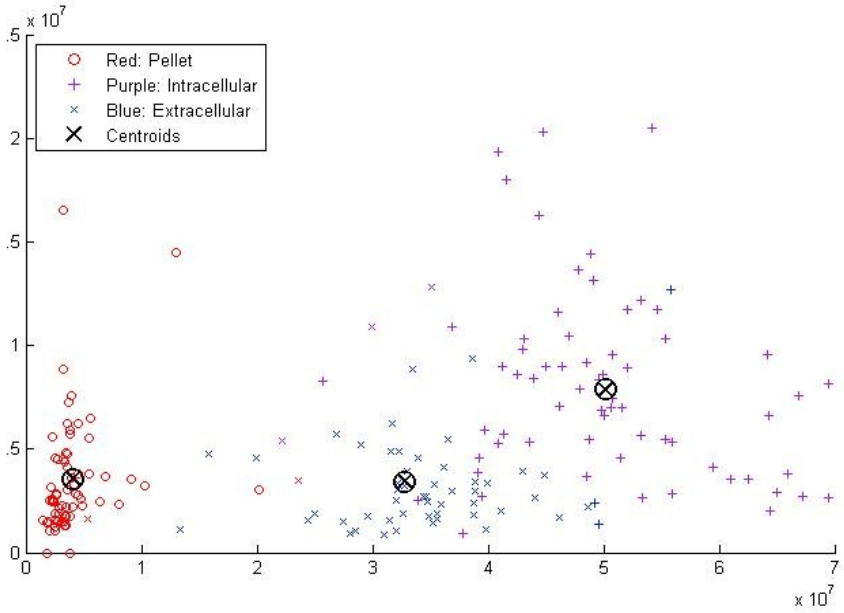

Figure 4.3B: k-means pseudo-plot – two selected molecular masses were chosen to reflect the algorithm's output. The three clusters of samples detected by k-mean are marked with symbols 'o', 'x', and '+'. Colours were added for cellular classes (blue - extracellular, purple - intracellular, red - pellet) in order to see how they coincide with the clusters of k-means.

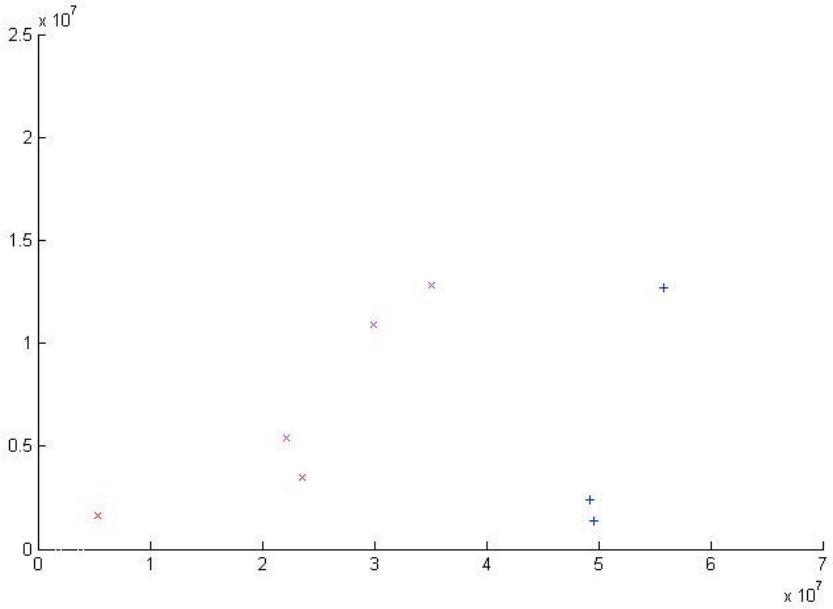

Figure 4.3C: Falsely clustered data points in k-means pseudo-plot. The three clusters of samples detected by k-mean are marked with symbols 'o', 'x', and '+', with only misclassified objects appearing in the diagram. Colours were added for cellular classes (blue - extracellular, purple - intracellular, red - pellet)) in order to see how they coincide with the clusters of k-means.

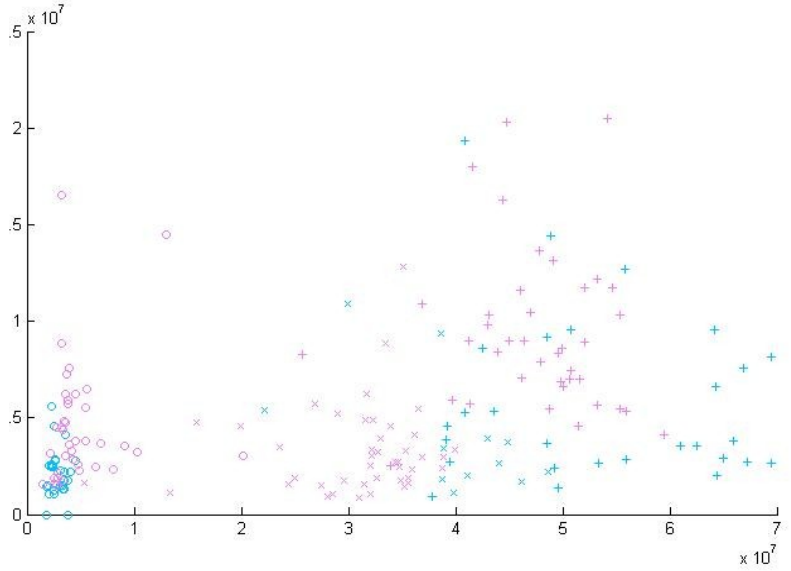

Figure 4.3D: k-means pseudo-plot – two selected molecular masses were chosen to reflect the algorithm's output. The three clusters of samples detected by k-mean are marked with symbols 'o', 'x', and '+'. Colours were added for regional classes (cyan - Mallorca, magenta - Santa Pola) in order to see how they coincide with the clusters of k-means.

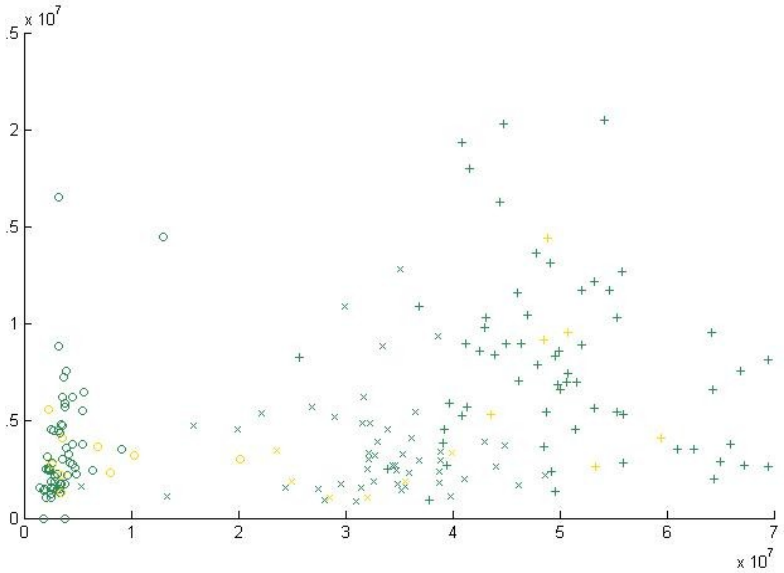

Figure 4.3E: k-means pseudo-plot – two selected molecular masses were chosen to reflect the algorithm's output. The three clusters of samples detected by k-mean are marked with symbols 'o', 'x', and '+'. Colours were added for temporal classes (yellow - old, green - new) in order to see how they coincide with the clusters of k-means.

4.3.3 Principal Component Analysis

The output of principal component analysis is a two-dimensional visualisation of the two principal components that contain most of the data's information. Figure 4.4A depicts the output plot of PCA applied on the microdiversity dataset. Unlike the rest of the algorithms we tried out so far, PCA offers no fixed sets of clusters in its output, and pattern observation on the two-dimensional plot relies largely on expert judgement. The x and y axes crossing the origin of the plot are used as the separators of clusters, which would imply that there is only a maximum of four clusters that can be determined in silico, one for every quadrant. This rule of the thumb, however, does not provide any useful results on any microdiversity classification scenario. If we add the colours of the cellular scenario to the samples of a PCA plot we see that the lines x=0 and y=0 cut biological groups down in the middle, therefore, instead of clusters we obtain only patterns (figure 4.4B). In the regional and temporal scenarios of figures 4.4C and 4.4D, respectively, the colours appear very mixed with hardly any patterns forming.

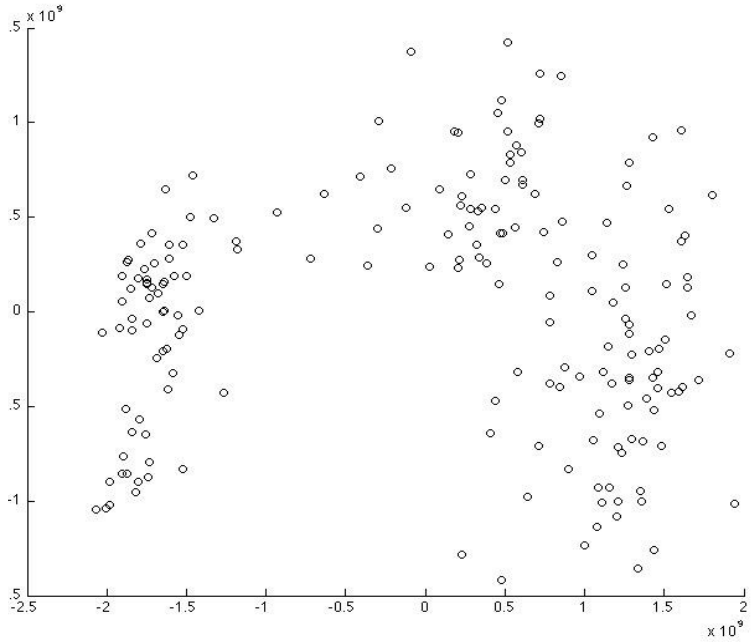

Figure 4.4A: PCA – scatter plot of the first and second principal components of the Microdiversity dataset. Without adding any colours to known patterns we can vaguely observe three to five centres of densely concentrated dots, concluding that no cluster can be detected with certainty.

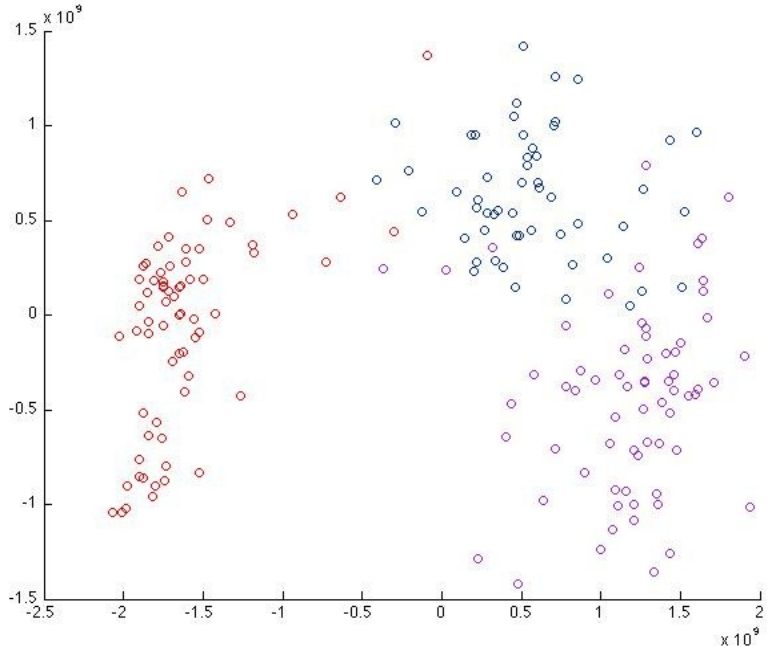

Figure 4.4B: PCA – scatter plot of first and second principal components of the Microdiversity dataset with cellular colouring (blue - extracellular, purple - intracellular, red – pellet). After adding colours for the cellular biological groups, we see that the patterns of the most densely concentrated objects correspond vaguely to those groups. The diagnostic capacity of PCA on the cellular group is weak.

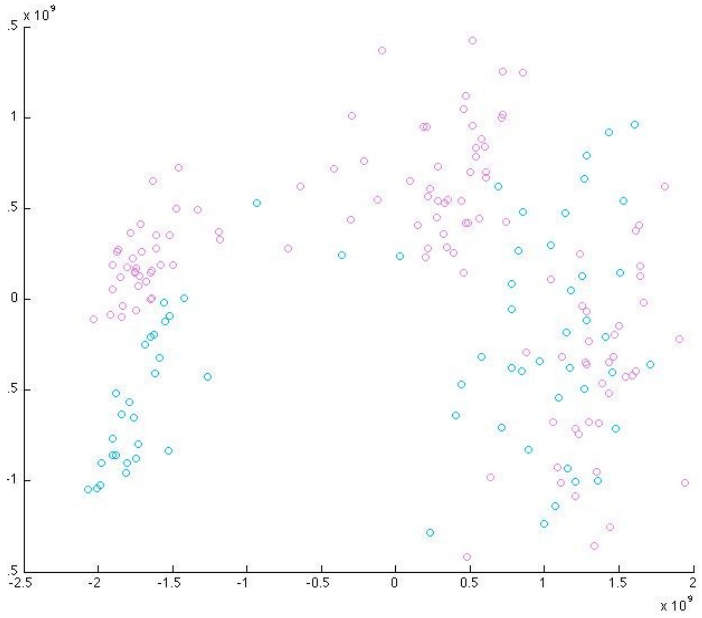

Figure 4.4C: PCA – scatter plot of first and second principal components of the Microdiversity dataset with regional colouring (cyan - Mallorca, magenta - Santa Pola). After adding colours for the regional biological groups, we see that almost no pattern can be observed. The diagnostic capacity of PCA on the regional group is almost non-existent.

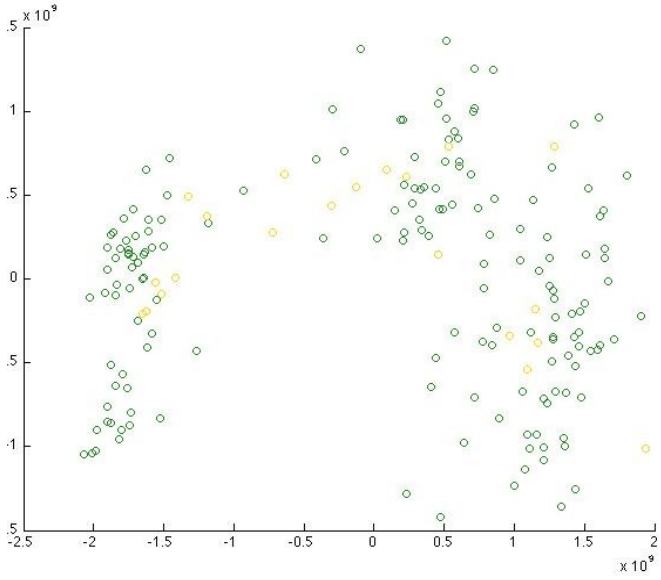

Figure 4.4D: PCA – scatter plot of first and second principal components of the Microdiversity dataset with temporal colouring (yellow - old, green – new). After adding colours for the temporal biological groups, we see that no pattern can be observed whatsoever. The diagnostic capacity of PCA on the temporal group is non-existent.

4.3.4 Self-Organizing Maps

As part of my work on Artificial Neural Networks, I tested the performance of self organizing maps (SOM); a clustering technique that differs significantly to the rest of the algorithms used in this section. A SOM is a type of artificial neural network whose nodes (neurons) are associated to a weight vector (equal in dimension to the input data vectors) and a position in the map space. The SOM's principal difference to the other clustering methods is that it does not build up a similarity matrix to use as a starting point for learning. Instead, a SOM applies a similarity metric on the fly during the training stage, via which it calculates heuristically the distances between its neurons. It produces a two-dimensional representation of the input space of the training samples that is called a 'map', where topological closeness derives from input similarity. Figure 4.5A illustrates such a map obtained by the algorithm's application on the Microdiversity dataset. In a way similar to a heatmap, neurons are represented on the horizontal and vertical axes while pairwise distances are reflected through colours in a third dimension. In this case, the size of the u-matrix is 30×30 neurons and the similarity metric used in training is the Euclidean distance. The three cellular classes are distinctively formed as patterns in the U-matrix. One of the downsides of the SOM method is that clusters are not provided as discrete sets of objects but have to be manually detected within the map's patterns (as in the case of PCA); a task which is not always as obvious as in this dataset (figure 4.5A). There are various ways to discretise SOM's clustering output but none of them is part of the canonical algorithm. In figure 4.5B, three clusters are clearly distinguishable as blue patterns in the raw U-matrix, allowing us more or less to assume where the cellular classes are. A flooding algorithm is applied on the default map in order to make these clusters more visible. Figure 4.5B depicts the output of the flooding algorithm with neurons (horizontally) versus intensity values (vertically). This line can be thought of as the “depth” of the map, i.e. its coloured third dimension, and every cluster is represented by a large depth change in the line of the plot. The resulting improved U-matrix is shown in figure 4.5C. Since data points in a U-matrix do not directly correspond to clustered objects, it is not as easy to visualise more than one classification scenarios in the same dataset. Therefore, in this case we focus only on the cellular clusters and study the

patterns that are formed between them. It would be possible to make the numerical indices of samples appear on the map (in order to point out more clearly object association to corresponding clusters) but I chose to leave this representation out since my study of SOM does not focus on clustering effectiveness.

SOMs generally come with powerful classification potential, the ability to cluster very large datasets with fairly good computational performance. In my opinion, however, the main advantage of SOM is its capability to visualise the relations of individual input variables to the detected clusters. In the map of (figure 4.5D) we see how the input vector of mass number #1000 (chosen arbitrarily) is expressed in the clusters discovered during the training process (blue and red being very low and high distance values, respectively). This is a property which makes SOM a powerful tool for the detection of over- and under- representation of discriminative features.

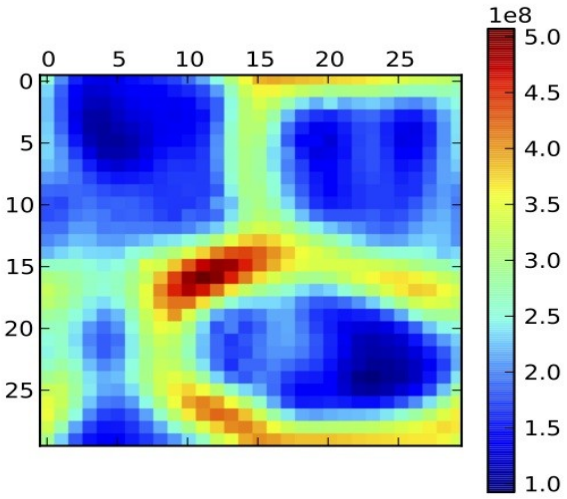

Figure 4.5A: U-matrix produced by applying a SOM (self-organizing map) on the Microdiversity dataset. The axes are made up of 30×30 neurons and the colours are derived by pairwise distances between them. Three data clusters are formed as blue coloured patterns on the map, most likely corresponding to the cellular classes.

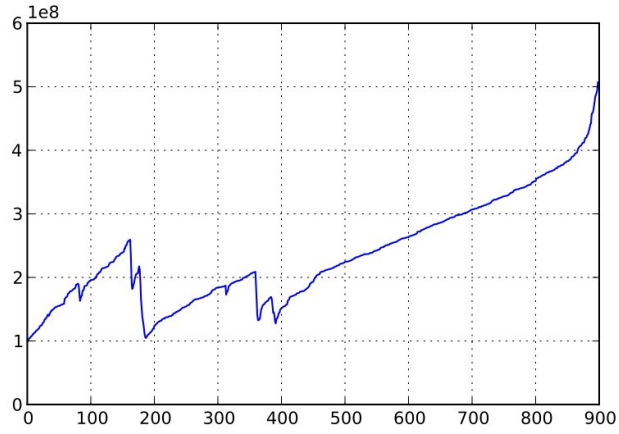

Figure 4.5B: Plot of the 'flooding' algorithm showing the depth of the map, i.e. the coloured third dimension of the U-matrix. A number of 30×30 (900) neurons are plotted horizontally against their distance variations vertically. The three clusters appear where depth changes take place, notably between neurons 100-200 and 300-400.

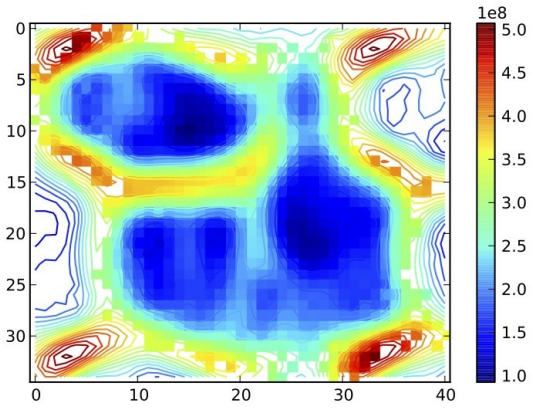

Figure 4.5C: U-matrix produced by applying SOM (self-organizing map) and 'flooding' on the Microdiversity dataset. The axes are made up of 30×30 neurons and the colours are derived by pairwise distances between them. Three data clusters are formed as blue coloured patterns on the map, most likely corresponding to the cellular classes. The flooding algorithm has added some white regions to the map, filtering out less important information and making the cluster patterns appear more intense.

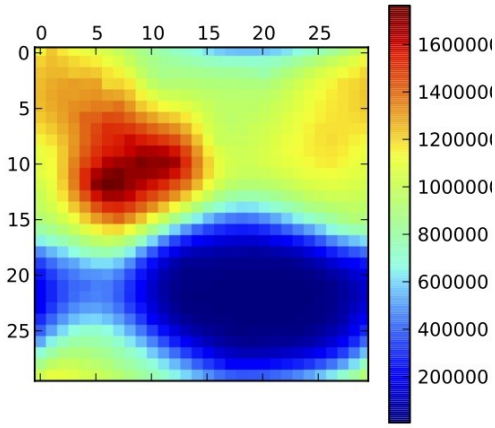

Figure 4.5D: U-matrix produced by applying a SOM (self-organizing map) on the Microdiversity dataset and visualised for input vector of index #1000. The axes are made up of 30×30 neurons and the colours are derived by pairwise distances between them. In this map we observe how mass #1000 relates to the three data clusters formed as blue coloured patterns on the full map of figure 4.5C. The regions of red and blue reveal the distance of this mass to the neurons of those clusters.

4.3.5 Community structure partition

We constructed a co-intensity network in which nodes represent the 187 samples while edges depend on the similarity information between them (figures 4.6A, 4.6B). Community structure partition through modularity optimisation yields 3 main clusters, distributed over 2 disconnected subgraphs. If we add the cellular scenario's colouring scheme on the edges of this network (figure 4.6C), the modules obtained through community structure correspond with precision to that scenario's biological groups. The network structure reflects experimental setup information: the upper subgraph (coloured red) represents the pellet, while the lower subgraph represents extra and intra-cellular classes through its two clusters (coloured blue and purple respectively).

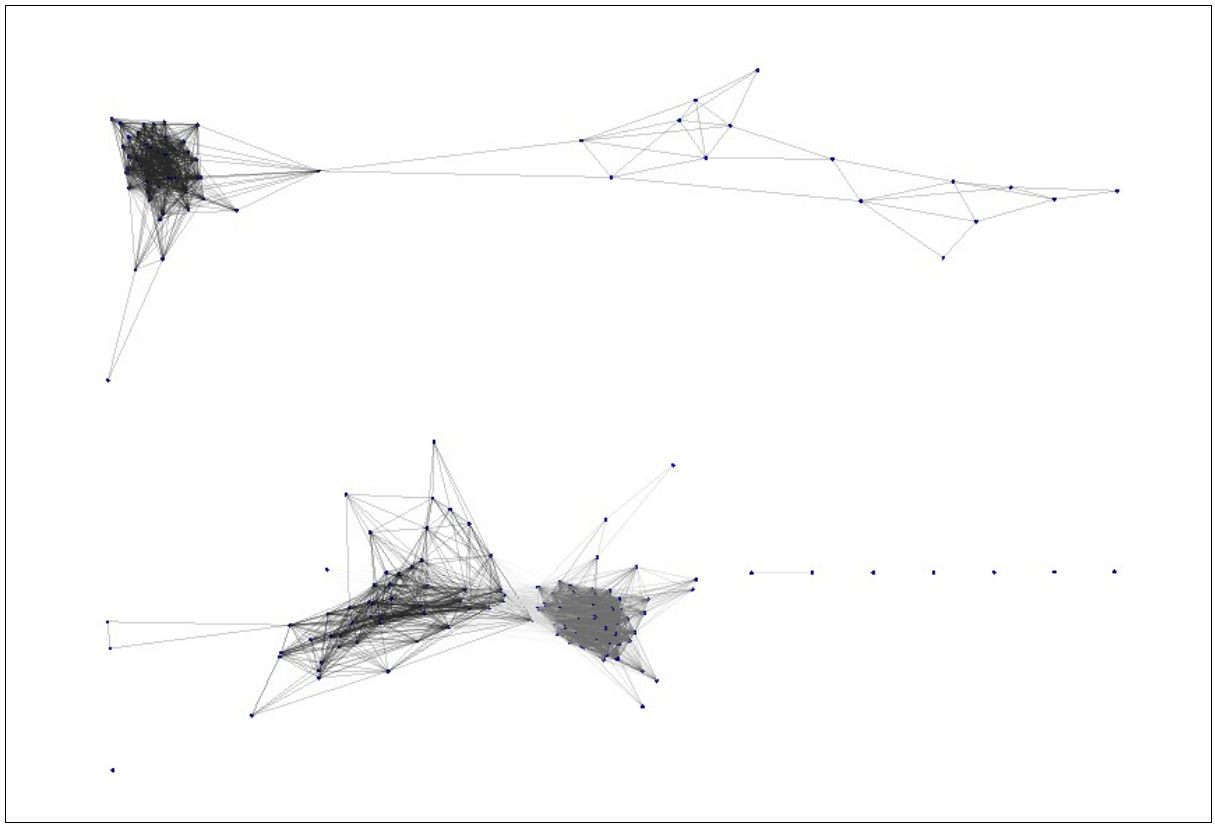

Figure 4.6A: Co-intensity network of the microdiversity dataset. In a purely unsupervised visualisation (where no colours are added for known groups) we observe the presence of three distinct subgraphs and two disconnected network components.

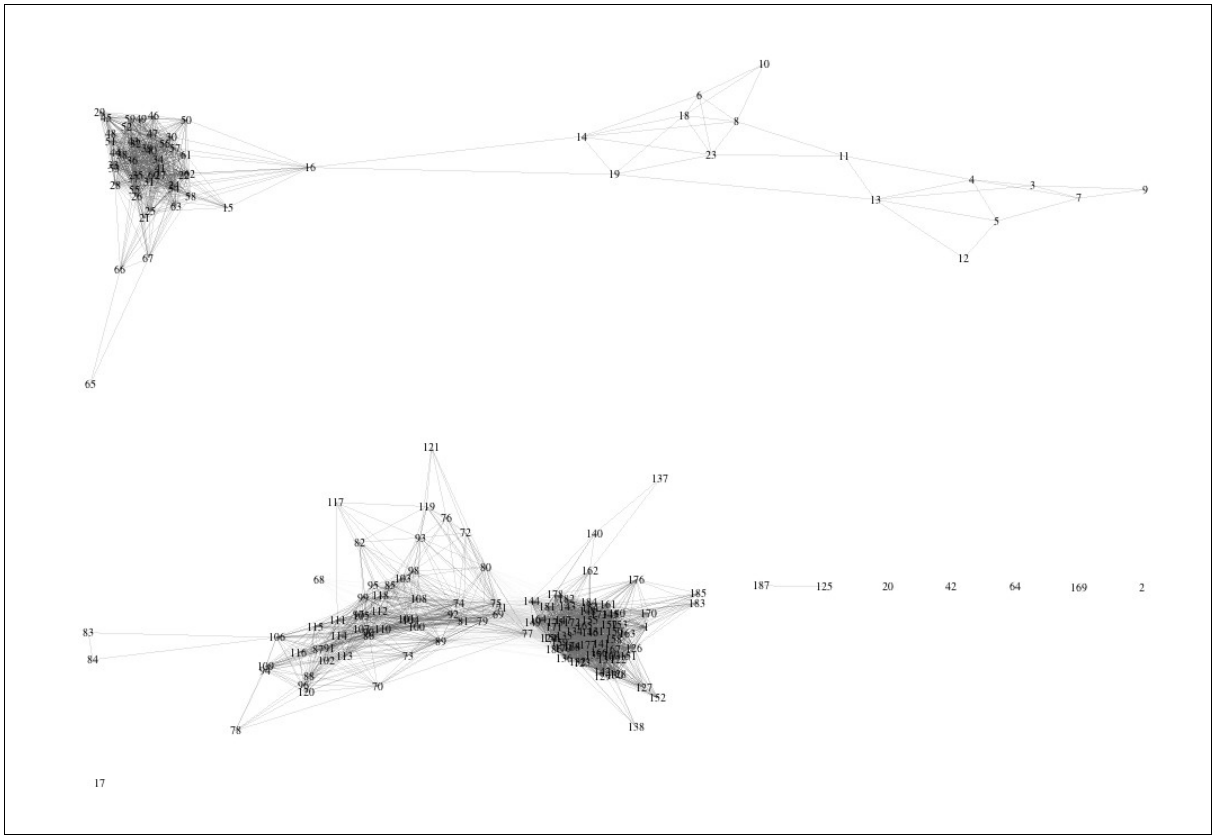

Figure 4.6B: Co-intensity network of the microdiversity dataset with sample indices on nodes. In a purely unsupervised visualisation (where no colours are added for known groups) we observe the presence of three distinct subgraphs and two disconnected network components.

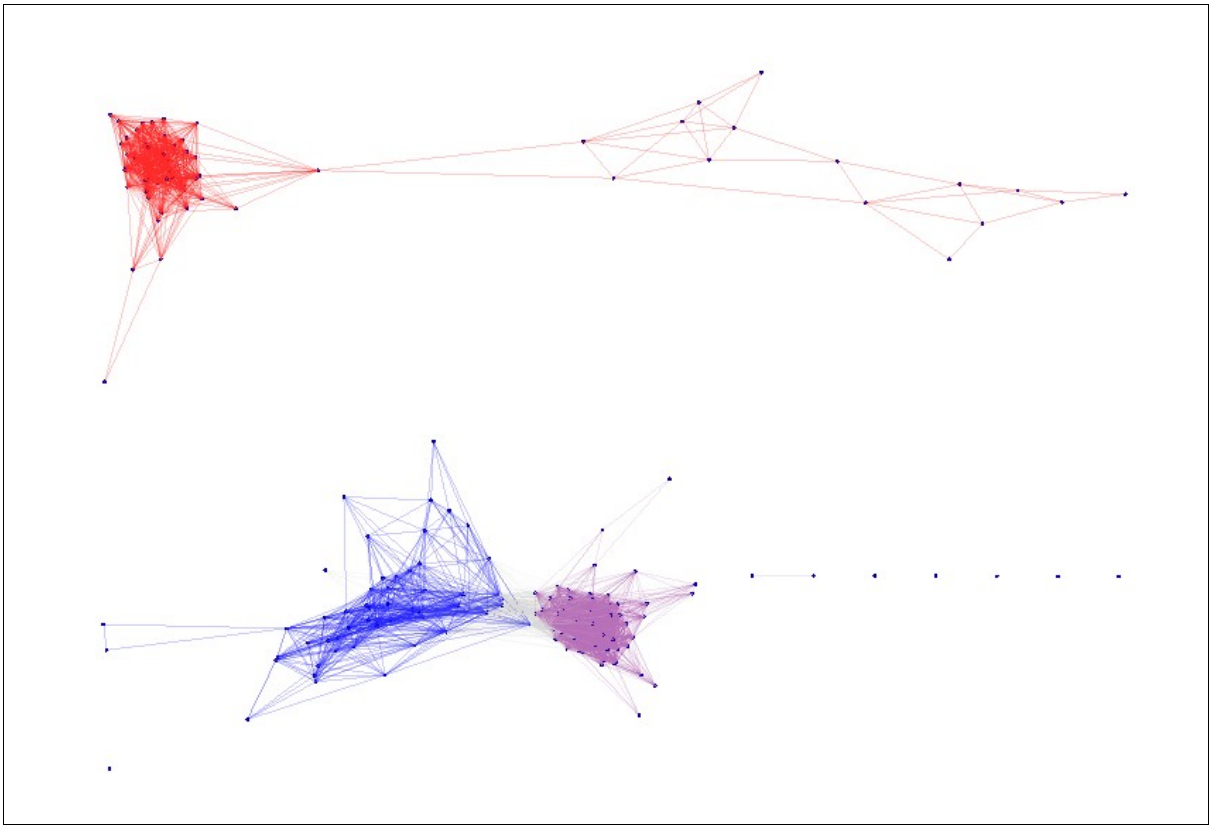

Figure 4.6C: Co-intensity network of the microdiversity dataset (at threshold 0.90) with cellular colouring (regions of blue, purple, red correspond to cellular classes extracellular, intracellular and pellet, respectively). The nearly perfect matching of colours and graph modules shows an efficient diagnostic performance of co-intensity network clustering on cellular groups.

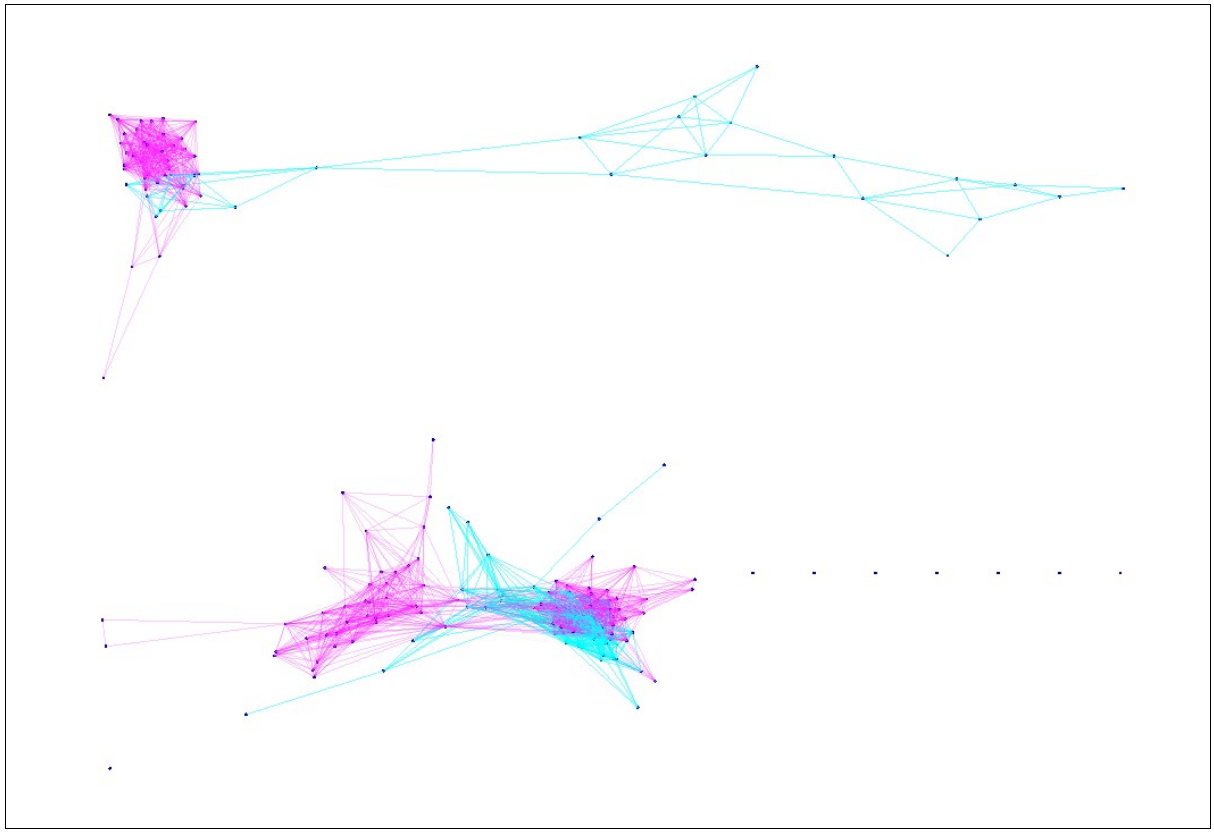

Figure 4.6D: Co-intensity network of the microdiversity dataset at threshold 0.90 with regional colouring (regions of cyan and magenta correspond to the regional classes Mallorca and Santa Pola). The strong matching of colours and modular subgraphs shows a diagnostic performance of co-intensity network clustering on regional groups.

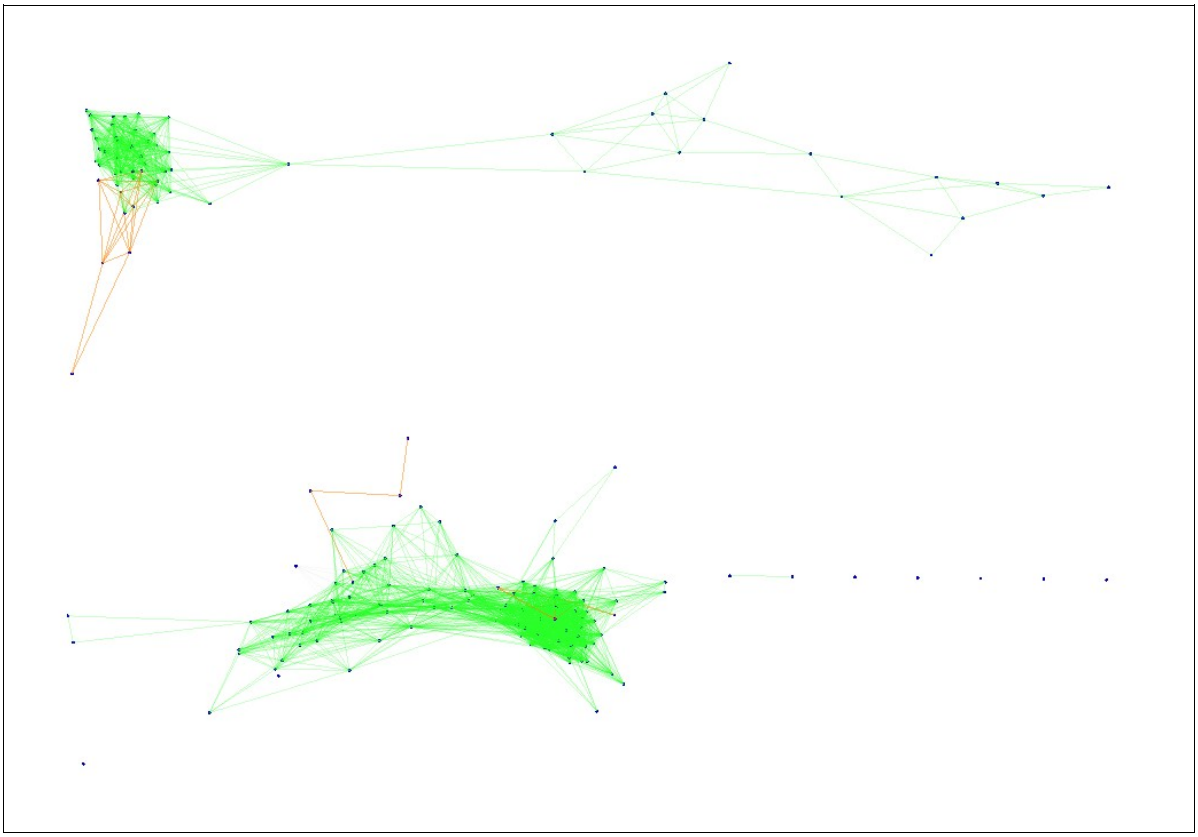

Figure 4.6E: Co-intensity network of the microdiversity dataset at threshold 0.90 with temporal colouring (regions of yellow and light green correspond to the temporal classes old and new). The overlapping of colours with parts of the graph shows the ability of co-intensity network clustering to reveal patterns on temporal groups.

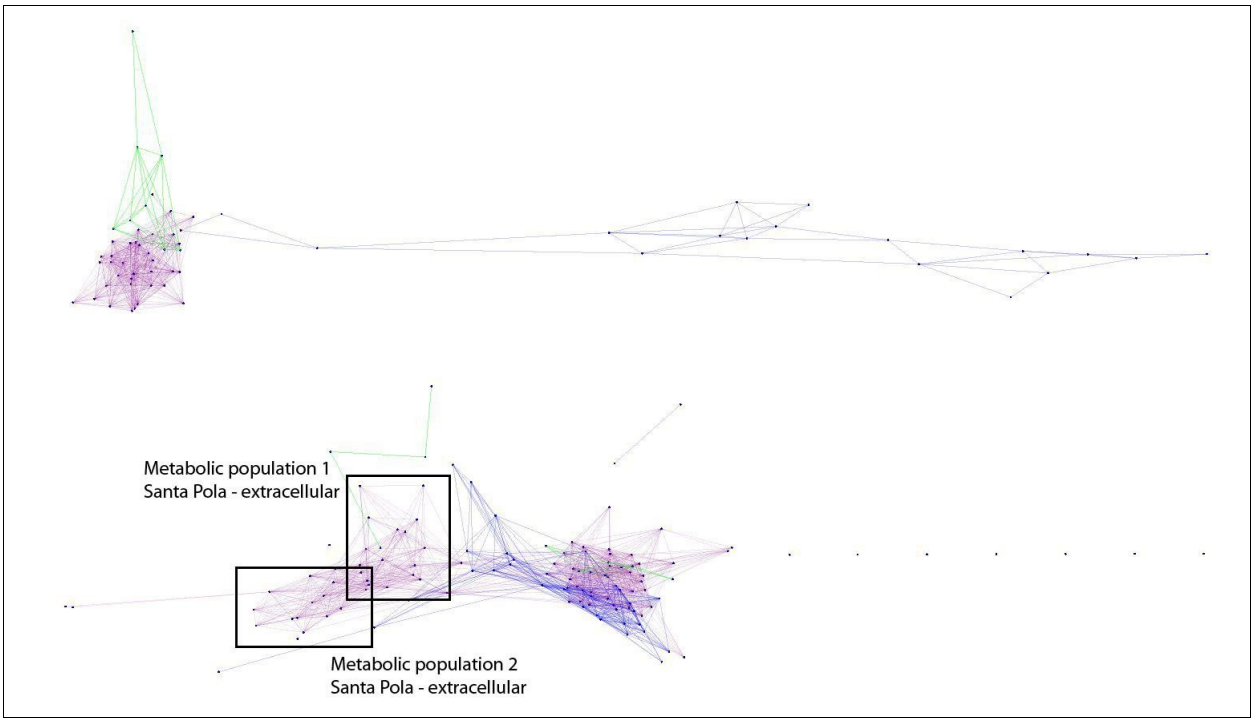

Figure 4.6F: Co-intensity network of the microdiversity dataset created at threshold 0.90 with information for all biological scenarios. Edges of blue and purple colour correspond to the regional classes Mallorca and Santa Pola, respectively. Edges of green highlight connections between the older reference strains. Extracellular extracts of the Santa Pola strains are separated into two topological regions of the network. The assertion of the publication on the existence of two distinct metabolic populations is verified by the presence of two clustered regions on the data's co-intensity network (marked in black squares).

The network of figure 4.6C can be replotted, using different colouring schemes, to reflect the regional (cyan - magenta) and temporal (light green - orange) classes of figures 4.6D and 4.6E, respectively. The cellular, regional, and temporal information of all three networks can next be combined into a single network; figure 4.6F. The new colouring scheme corresponds to regional (purple-blue) and temporal (green) classes, while the experimental setup (cellular) information is retained within the network's structure. Therefore, such a representation allows us to merge and compare all knowledge on the given samples. The Santa Pola and Mallorca strains tend to form modules among their own geographical classes, regardless of the sample preparation setup. Both blue and purple regions form modules which differentiate between extracellular and intracellular components. The green-coloured reference strains tend to separate from the new Santa Pola and Mallorca strains without exhibiting any geographical preferences among themselves. The old strains appear in peripheral network regions, inside the pellet, and extracellular extracts. Inside the intracellular extracts no such preference can be observed. This finding may be in consensus with the low separation power of genetic approaches [66]. Statistically predefined metabolic populations among the Santa Pola strains were found inside two distinctly different regions of the network, confirming thus the assumption of the existence of two different metabolic populations within this microhabitate.

4.4 Comparison of distance metrics

The starting point of most clustering algorithms is a similarity matrix, i.e. a matrix of pairwise similarities between the objects to be classified [38]. A given similarity metric can directly affect the performance of a clustering algorithm and is, therefore, a parameter that must be chosen wisely [38]. After deeming community structure partition as the optimal clustering algorithm, we study how its performance varies in combination with each of the selected similarity metrics described herein. Every subsection presents a different distance metric and uses it in the construction of a co-intensity network, in order to test its clustering performance on the cellular scenario.

4.4.1 Pearson correlation

The measure of linear dependence between two variables X and Y, evaluated within the range [-1, 1].

$$r = \frac{1}{(n-1)} = \sum_{i=1}^{n} \left(\frac{X_i - \bar{X}}{s_X}\right)\left(\frac{Y_i - \bar{Y}}{s_Y}\right)$$

where $\bar{X}$ and s_X are the mean and standard deviations of X, respectively.

The Pearson distance is defined as $1-|r|$. Figure 4.6 illustrates a co-intensity network whose nodes are linked via Pearson distance.

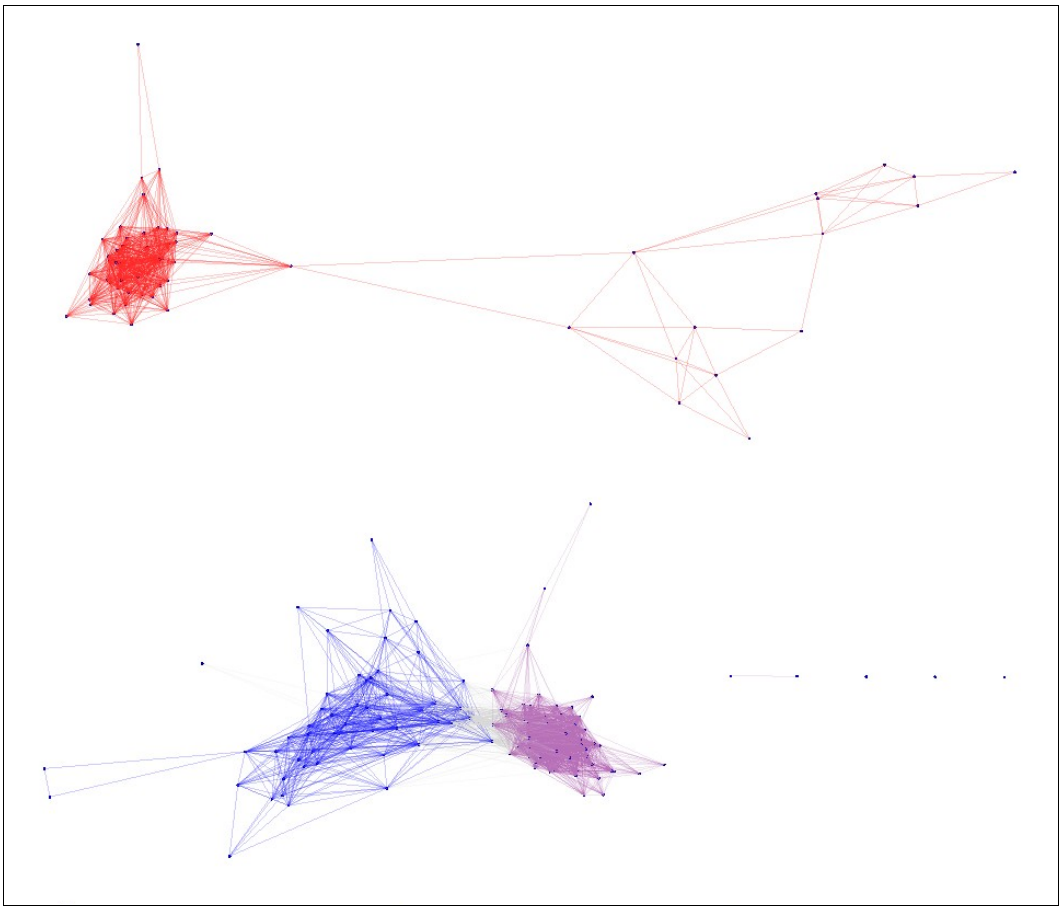

Figure 4.6: Co-intensity network of the microdiversity dataset using Pearson distance and cellular group colouring. The three biological groups appear distinctively as in three densely connected subgraphs.

4.4.2 Euclidean distance

The ordinary distance between two points, deriving from the Pythagorean formula.

For two points $A = (A_1, A_2, \ldots, A_n)$ and $B = (B_1, B_2, \ldots, B_n)$, the euclidean distance is measure by the formula:

$$d(A, B) = \sqrt{\left(\sum_{i=1}^{n} (A_i - B_i)^2\right)}$$

The normalised euclidean distance is calculated on an input matrix that has been scaled by dividing each element by its corresponding feature vector's maximum element, giving it a value in the rage [0,1]. Figure 4.7 illustrates a co-intensity network whose nodes are linked via normalised Euclidean distance.

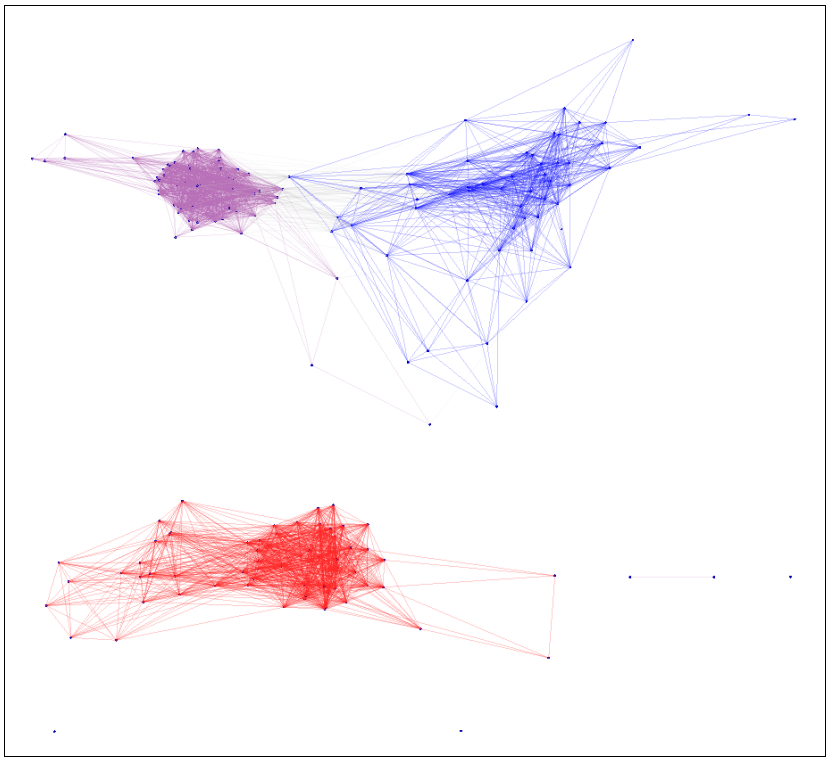

Figure 4.7: Co-intensity network of the microdiversity dataset using normalised Euclidean distance and cellular group colouring. The three biological groups appear distinctively as in three densely connected subgraphs.

4.4.3 Standardised Euclidean distance

Euclidean distance calculated on an input matrix that has been scaled by dividing each element by its corresponding feature vector's standard deviation. Figure 4.8 illustrates a co-intensity network whose nodes are linked via standardised Euclidean distance.

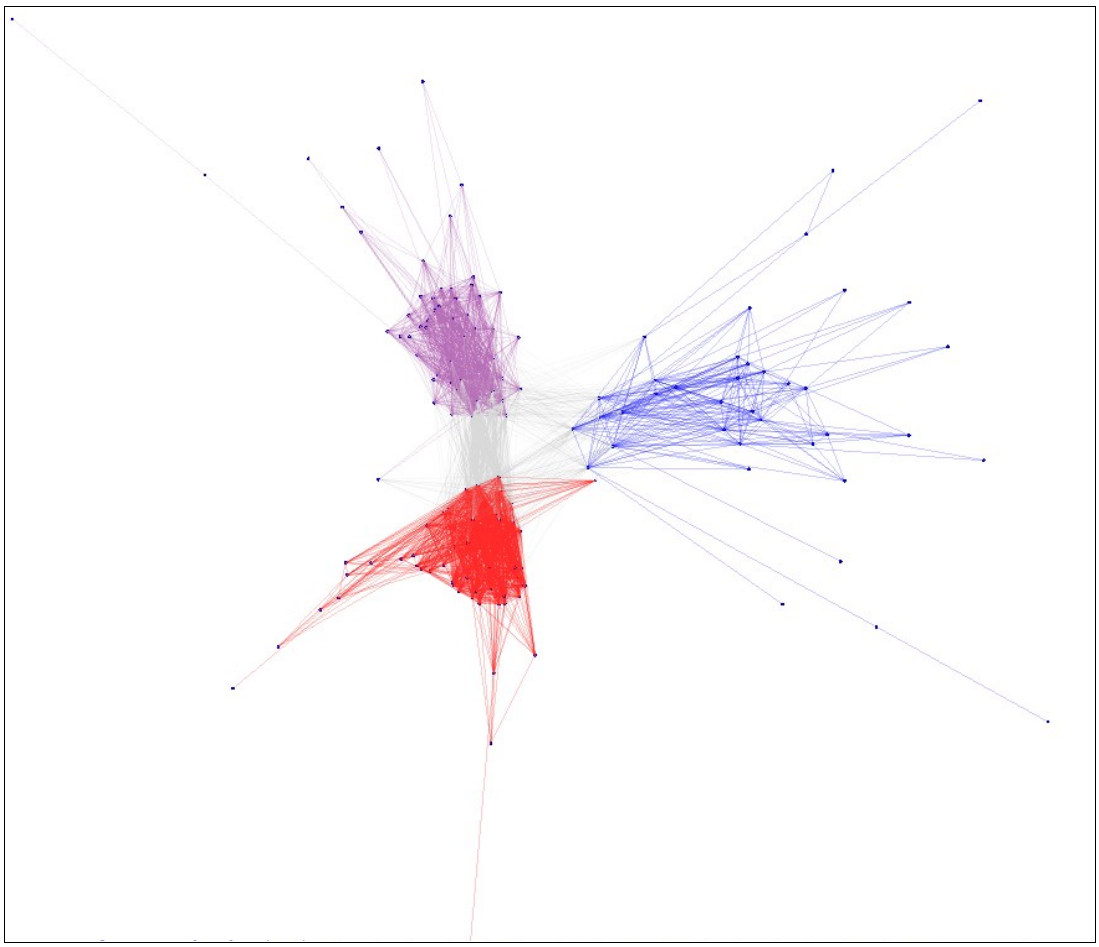

Figure 4.8: Co-intensity network of the microdiversity dataset using standardised Euclidean distance (normalised) and cellular group colouring. The three biological groups appear distinctively as in three densely connected subgraphs.

4.4.4 Cosine similarity

The metric of similarity between two vectors that measures the cosine of the angle between them. Using the Euclidean dot product formula, the cosine similarity between two vectors of attributes A and B is calculated by:

$$\cos(\theta) = \frac{\sum_{i=1}^{n} A_i \times B_i}{\sqrt{\sum_{i=1}^{n} (A_i)^2} \times \sqrt{\sum_{i=1}^{n} (B_i)^2}}$$

Figure 4.9 illustrates a co-intensity network whose nodes are linked via cosine similarity.

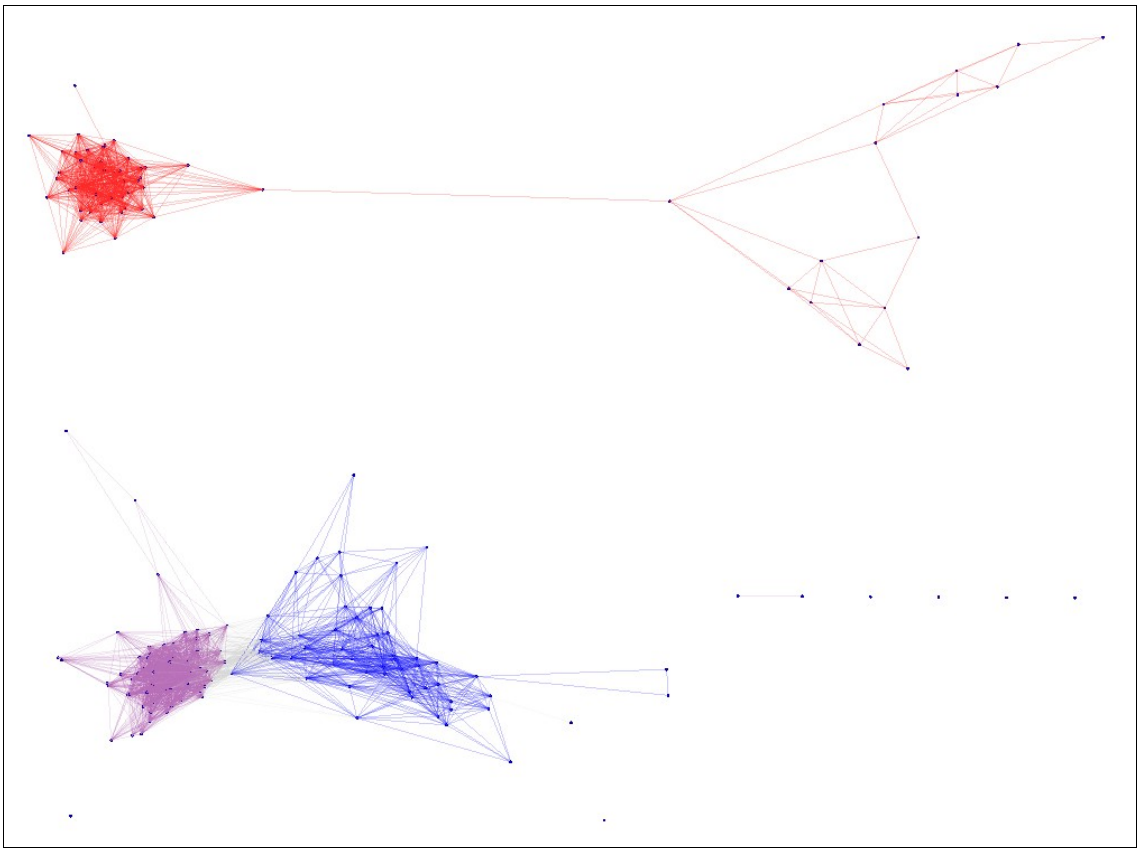

Figure 4.9: Co-intensity network of the microdiversity dataset using cosine similarity and cellular group colouring. The three biological groups appear distinctively as in three densely connected subgraphs.

4.4.5 Manhattan distance

The metric in which the distance between two points is measured by the sum of the absolute differences of their coordinates. For two points A and B in an *n*-dimensional real vector space with fixed Cartesian coordinate system is calculated by:

$$d_m(A, B) = \sum_{i=1}^{n} |A_i - B_i|$$

Figure 4.10 illustrates a co-intensity network whose nodes are linked via Manhattan distance.

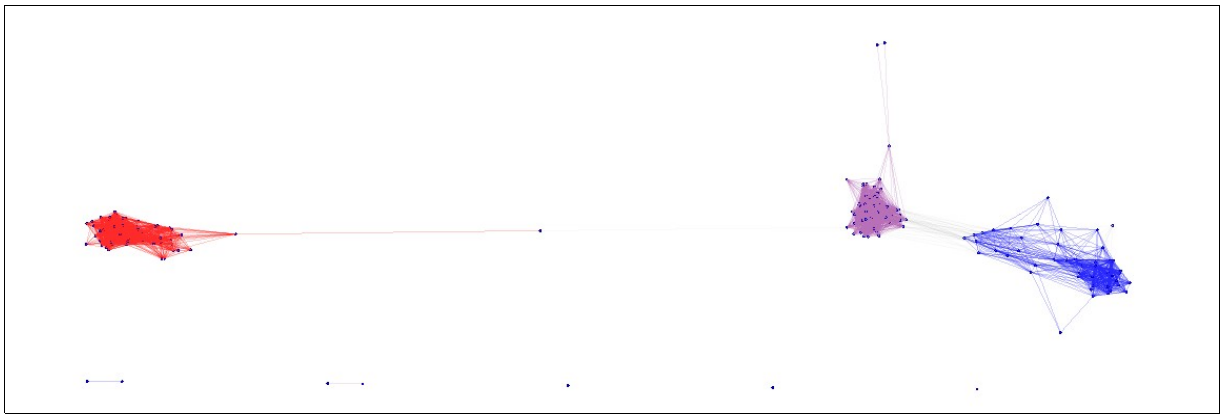

Figure 4.10: Co-intensity network of the microdiversity dataset using Manhattan distance (normalised) and cellular group colouring. The three biological groups appear distinctively as in three densely connected subgraphs.

4.5 Conclusion

In this chapter I first tested the performance of various widely used clustering algorithms in respect to the microdiversity dataset and its three classification scenarios. Hierarchical clustering performs well only when Spearman correlation is used as a similarity metric but fails to reflect any biological grouping beyond the basic scenario (cellular). The canonical version of the k-means algorithm performs almost equally well but is also restricted to the basic scenario due to its limited visualisation capabilities. In addition, the number of classes to be detected must be provided as an input parameter. The PCA algorithm displayed the poorest performance both in terms of classification on the basic scenario as well as feature extraction on the remaining scenarios (regional, temporal). Community structure partition displayed an outstanding clustering performance on cellular biological grouping and was the only algorithm to reflect patterns of the other two scenarios (all in a single visual display). The cellular information is reflected between disconnected subgraphs and modules provided by community structure partition, while the regional and temporal classes are forming patterns within those subgraphs. This biological information could not be observed in the output of the other algorithms; a fact which made community structure partition my method of preference in this work. After choosing graph-based clustering as the algorithm of preference, I tested its performance over a number of distance metrics in order to determine the optimal one. Normal and standardised Euclidean distance would classify the groups of the cellular scenario but would corrupt the structural information on the regional and temporal cases. Manhattan distance would improve the quality of clustering on the cellular scenario but, at the same time, would almost entirely lose the information on the other two scenarios. However, cosine similarity and Pearson distance were able to reflect all clustering scenarios in one single graph. Spearman distance was left out of the evaluation due to the closeness of its results to Pearson and cosine similarity.

CHAPTER V

A combinatorial learning framework for sample classification and discriminant signal identification in complex datasets

In this chapter, I present one of the main research topics of this work; namely the quantitative method, inspired by *Operational Research*, that I refer to as *combinatorial learning*. The framework is formally defined and developed into algorithms using discrete mathematical modelling. The last part the chapter deals with the applications of these models on real-world mass-spectrometric data, which yielded statistically significant results.

5.1 Abstract

Motivation**:** Fourier Transform Ion Cyclotron Resonance Mass Spectrometry (ICR-FT-MS) in non-targeted metabolomics is a tool of unmatched analytical power, which produces amounts of data comparable in size and complexity to those of DNA microarrays. The bottleneck in the quantitative analysis of such vast datasets lies on the computational identification of masses discriminant to different metabolic states combined with the efficient classification of samples into regions of varying risk. Conventional data mining approaches used to achieve these tasks are characterised by the introduction of bias through statistical assumptions and transformations, which puts their flexibility and efficiency into question [3]. To date, very few computational methods have

been developed for, or adapted to, ICR-FT-MS metabolomics in order to explore the vast potential of this analytical technique.

Results**:** The limitations of the current “standard” methods and the sparsity of in-depth quantitative research on the field of Fourier transform mass spectrometry metabolomics have inspired us to propose a combinatorial machine learning approach to the problem of discriminant signal identification and sample classification. The key aspect of this approach is the intuitive and flexible modelling which aims to minimise the statistical bias and biological inconsistency of conventional “black box” approaches. I investigated the applicability of an Operational Research model, which applies combinatorial optimisation with metaheuristic search algorithms in order to maximise an objective function proportional to the biological clusterability of a graph-theoretical object that I call *co-intensity network*. We tested the method on Crohn's disease dataset [4] and received biologically pertinent results in the areas of semi-supervised classification, diagnosis, and prediction. Due to the robustness and flexibility of the approach, I believe that it has the potential of becoming an alternative method to multivariate analysis, adapted for ICR-FT-MS metabolomics and possibly other fields of bioinformatics.

5.2 Introduction

Non-targeted metabolomics necessitates broadband detection width combined with high resolution and mass accuracy, which can only be provided by Orbitrap or ICR-FT-MS instrumentation. ICR-FT-MS's ultra-high resolution and mass accuracy allows for immediate annotation of thousands of chemical formulae and is able to produce extremely large datasets. Given such data size, the need of alternative computational techniques becomes evident. In this section I describe methods that have been primarily developed for and tested on ICR-FT-MS data but are, in theory, compatible with multivariate datasets of other -omics fields.

The distribution of Metabolomics data is not symmetrical nor normal and, in order to apply parametric methods such as *t-test* or *MANOVA*, the data has to undergo statistical pre-treatment which may lead to the introduction of bias and possibly affect downstream

statistical biomarker discovery [67]. Despite that risk, scaling and normalisation are applied in most cases of metabolomics analyses independently of what method will follow. Multivariate analysis typically starts with the application of a clustering algorithm on the normalised data in order to observe how the formed clusters compare to the actual biological groups, which comes down to estimating the amount of noise that exists in the data. Though not a true clustering algorithm, *Principal Component Analysis* (PCA) is one of the standard techniques used for such feature extraction due to its simplicity and ease of use [25][26]. The approach provides a general data overview but does not actually associate objects to clusters. More specifically, a PCA transformation performs dimensionality reduction through the creation of a new orthogonal basis whose axes are oriented in the directions of the maximum variance of the input matrix. The data can be consequently plotted over a two-dimensional space, from which the human expert may then empirically decide what objects cluster together. This efficient dimensionality reduction has proved PCA to be an appropriate method for data compression, however, not necessarily optimal for feature extraction, especially when features are meant to be used in a supervised classifier [25][26].

Regardless of what clustering algorithm is used, in the vast majority of cases the data is too noisy to be treated solely by unsupervised means, therefore a supervised approach is considered. *Partial Least Squares regression* (PLS) and its variants are currently among the most popular algorithms for supervised feature extraction on mass spectrometry data and chemometrics in general. The method was developed for the purpose of constructing predictive models for the case where predictor variables are numerous and highly collinear [68]. Nonetheless, PLS regression comes with a number of important limitations; in practical scenarios where both the number of cases in the sample and the number of indicators per latent variable will be finite, the algorithm tends to underestimate correlations between latent variables and overestimate the loadings [3], and there is globally “a higher risk of overlooking 'real' correlations and sensitivity to the relative scaling of the descriptor variables” [69]. Consequently, there are practical situations, such as when the number of samples is lower than a certain value, where the results of PLS cannot be taken for granted [3].

As an alternative to the existing methods, I developed a nonparametric machine learning model that is based on the combinatorial optimisation of a function that evaluates the '*quality*' of an arbitrary solution in respect to the latter's capability of clustering the data according to the known biological groups. A given solution comes in the form of a feature vector produced by an approximation algorithm and, in its optimal form, it can be used for classification/prediction, and feature extraction.
For the purposes of testing the methods presented in this section, I selected a dataset in which unsupervised classification reveals patterns but no clear clusters. In the Crohn's dataset [4], we can see some of the related data points appearing in close proximity in the feature space, albeit the clustering algorithm is unable to output discrete sets points that correspond to known biological groups (figure 5.1).

In the rest of this section, I first define the theoretical framework for my models within the context of Operational Research problem-solving. Next, I derive the algorithms required for solving these combinatorial problems and I present the results of my method's application on Crohn's disease dataset. Finally, I investigate potential improvements which may lead to a more efficient performance.

5.3 Metabolomics study on Crohn's disease

Crohn's disease is an inflammatory bowel disease characterised by chronic inflammation of the gastrointestinal tract. The causes of Crohn's disease are largely unknown, albeit dependent on both genetic and environmental factors [4]. Efficient, non-invasive, diagnostic and monitoring tools remain currently inadequate even though various biomarkers have been proposed in literature. Recent advances in nuclear magnetic resonance (NMR) and mass spectrometry (MS) have led to the assessment of the metabolites that make up the ‘‘metabolome’’, consequently, determining end-points of metabolic processes in living systems [4][70]. ICR-FT-MS with an ultra-high mass resolution is able to differentiate subtle variations between thousands of mass signals, including higher molecular weight metabolites [4][71]. As shown in a previous study on markers of diabetes and early stage insulin resistance, the combination of coupled metabolite separation techniques with spectrometry and spectroscopy contributes to the

structural identification of new metabolites [4][72].

The aim of the study in question [4] was to discover metabolic biomarkers of Crohn's disease as evidence of microbial functions in the gut. The high dynamic range and mass accuracy of ICR-FT-MS was used to obtain non-targeted profiles of elementary compositions in samples of individuals diagnosed with Crohn's disease [4]. Classical techniques, such as principal component analysis (PCA) and partial least square regression (PLS), were used for all data analysis tasks [4].

5.4 Methods, models, and algorithms

5.4.1 Method framework overview

The typical format of a ICR-FT-MS dataset comes in the form of a *mass-sample intensity matrix*, i.e. an $n \times m$ matrix $A = [a_{ij}]$, where a_{ij} holds the value of an ICR-FT-MS intensity. The row-vectors of A correspond to the n measured masses while the column vectors correspond to the m sampled observations, therefore, the value of a given element a_{ij} of A signifies the intensity of the exact mass corresponding to row i measured for subject j. Furthermore every ICR-FT-MS dataset, represented by matrix A, is associated to number of q biological groups which classify the m number of samples (column-vectors of A). I refer to the known biological groups as our *supervised information* and we may represent this classification as a collection of q disjoint sets $C = \{C_1,...,C_k,...,C_q\}$. Every sample in our dataset is associated to one of these classes/groups, therefore, if $M=\{x \in Z: 1 \geq x \geq m\}$ is the set representing the m sampled observations in A, we have:

$$\forall k\left(C_k \subset M\right), \quad \cup C = M, \quad \cap C = \emptyset \quad \text{and} \quad \forall x \exists y \left(x \in M\right)\left(y \in C\right), x \in y.$$

By applying a clustering algorithm on A, we get to observe the natural grouping formed by the sampled observations M over the raw dataset. I introduce the measure of *biological clusterability* in order to evaluate the tendency of these *in silico* clusters to coincide with known biological groups. If the clustering output is in accordance with the known grouping, then the value of clusterability is directly proportional to the graph's

modularity (a formal definition of clusterability is provided in a later section). In the case of a noisy raw dataset, clustering matrix *A* by means of any algorithm will yield classes which do not coincide well (or at all) to the supervised information in *C*. In this case, we say that these clusters are of no biological pertinence and, therefore, we consider that the *quality of clusterability* of the current sample-intensity matrix is low due to the existence of noise. In other words, the amount of noise within an ICR-FT-MS dataset can be evaluated via the quality of the data's *clusterability* in respect to the experimentally known biological groups, i.e. to what extent do the in silico discovered classes correspond to the actual biological ones.

Given this framework, the key objectives of data mining in metabolomics are the following:

(a) Discover a group of masses which contributes to the known biological grouping as a whole, i.e. keep only the row-vectors of *A* which reflect biological pertinence in respect to *C*.

(b) Discover which groups of masses are contributing to the formation of each individual biological group, i.e. isolate *q* groups of row-vectors in *A* which reflect biological pertinence in respect to each of the *q* classes in *C*.

The intuitive assumption, on which my method is based, is that we are able to obtain an almost-perfect, biologically-pertinent clusterability by removing the row-vectors (corresponding to exact masses) from the *n* rows of the raw mass-sample intensity matrix which make up the noise. The question regarding which row-vectors to remove out of a total of (e.g.) *n*=20,000 metabolites, is modelled as an Operational Research *combinatorial problem* that can be solved by means of mathematical optimisation. As it is later explained in more detail, a possible *solution to* such an optimisation problem refers to a given mass-sample intensity matrix *A'*, sub-matrix of *A*, from which some row-vectors are missing. Such a *solution* can be represented mathematically using two separate models of varying characteristics and combinatorial complexity. For each model, the *quality of clusterability* of a given solution to the problem (i.e. the *quality of a solution*) is expressed in the form of a real number, which is yielded by a pre-defined

objective function. From any candidate solution, a co-intensity network can be constructed and the *m* samples of the initial dataset can be clustered with varying efficiency in respect to the supervised information in *C*. An obvious goal would, therefore, be to find an *optimal solution* which will cluster the data more efficiently than most other candidate solutions. An objective function receives a candidate optimal solution to the problem and estimates to what extent the clusterability of the co-intensity network of that solution is of biological significance. The larger the real number yielded by the objective function, the better the *quality of the solution*, therefore the optimisation consists of maximising the given objective function; a task which can be achieved in numerous way.

Searching for the optimal solution through a dataset of 20,000 variables is a task of immense computational complexity. Assuming that a solution of size *n* is modelled by a binary vector of *n* bits, $S = [s_i]_n$, where each s_i signifies the absence or the presence of the current variable within a candidate solution, then our search space makes up a total of 2^n combinations. An exhaustive search would therefore need to go through $2^{20,000}$ candidate solutions in order to find the globally optimal one. The time required for an exact algorithm (performing brute-force search) to go through all possible combinations increases exponentially with the size of the problem *n*, therefore for a problem size as small as *n*=1023 the algorithm would have to go through $2^{1023} = 8.9885 \times 10^{307}$ possible combinations. Assuming that, using an ultra-powerful computer, the algorithm examines fifty million combinations per second, it then would require at least $2^{1023} \div 5 \times 10^6 = 5.7526 \times 10^{297}$ years to perform an exhaustive search over all possible solutions over the entire search space. Hence, for obvious reasons, it is deemed practically impossible that such an algorithm can yield an optimal solution to combinatorial problems of larger sizes, independently of the machine power available. This computational explosion in unconstrained combinatorial optimisation is dealt with a class of nondeterministic approximation optimisation algorithms known as *metaheuristics*. Stochastic optimisation involves the application of “intelligent” search via probabilistic means in order to find optimal or near-optimal solutions to hard problems in polynomial time [73].

For the evaluation of a candidate solution and the purpose of unsupervised classification, I chose the graph-theoretical method of *community structure partition* over the other clustering algorithms due to its superior performance in terms of classification precision and information richness (as shown in chapter IV). This kind of graph-based representation is integrated with varying efficiency in many software tools and is frequently used in other branches of bioinformatics when feature extraction is combined with visualisation (though rarely with clustering capabilities); however, when cluster analysis is the main focus, simpler algorithms such as hierarchical clustering, k-means or even PCA are preferred. Graph clustering is typically an NP-complete problem [74] which involves mathematical optimisation in order to discover the optimal modular partition of a network in polynomial time and, due to this high computational complexity, there is an added difficulty in re-implementing the technique from scratch. Nonetheless, in graph-based clustering (as in most clustering algorithms) a similarity/distance matrix is constructed out of the raw dataset. This matrix must be transformed into a binary adjacency matrix in order to produce the actual graph which represents the data. An $n \times m$ ICR-FT-MS mass-sample intensity matrix is therefore used to create a $m \times m$ similarity matrix holding the distances between all m samples. An $m \times m$ adjacency matrix is then constructed by setting a threshold to the distance values in the similarity matrix. This adjacency matrix represents an unweighted, undirected graph, which I refer to in this work as a *co-intensity network*. We can visualise a co-intensity network and extract its community structure via specialised algorithms [61] in order to examine the biological clusterability of the data at hand. The co-intensity network of the Crohn's disease raw dataset can be seen in figure 5.1. The colours of the edges correspond to the links between biological groups which are meant to cluster together and, even though the modules of the graph have a seemingly distinct pattern, the separation is not clear enough for the community structure partition algorithm to detect them automatically. The output of the clustering algorithm yields modules which do not correspond to the expected biological groups in C, and if the colours were to be removed then the graph would be hardly at all informative.

5.4.2 Community structure clustering of co-intensity networks

Similarity and adjacency matrix construction:

Let $A=[a_{ij}]_{n\times m}$ be a mass-sample intensity matrix.

The similarity matrix ${}^{S}A$ of size $m\times m$ is obtained by:

$${}^{S}A=[{}^{S}a_{ij}]_{m\times m},$$

where

$${}^{S}a_{ij}=\underset{p=1}{\overset{n}{\forall}} p \overset{m}{\forall} i \overset{m}{\forall} j, F_{\text{sim}}(a_{pi}, a_{pj}),$$

and F_{sim} is a similarity or normalised distance function applied on the column vectors of A (in my case the similarity metric used was a variation of the Pearson distance $P \equiv 1 - r$, where r is the Pearson correlation coefficient).

At this point any clustering algorithm may be applied on ${}^{S}A$. However, since we have chosen a graph-based approach (community structure partition clustering), we first need to create a graph in the form of an adjacency matrix by applying a threshold value on ${}^{S}A$.

The binary or *adjacency matrix* ${}^{B}A$ of size $m\times m$ is obtained by:

$${}^{B}A = [{}^{B}a_{ij}]_{m\times m},$$

where

$${}^{B}a_{ij}= \begin{cases} 1 \;\; iff \;\; {}^{S}a_{ij} \geqslant T \\ 0 \;\; iff \;\; {}^{S}a_{ij} < T \end{cases}$$

Therefore, ${}^{B}A$ is used as the adjacency matrix of a *sample co-intensity network*.

Co-intensity network:

A co-intensity network is a graph-theoretical tool which allows us to model a ICR-FT-MS mass-sample intensity matrix in the form of a similarity network by using intensity information solely. The graph is obtained by creating an $m \times m$ adjacency matrix out of the m column vectors of an $n \times m$ mass-sample intensity matrix.

We can formally define a *sample co-intensity network* as an undirected graph $G = (V, E)$, where:

1. The set of vertices V represents a set of sampled subjects M corresponding to the intensity column-vectors of a mass-sample intensity matrix $n \times m$;
2. The set of edges E represents the set of biological similarities which exist between pairs of samples in M; where the cardinality of E is equal to the number of nonzero elements in a solution vector S.
3. An adjacency function $f : M \rightarrow R_S$ which, given the correlation coefficient $r(x,y)$ of two intensity column-vectors, determines the existence of a biological similarity between two samples in M, i.e. the adjacency relation R_S between two nodes in G. The domain of the binary relation R_S is the set of samples $x \in M$, such that there exists a $y \in M$ with $(x , y) \in R_S$. The similarity relation $x \; R_S \; y$ is true when the output of the adjacency function is equal to one, i.e. when the correlation coefficient between x and y is superior (or inferior) to a fixed threshold value T.

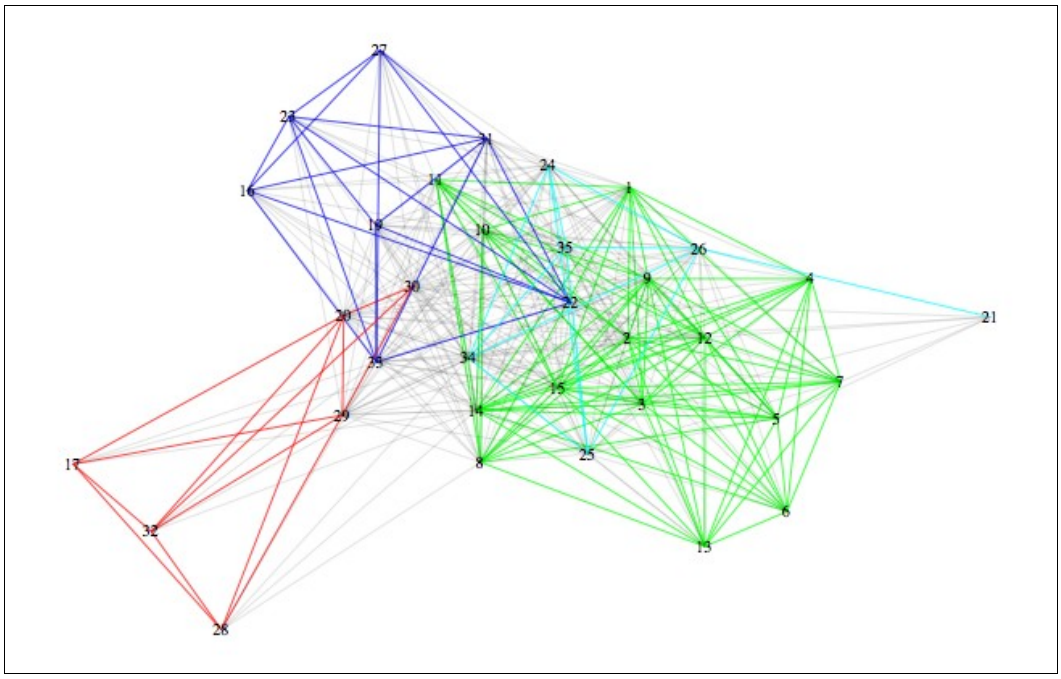

Figure 5.1: Co-intensity network created out of the raw Crohn's dataset. Green, blue, and red edges link samples belonging to the same biological group, HH, CCD, ICD, respectively. The graph shows patterns but is not modular enough to yield sets of nodes forming distinct clusters.

5.4.4 Combinatorial problem modelling

Axiom 1:

Let Δ *be a set of n distinct elements,* Δ'_k *denote an arbitrary subset of* Δ*, and P(*Δ*) denote the powerset of* Δ *such that* $|P(\Delta)|=2^n$ and

$$\underset{k=1}{\overset{2^n}{\forall}} k\,,(\Delta_k^{'}\in P(\Delta))\wedge(\Delta_k^{'}\subseteq\Delta)\,.$$

There exists a function $\Phi_o: \Delta'_k \rightarrow R^+$ *such that:*

$$\overset{k\neq\alpha}{\forall} k\ \exists\,\Delta_\alpha^{'}\subseteq\Delta\,,\Phi_o(\Delta_\alpha^{'})>\Phi_o(\Delta_k^{'})$$

The above axiom expresses the fundamental hypothesis used to model our biological scenario as a combinatorial problem and corresponds the 1st objective from chapter I, section 1.1. The set Δ represents an abstraction of our raw dataset. The elements $x_i \in \Delta$ represent the row vectors (masses) of a mass-sample intensity matrix *A*, which can be either included or omitted as elements in an arbitrary subset Δ'_k with a total of 2^n combinations. Function Φ_o is the evaluation or objective function which estimates the proximity of the output of a given clustering algorithm applied on a mass-sample intensity sub-matrix to the known biologically-pertinent classes. Therefore, the real value returned by the function represents the "quality" of clusterability of the given submatrix in respect to that supervised information, i.e. to what extent the chosen masses/row-vectors are capable of clustering the data correctly over the samples. The objective is, intuitively, to determine the Δ'_α which yields the near optimal value in Φ_o, something which is equivalent to the removal of noise from the raw data. The unconstrained combinatorial problem deriving from *axiom 1* can be solved efficiently using metaheuristic search algorithms. In theory, such a task involves finding a near-optimal solution (global optimum), however, from the perspective of biology every "good" solution (local optimum) may have its own unique importance.

General framework of combinatorial optimisation in Operational Research:

In order to understand how axiom 1 can be modelled into a combinatorial problem we need to consider the general framework of discrete optimisation.

In mathematics and Operational Research, the term 'optimisation' comprises all methods that yield the optimum of a function. A combinatorial optimisation problem is defined by the set of its instances along with their corresponding solutions. For a given instance of the problem, the goal is to find one of the "best" admissible solutions $\lambda \in \Lambda$ (where Λ is the discrete set of admissible solutions corresponding to that instance). The quality of a solution is determined by the objective function that we are wishing to optimise. Therefore an optimisation problem consists of determining:

$$max\{\Phi_0(\lambda) : \lambda \in \Lambda\}$$

Let Λ be the set of solutions of an optimisation problem and Φ_o be the objective function. The neighbourhood structure is a function N which associates a subset of Λ to all solutions $\lambda \in \Lambda$. A solution $\lambda' \in N(\lambda)$ is called a *neighbour* of λ. A solution $\lambda' \in \Lambda$ is a local optimum to the neighbourhood structure N if $\Phi_o(\lambda) \leq \Phi_o(\lambda')$, $\forall \lambda' \in N(\lambda)$. A solution $\lambda' \in \Lambda$ is a global optimum if $\Phi_o(\lambda) \leq \Phi_o(\lambda')$, $\forall \lambda' \in \Lambda$.

Model 1 (binary encoding):

The set of solutions Λ in the general framework corresponds to the powerset $P(\Delta)$ of *axiom 1*. An admissible solution λ in the general framework, i.e. an arbitrary subset Δ'_k in *axiom 1*, was modelled as a binary vector $S = [s_i]_n$ where $s_i \in \{0,1\}$, $\forall i$. In a machine learning context this decision vector is called a *feature vector* or a *predictor variable* and, combined with the objective function, makes up the model to be learned. The objective function Φ_o can be considered the same as in (formula1) and (formula2).

The neighbourhood structure function N: $S' \leftarrow S$ is obtained by the following algorithm:

Neighbourhood function : *N*

input: current solution S where $|S| = n$, index p where $1 \leq p \leq n$

output: neighbour solution S'

$$S' \leftarrow S$$
$$S'_p \leftarrow \neg S_p$$

Axiom 2:

Given a set $N=\{x \in Z: 1 \geq x \geq n\}$ and a collection of q disjoint sets $C=\{C_1,...,C_k,...,C_q\}$, there exists an irreflexive, symmetric, binary relation R on N and C such that:

$$\exists x \exists y, (x \in N) \wedge (y \in C) \wedge significant(x, y) \leftarrow x \, R \, y$$

The second axiom expresses the second main objective of quantitative analysis in metabolomics as described in chapter I, section 1.1, meaning to find which masses, represented by the elements of *N*, are significant for each of the biological groups of samples represented by the elements of C. The added information is the constraint that every mass found to be present in a solution must be additionally associated to one and only one biological group. Consequently, a second model is necessary in order to represent a solution vector which holds that additional information.

Model 2 (base-*q* encoding):

The binary vector *S* of *Model 1* holds the information of whether or not a variable (row-vector of a mass) is present in a solution and has an impact on the solution's quality. Unless enforced by the objective function, this model does not directly associate every individual row-vector to a column-vector, which may be of interest in the case of biomarker identification as expressed by the binary relation *R* of *axiom 2*. An extension of model 1 is the base-q encoded vector ${}^{q}S = [{}^{q}s_i]_n$ where ${}^{q}s_i \in \{0...q\}, \forall\, i$, and $q = \left| C \right|$ is the number of known biological groups. In the case of Crohn's disease, the instance of the problem has $q = 3$ for each of the three biological groups ICD, CCD, HH, i.e. four possible states for every row-vector ${}^{q}s_i \in \{0, 1, 2, 3\}$, where ${}^{q}s_i = 1$, ${}^{q}s_i = 2$, ${}^{q}s_i = 3$ means that mass m_i is associated to class 1 (ICD), class 2 (CCD), or class 3 (HH), respectively, and ${}^{q}s_i = 0$ being the state of the absence of the row-vector corresponding to m_i from the given solution. This encoding allows us to integrate a unique mass-to-sample association into a single solution vector and introduces a constraint to the optimisation problem.

The neighbourhood and objective functions optimising this model have to be modified accordingly.

Neighbourhood function : ${}^{q}N$

Input: current solution ${}^{q}S$ where $\left|{}^{q}S\right| = n$, index p where $1 \le p \le n$, classification size q

Output: neighbour solution ${}^{q}S'$

$${}^{q}S' \leftarrow {}^{q}S$$

$${}^{q}S'_{p} \leftarrow \underset{{}^{q}S_{p}}{argmax}[{}^{q}\Phi_{o}({}^{q}S_{p})], \forall ({}^{q}S_{p}) \in \{0,\ldots,q\}$$

5.4.5 Metaheuristic algorithms and problem resolution

Objective function 1: Φ_o

The objective function Φ_o, which is adapted to *model 1*, evaluates a given mass-sample intensity matrix, which can be either the raw dataset or any sub-matrix of the raw dataset returned by the neighbourhood function *N*. The function receives an intensity matrix $A = [a_{ij}]_{n \times m}$, a solution vector $X = [x_i]_n$ in the form of *model 1*, and class set $C' \in P(C)$. A similarity matrix ${}^{S}A$ is constructed from *A* and the information in *X* by removing the row-vectors of *A* which have zero values in *X*:

$${}^{S}A = [{}^{S}a_{ij}]_{m \times m} \text{ where } {}^{S}a_{ij} = \overset{x_q=1}{\forall} q \overset{m}{\forall} i \overset{m}{\forall} j, F_{\text{sim}}(a_{qi}, a_{qj})$$

A *clusterability score* value E_s is calculated by counting how many column-vector in ${}^{S}A$ cluster correctly with respect to *C'*. This is achieved by checking whether the similarities between the samples expected to cluster together are above (or below) a threshold value *T* which iteratively varies over a certain range. This score, which represents the quality of solution *X*, is returned by the objective function and used as the value to be maximised in the optimisation process.

Hence, the *clusterability score* can be defined algebraically as:

$$E_s = \sum_{k=1}^{|C'|} \left[\sum_{i=1}^{|C_k|} \sum_{j=1}^{|C_k|} g_1({}^{S}a_{C_k^i C_k^j}) + \sum_{k'=k+1}^{|C'|} \sum_{i=1}^{|C_k|} \sum_{j=1}^{|C_k|} g_0({}^{S}a_{C_k^i C_{k'}^j}) \right], \; C_k \in C'$$

where g_1 and g_0 are activation functions of the form:

$$g_1(x) = \begin{cases} 1 \; iff \; x \geq T \\ 0 \; iff \; x < T \end{cases}$$

and

$$g_0(x) = \begin{cases} 1 \; iff \; x < T \\ 0 \; iff \; x \geq T \end{cases}$$

The above mathematical expression describes that the clusterability score is equal to the sum of all intra-modular positive node similarities plus the sum of all inter-modular negative node similarities. The higher the clusterability score, the better the quality of the solution, and the search algorithm tries to maximise the objective function by favouring higher quality values. The objective function returns additionally the similarity matrix ${}^{S}A$ as well as the optimal similarity threshold T_{opt} required to construct a co-intensity network for mass-sample intensity sub-matrix A (information necessary for the optimal solution's application on the raw dataset). Therefore, the tuple $< E_s, {}^{S}A, T_{opt} >$, returned by the objective function for a given solution vector X, denotes the quality E_s of solution X for a graph constructed by setting threshold T_{opt} on similarity matrix ${}^{S}A$.

The internal structure of the objective function, such as which subset of biological groups $C' \subseteq C$ to take into account or how the quality of clusterability score E_s is to be calculated or even whether a completely different evaluation algorithm is to be used, is subject to parametrisation according to the needs of the search and the biological problem at hand. This potential for parametrisation (dependent on expert judgement) is something that offers near-limitless flexibility to the method.

Algorithm 3: Objective function Φ_o

${}^{S}A$ derives from objective function Φ_o over the row indices of A whose values in X are equal to one:

$${}^{S}A=[{}^{S}a_{ij}]_{m\times m} \text{ where } {}^{S}a_{ij}=\overset{x_q=1}{\forall}\, q \overset{m}{\forall} i \overset{m}{\forall} j, F_{\text{sim}}(a_{qi}, a_{qj})$$

Input: intensity matrix $A = [a_{ij}]_{n\times m}$, solution vector $X=[x_i]_n$, class set $C' \in P(C)$
Output: quality of clusterability E_s, similarity matrix ${}^{S}A=[{}^{S}a_{ij}]_{m\times m}$, optimal similarity threshold T_{opt}

$E_s \leftarrow 0$
$T_{opt} \leftarrow -1$

$\forall T \in (0.01, 0.02, \ldots, 1)$
  $E_s \leftarrow 0$
  $\forall k \in (1,2,\ldots,|C'|)$
    $\forall i \in (1,2,\ldots,|c_k'|), c_k' \subset C'$
      $\forall j \in (i+1, i+2, \ldots, |c_k'|), c_k' \subset C'$
        if $({}^{S}a_{c_k^i c_k^j} \geq T)$
          $E_s \leftarrow E_s + 1$
          $T_{opt} \leftarrow T$
        end-if
      end-for
      $\forall k' \in (1,2,\ldots,|C'|), k' \neq k$
        $\forall i \in (1,2,\ldots,|c_{k'}'|), c_{k'}' \subset C'$
          $\forall j \in (i, i, \ldots, |c_k'|), c_k' \subset C'$
            if $({}^{S}a_{c_{k'}^i c_k^j} < T)$
              $E_s \leftarrow E_s + 1$
              $T_{opt} \leftarrow T$
            end-if
          end-for
        end-for
      end-for
    end-for
  end-for
end-for

Objective function 2 (base-*q* model): ${}^{q}\Phi_o$

Objective function ${}^{q}\Phi_o$ is adapted for *model 2* and produces a clusterability quality score from an intensity matrix *A* produced by neighbourhood function ${}^{q}N$. Its input is the same as in *Objective Function 1*, i.e. an intensity matrix $A=[a_{ij}]_{n\times m}$, a solution vector $X=[x_i]_n$ in the form of *model 2*, and class set $C'\subseteq P(C)$. A similarity matrix ${}^{s}A$ is constructed in the same way as in *Objective Function 1* and additionally a *q* number similarity matrices ${}^{S}_{\kappa}A=[{}^{S}a_{ij}]_{m\times m}$ are constructed over the row indices of *A* whose values in *X* are equal to κ, $\forall\,\kappa\in\{1,\ldots,q\}$, $C_\kappa\in C'$, C_κ corresponding to a class in *C'* such that:

$$\forall\,\kappa\in\{1,\ldots,q\},\ {}^{S}_{\kappa}A=[{}^{S}_{\kappa}a_{ij}]_{m\times m}\ \text{where}\ {}^{S}_{\kappa}a_{ij}=\overset{x_q=\kappa-1}{\forall}\,q\overset{m}{\forall}\,i\overset{m}{\forall}\,j,\ F_{\text{sim}}(a_{qi},a_{qj})\ \text{ and }\ q=|C'|.$$

The quality score E_s is evaluated, just as in *Objective function 1*, by summing up the column-vector which cluster correctly with respect to *C'*, however in this case this is only done over the similarity matrices ${}^{S}_{\kappa}A$ for their respective class index κ, $C_\kappa\in C'$. In this case as well, the objective function is flexible enough to be adapted to the needs of the search, which in turn depend on the biological context at hand.

Algorithm 4: Objective function ${}^{q}\Phi_{o}$

${}^{S}A=[{}^{S}a_{ij}]_{m\times m}$ where ${}^{S}a_{ij}=\overset{x_q\neq 0}{\forall}\, q \overset{m}{\forall} i \overset{m}{\forall} j, F_{\text{sim}}(a_{qi}, a_{qj})$

$\forall \kappa\in\{1,\ldots,q\}$, ${}^{S}_{\kappa}A=[{}^{S}_{\kappa}a_{ij}]_{m\times m}$ where ${}^{S}_{\kappa}a_{ij}=\overset{x_q=\kappa-1}{\forall}\, q \overset{m}{\forall} i \overset{m}{\forall} j, F_{\text{sim}}(a_{qi}, a_{qj})$ and $q=|C'|$

Input: intensity matrix $A = [a_{ij}]_{n\times m}$, solution vector $X = [x_i]_n$, class set $C'\subseteq P(C)$
Output: quality of clusterability E_s, similarity matrix ${}^{S}A_i=[{}^{S}a_{ij}]_{m\times m}$, optimal similarity threshold T_{opt}

$E_s \leftarrow 0$
$T_{opt} \leftarrow -1$

$\forall T\in(0.01, 0.02, \ldots, 1)$
  $E_s \leftarrow 0$
  $\forall k\in(1,2,\ldots,|C'|)$
    $\forall i\in(1,2,\ldots,|c_k'|), c_k'\subset C'$
      $\forall j\in(i+1, i+2,\ldots,|c_k'|), c_k'\subset C'$
        $\overset{\kappa=k}{\forall}\,\kappa$
          if $({}^{S}_{\kappa}a_{c_k^i c_k^j}\geq T)$
            $E_s \leftarrow E_s + 1$
            $T_{opt} \leftarrow T$
          end-if
        end-for
      end-for
    end-for
  end-for
end-for

5.5 Empirical results

A series of optimisation runs with varying parametrisation were conducted in order to test the performance of the models regarding the quality factors of statistical significance, diagnostic and predictive ability. I used a *deterministic local search* variant of the *gradient descent* algorithm for most of the experiments in order to obtain a deterministic path to every solution produced by the optimisation process. This algorithm yields in most cases locally optimal solutions, which aim at showing the biological sample variation in respect to varying metabolic combinations. Global optimisation via a Genetic Algorithm and Simulated Annealing was applied in *experiment 4*. The main evaluation criterion was the clusterability of the optimised solution vectors with respect to the three known biological groups. All clustering evaluation was performed using the graph-based community structure partition of the respective co-intensity networks. The purpose of these experiments is to provide a general overview of the potential of the framework without deepening on every individual run. There are countless of ways of varying the experiments, each of which can be the subject of its own study. For the purposes of this work, I have favoured intuitive and simple ways over specialised and possibly more efficient ones, such as using deterministic local search for most of the experiments over stochastic global search, as well as using a scoring objective function over a continuous function associated to modularity optimisation.

I tested the compatibility of the results of the Crohn's study with my method's assumption by isolating the 6960 masses that were deemed discriminant by Partial Least Square regression [4] and using them to construct the co-intensity network of figure 5.2. According to my hypothesis, discriminant masses should be able to “characterise” the sample and, when processed separately, successfully cluster the data into its actual biological grouping. Nonetheless, we observe that there are hardly any modules or patterns of biological significance forming in the graph's structure, which is perhaps even less descriptive than that of the raw data. This observation alone would be enough to justify the motives of my approach, without necessarily implying that the original study's

results were erroneous. It is likely that the set of truly discriminant masses (from my method's perspective) is either a subset or at the intersection of the set of 6960 metabolites (found in [4]). The Crohn's study has, in fact, short-listed a total of 25 metabolites [4], which are regarded as the actual biomakers and have been used to validate my results.

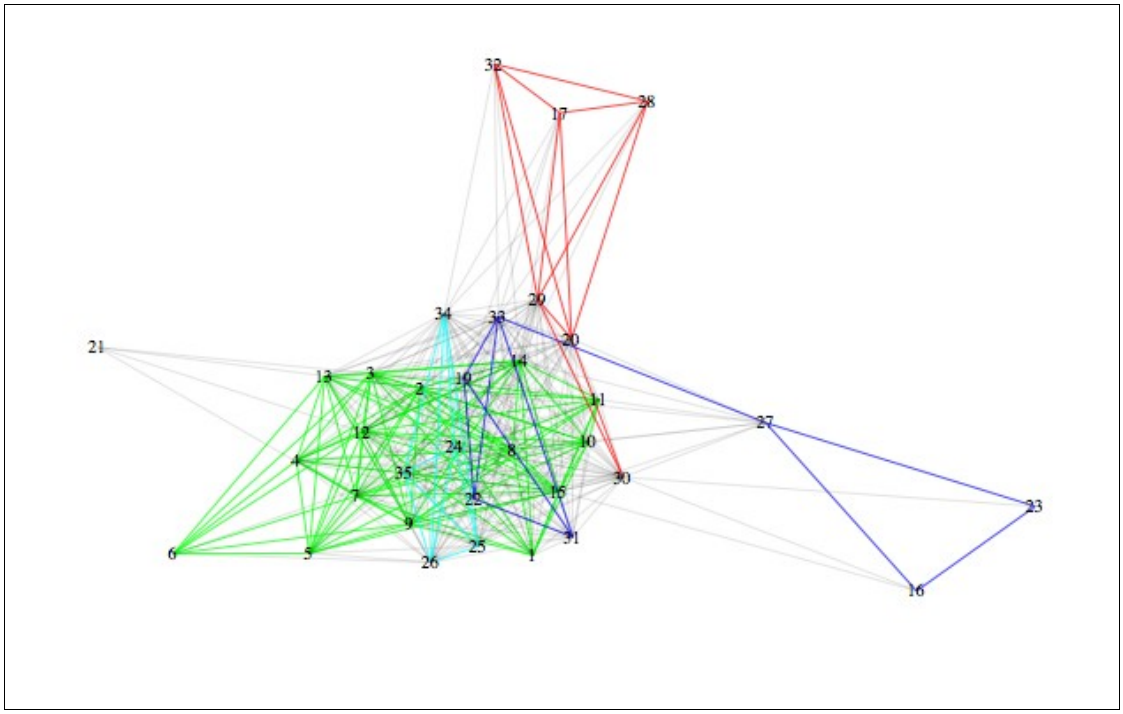

Figure 5.2: Co-intensity network created from the masses deemed to be significant by the PLS algorithm in [4]. Green, blue, and red edges link samples belonging to the same biological group, HH, CCD, ICD, respectively. No modules can be observed on the graph while the coloured patterns are less distinct than in the raw dataset.

5.5.1 Optimisation of two classes over raw dataset (model 1)

I used the deterministic local search algorithm to optimise the objective function over the raw dataset by maximising the quality of clusterability between two out of the three biological groups. The co-intensity network of the resulting solution vector can be seen in figure 5.3. Green edges in the graph imply a correctly assigned similarity relation between samples in class HH, blue edges accordingly in class ICD, and red edges in class CCD. The objective function optimised the solution vector only for classes HH and ICD, however the third biological group (CCD) forms its own distinct community in the graph, even though it was ignored during the optimisation process. The fourth biological group HD, which is theoretically linked with both ICD and HH, is represented by cyan edges and appears naturally between green and blue. This group is of secondary importance and its classification is not evaluated, however, as we shall see it almost always forms a distinct pattern in co-intensity networks even though it is never accounted in the optimisation. By looking at the community structure results we observe that 34 out of 35 samples were classified correctly using only partial supervised information; the only outlier being sample 21, which as we shall see is a constant pattern. The 6 out of 6 samples of the CCD (red) class, which were ignored during the optimisation process, are classified correctly.

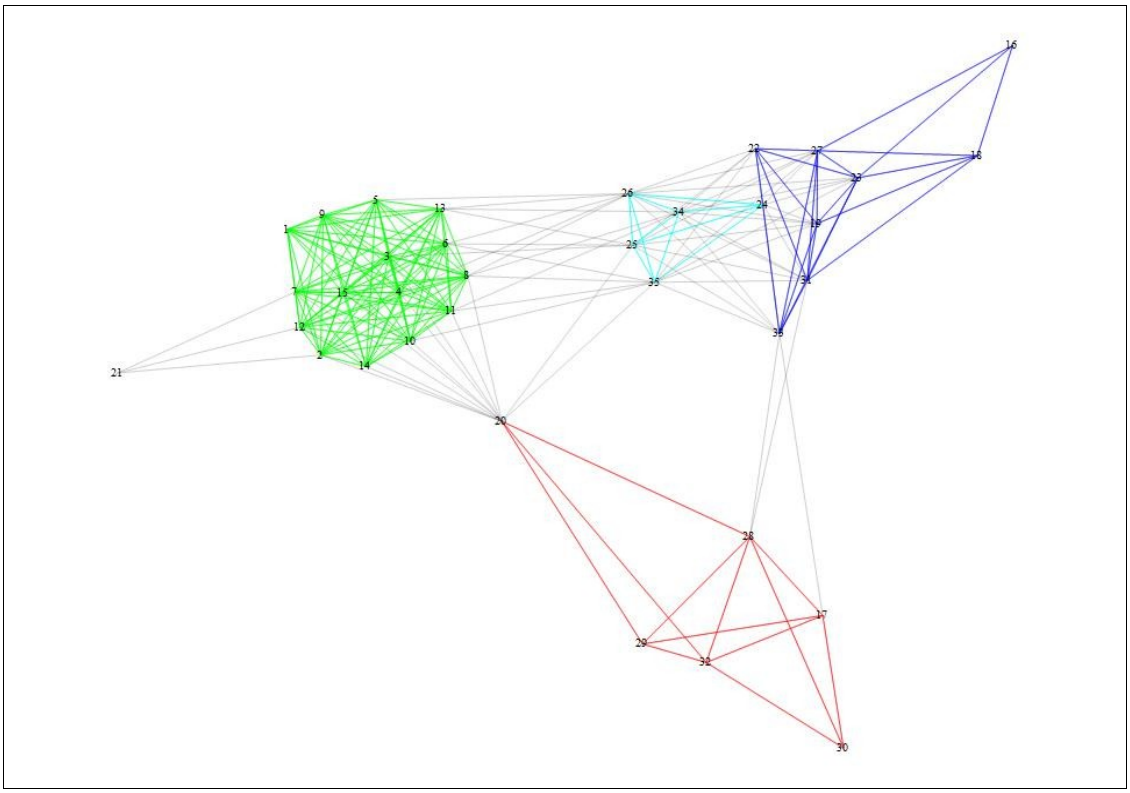

Figure 5.3: Co-intensity network produced by optimising for two out of three biological classes (2-class optimisation for HH - green and CCD - blue). The graph displays a very high clusterability with modules corresponding to the three biological groups, even though the ICD (red) class had not been considered by the objective function.

5.5.2 Optimisation over raw dataset and solution merging (model 1)

Three distinct solution vectors S_{ICD}, S_{CCD}, S_{HH}, were produced by applying three runs of local optimisation of *model 1* over the search space of the raw dataset using *objective function 1* adapted on each of the three classes (ICD, CCD, HH), for each run. The three solution vectors were unified into a single vector by merging all non-zero elements using the logical OR operator: $S_U \equiv S_{ICD} \vee S_{CCD} \vee S_{HH}$. The unified vector S_U was then used to cluster the raw dataset. The purpose of the experiment was to test the clustering potential of the unified vector independently of the clustering quality of the individual solution vectors. Since every individual solution was the product of the optimisation over only one biological group with no prior consideration of the other two, there is no guarantee that they would cluster the dataset correctly into its three classes. The assumption to be tested, however, was that the vector produced by the union of the three separately optimised solutions would integrate their information and cluster the raw data correctly, implying that an important chunk of noise was removed while the vital information on the biological groups was preserved. The target of the experiment was successful as the unified solution vector clustered the raw dataset into three classes with a classification precision of approximately 90% as seen in figure 5.4D, indicating the non-random nature of optimisation results. Interestingly enough, the individual vector solutions were also able to cluster the data with a significant precision as seen in figures 5.4A, 5.4B, 5.4C, something which reveals the diagnostic potential of a semi-supervised classification technique.

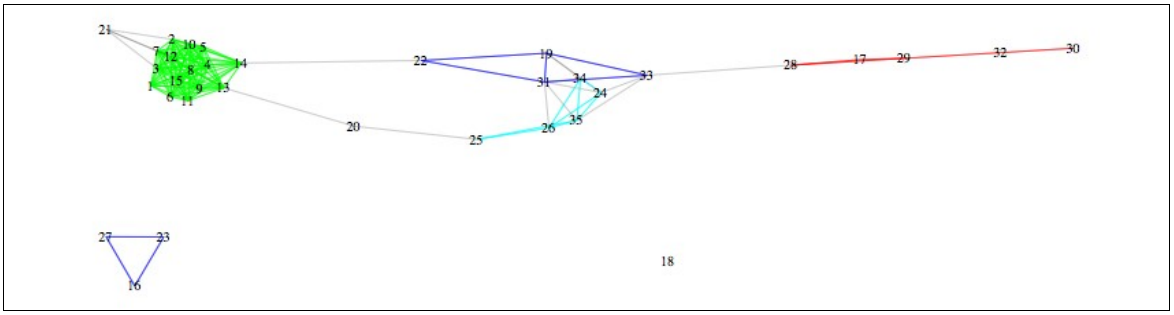

Figure 5.4A: Co-intensity network produced by solution vector optimised for one single biological group (HH - green), while we observe the other two groups (CCD - blue, ICD - red) cluster as separate graph modules.

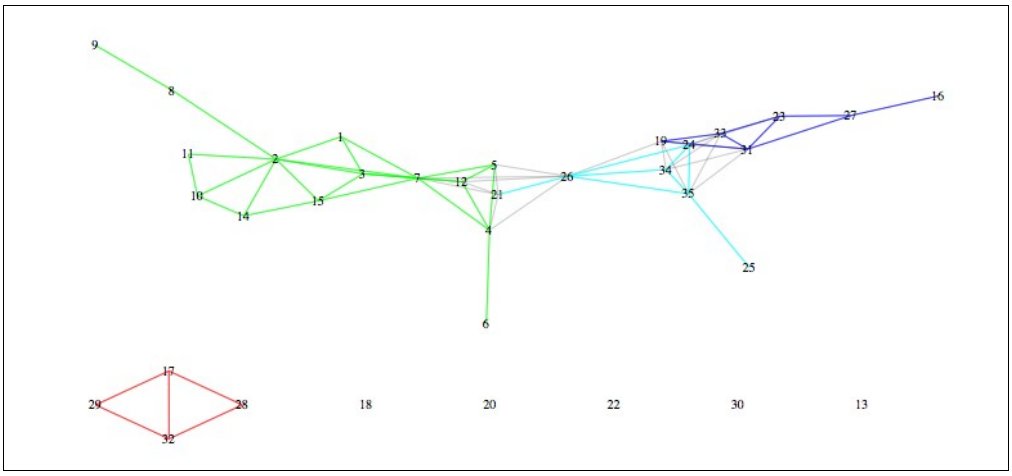

Figure 5.4B: Co-intensity network produced by solution vector optimised for one single biological group (CCD - blue). Green and red edges link samples belonging to the same biological group, HH and ICD, respectively. Graph modules largely correspond to known biological groups.

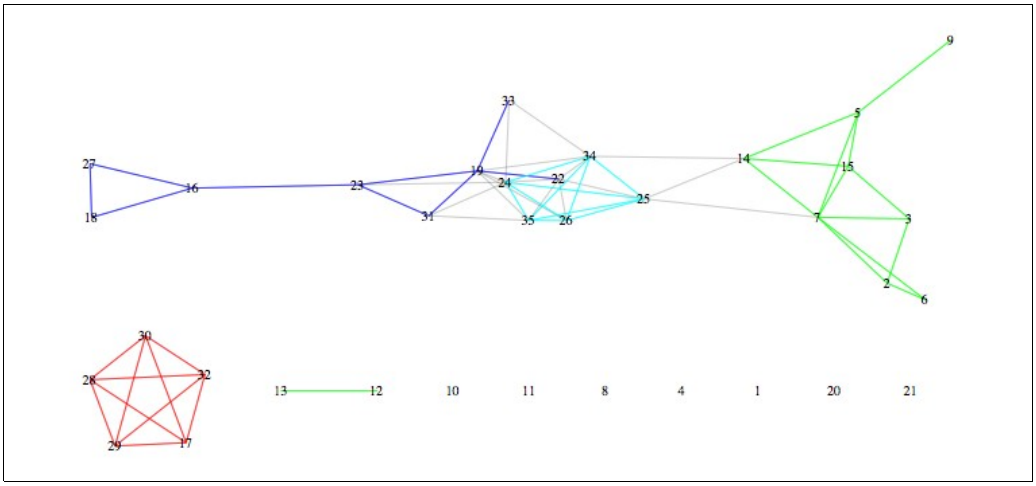

Figure 5.4C: Co-intensity network produced by solution vector optimised for one single biological group (ICD – red). Green and red edges link samples belonging to the same biological group, HH and ICD, respectively. Graph modules largely correspond to known biological groups.

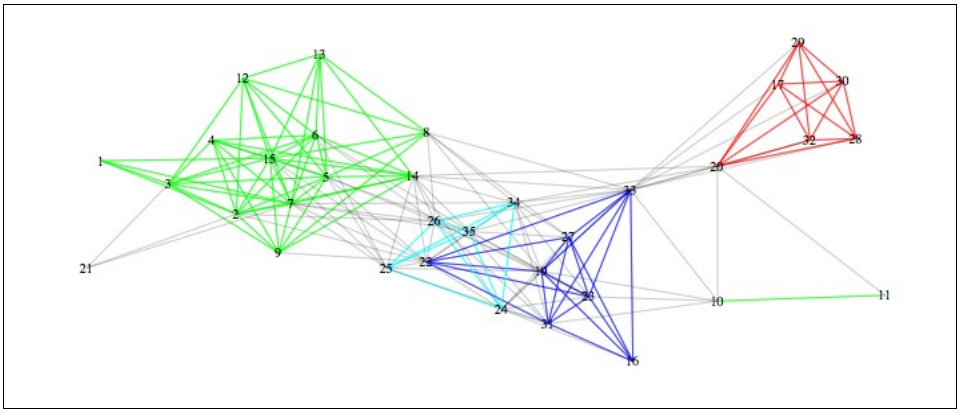

Figure 5.4D: Co-intensity network produced by merging the three binary solution vectors. The coloured biological regions correspond closely to the graph modules yielded by community structure partition. Green, blue, and red edges link samples belonging to the same biological group, HH, CCD, ICD, respectively. Graph modules largely correspond to known biological groups.

5.5.3 Optimisation over filtered dataset and solution merging (model 1)

In the second experiment the model was tested in a similar way as in 5.5.2 with the difference that the raw dataset had undergone a noise pre-treatment and reduced to 4072 out of the total 18783 masses prior to the 3-way optimisation and solution merging. The pre-treatment involves an optimisation of *model 1* over the raw dataset using all three biological groups, i.e. using the optimised solution vector to isolate the masses that maximise the clusterability of the raw data into its known biological groups. These 4072 masses were then used as the starting point for 5.5.2. As expected, the results were identical to those of 5.5.2 but much more accentuated. As it can be seen in figures 5.5A, 5.5B, and 5.5C, the biological clusterability of the individual solutions is now much more uniform and accurate at 100%, while in the case of the unified vector solution we observe an impressive partition of the co-intensity network into three disconnected subgraphs, one for each of the known biological groups (figure 5.5D). However, prediction in this case is not in the scope of the experiment since all three biological groups were used as supervised information during the initial optimisation stage for data reduction.

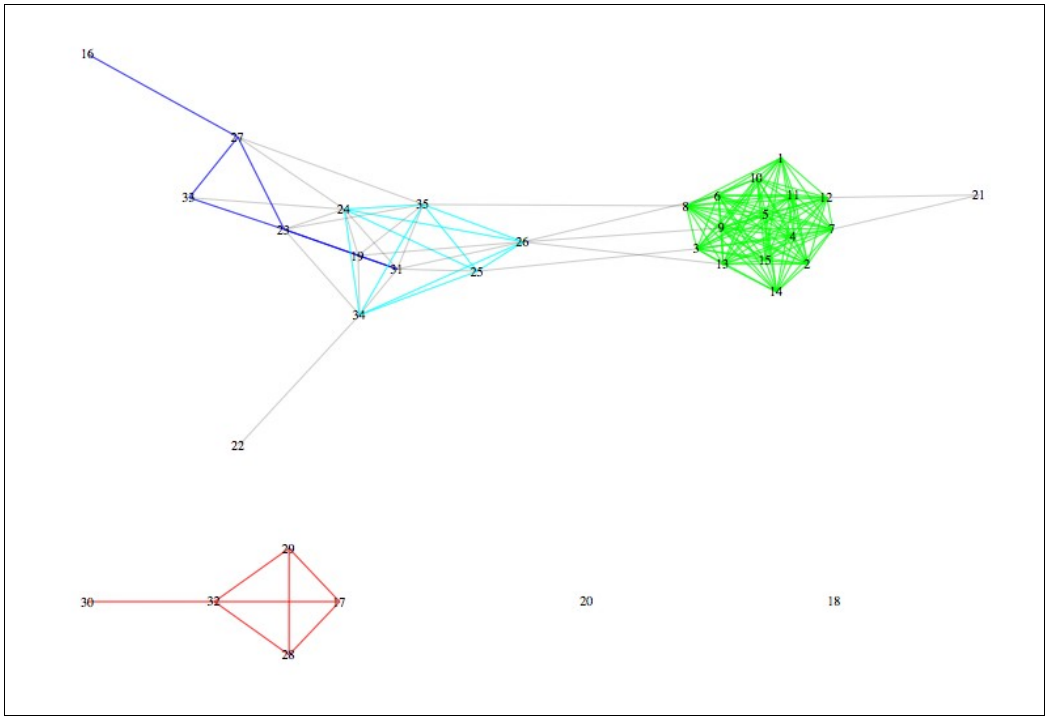

Figure 5.5A: Co-intensity network produced by solution vector optimised for one single biological group (HH – green). The graph was created by applying the binary vector on the filtered dataset. Biological groups that were not considered by the optimisation (blue and red) are clustered with high precision. Graph modules largely correspond to known biological groups.

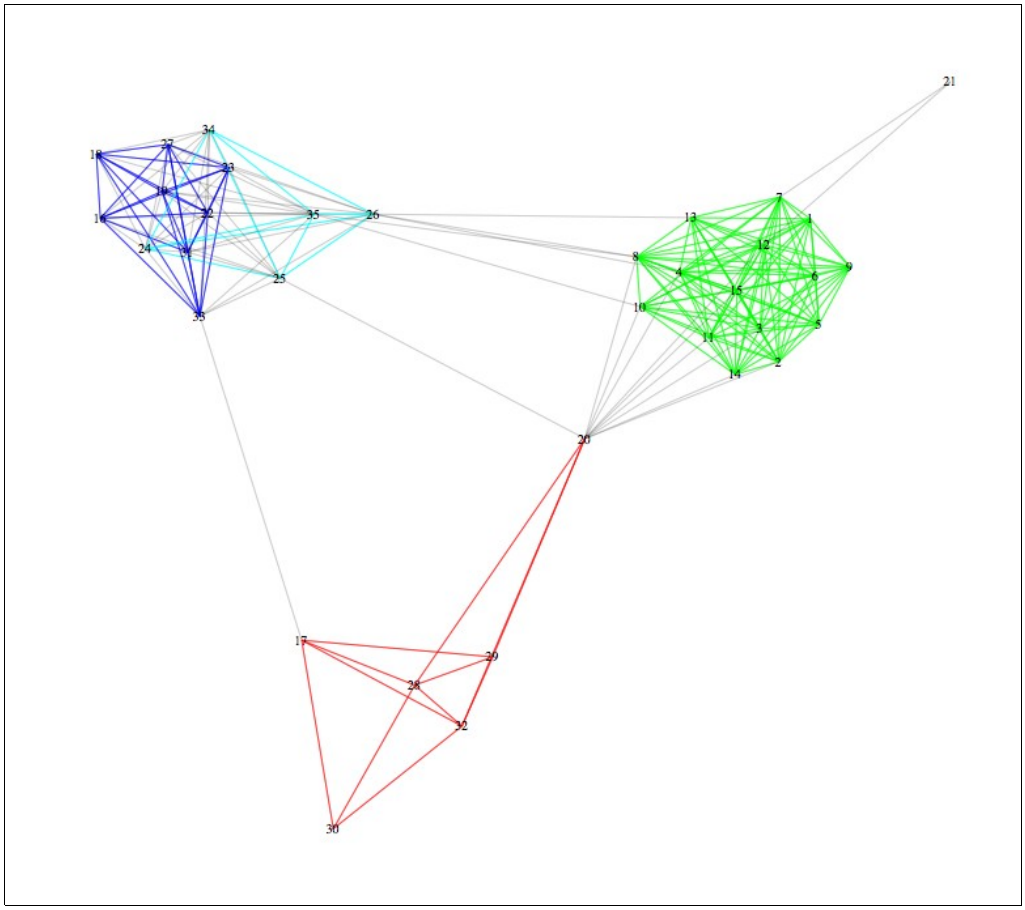

Figure 5.5B: Co-intensity network produced by solution vector optimised for one single biological group (CCD - blue). The graph was created by applying the binary vector on the filtered dataset. Biological groups that were not considered by the optimisation (green and red) are clustered with high precision. Graph modules strongly correspond to known biological groups.

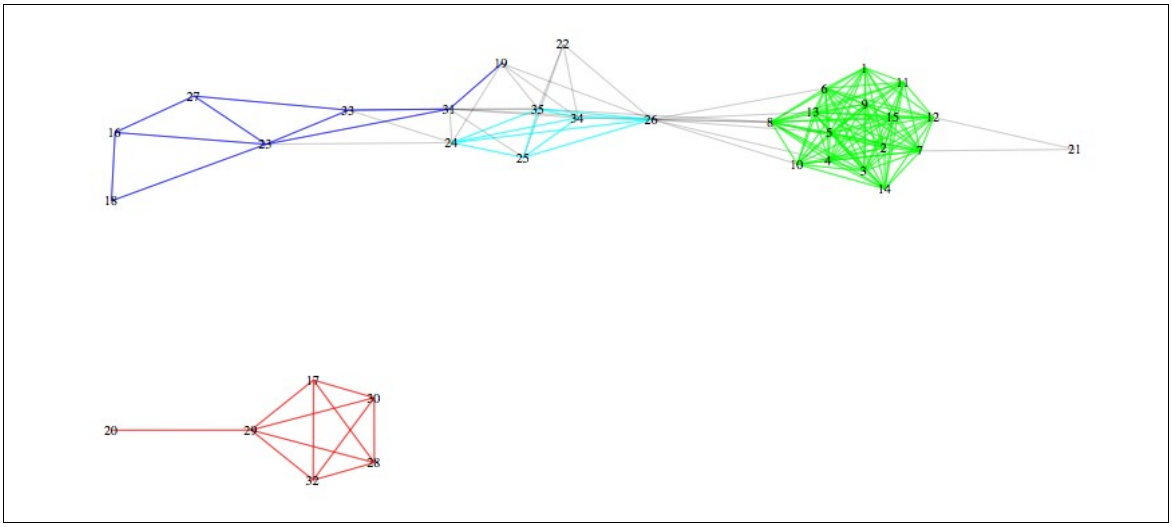

Figure 5.5C: Co-intensity network produced by solution vector optimised for one single biological group (ICD - red). The graph was created by applying the binary vector on the filtered dataset. Biological groups that were not considered by the optimisation (blue and green) are clustered with high precision. Graph modules strongly correspond to known biological groups.

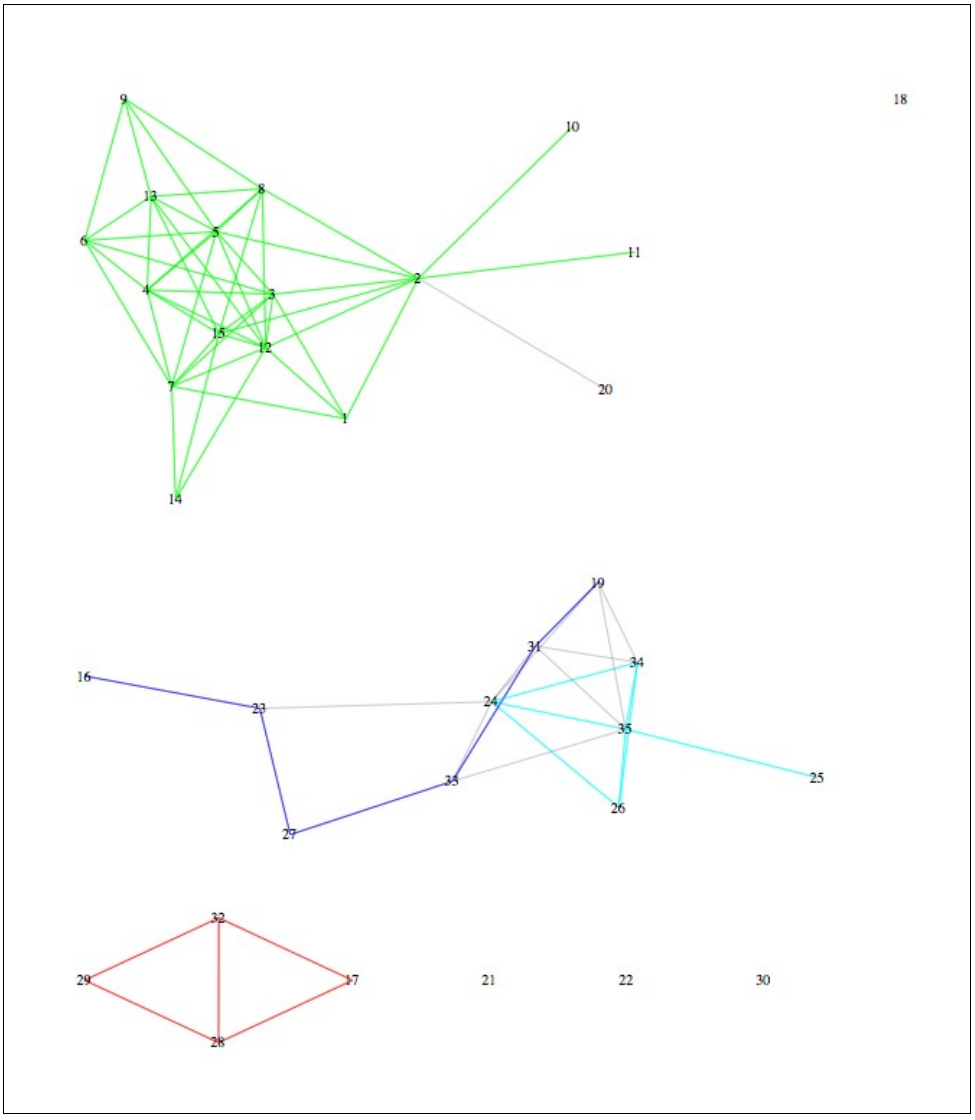

Figure 5.5D: Co-intensity network produced by merging the three binary solution vectors optimised on the filtered dataset. The three biological groups form three distinct disconnected subgraphs. Graph modules largely correspond to known biological groups.

5.5.4 Constrained optimisation: Base-q model and metabolite identification (model 2)

Model 1 achieves noise removal and data reduction by isolating the data points which are necessary in order to maximise the quality of clusterability of the co-intensity network over the raw dataset. *Model 2* goes one step further by assigning a class to every data point, i.e. associating all discriminant masses to a biological group. Naturally, the ultimate goal of this model is biomarker identification. Both deterministic and stochastic meta-heuristics were used to discover good solutions over several runs, two of which are depicted in figures 5.6A and 5.6 (B). Depending on the objective function, solutions yielded by different algorithms using different starting points were found to be highly similar and small in number (i.e. very few metabolites were identified as discriminant). The way to test the validity of those results would be to compare the metabolites of these solutions to the short-listed biomarkers of [4]. Interestingly, 2 out of 25 short-listed biomarkers were included in the 28 mass sized solution yielded by the gradient descent algorithm. We can perform a hypothesis test to find the statistical significance of this result and the probability to occur at random.

Let N = 18480 be the total number of elements, m = 25 be the labeled elements (the ones discovered in the Crohn's study), p = 28 be the number of elements picked out be combinatorial learning, and $k \in K$ be the elements of p at the intersection with m with $k_{obs} = 2$ being the instance of k observed in given experiment.
The null hypothesis H_0 is that combinatorial learning picked out p = 28 out of a total N = 18480 elements, at random. The alternative hypothesis H_1 would be that combinatorial learning specifically chose the m = 25 labeled elements. Consequently, we deduct that a larger observed k_{obs} would provide stronger evidence to support H_1.
Under the null hypothesis, K follows a *hypergeometric distribution* with probability mass function (pmf):

$$Pr\{K=k\} = \frac{\binom{m}{k}\binom{N-m}{p-k}}{\binom{N}{p}}, \quad k = 0,1,\ldots,min\{m,\, p\}$$

$$Pr\{K=2\} = \frac{\binom{25}{2}\binom{18455}{26}}{\binom{18480}{28}} \approx 0.00064$$

The *p-value* of the test (Fisher's exact test) is the probability of observing the test statistic to be at least as extreme as the observed k_{obs} under the null hypothesis.
Hence, the *p-value* is given by the tail probability:

$$Pr\{K \geqslant k_{obs}\} = \sum_{k=k_{obs}}^{min\{m,\, p\}} \frac{\binom{m}{k}\binom{N-m}{p-k}}{\binom{N}{p}}$$

$$Pr\{K \geqslant 2\} = \sum_{k=2}^{25} \frac{\binom{25}{k}\binom{18455}{28-k}}{\binom{18480}{28}} \approx 0.00065$$

Therefore, the probability of this event occurring at random is about 0.0006 and the null hypothesis is largely rejected at the 1% significance level.

It is noteworthy that the gradient descent algorithm converges always to a near identical solution using varying starting point vectors. More precisely, the above-described result was obtained by using as a starting point a binary vector whose fields were all initialised to '1' (full vector). Then, A solution of 40 discriminant masses was obtained by using as a starting point a binary vector whose fields were all initialised to '0' (empty vector). It was observed that these 40 masses contained, also, the same 2 metabolites that were detected by the earlier “full vector” run (masses 150.0560175 and 299.2591449). For k_{obs} = 2 and p = 40, the statistical significance of this result comes with a p-value of approximately

0.0013. The fact that we obtain such similar results when we use so distant starting points leads us to believe that the gradient descent algorithm converges to a local optimum of valuable biomarker information. I tried out several additional runs using my stochastic search algorithms in order to discover other similar optima which possibly hinted the discovery of different biomarker. The co-intensity network of this gradient descent solution can be seen in figure 5.6A. A stochastic optimisation run using the simulated annealing algorithm yielded a solution of 5 masses, 1 out of which was among the 25 short-listed metabolites of [4], albeit a different one than in the gradient descent solution (mass 407.2802529). Using the same significance test, the probability of this result to occur by chance is approximately 0.0067.

Overall, several different combinations of the gradient descent, simulated annealing and genetic algorithms yielded solutions in which 6 out of 25 short-listed metabolites were detected. Solution size varied from 16 to 40 metabolites and almost each solution vector contained one of those 6 metabolites present in the biomarker list of [4]: 407.2802529, 299.2591449, 329.2333294, 243.1714022, 447.3115566, 150.0560175, 297.1132029, 403.1510233. These results reveal that my model has a strong biomarker identification potential.

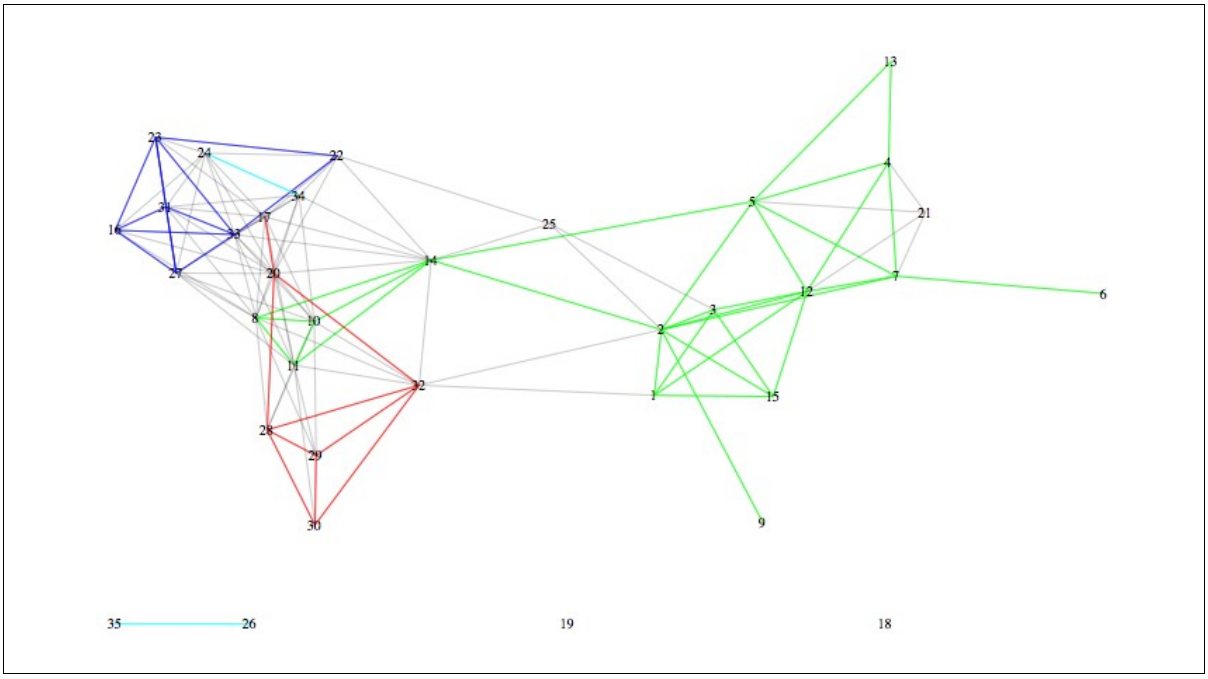

Figure 5.6A: Co-intensity network produced through constrained optimisation (model 2) with local search for two classes only (HH - green and ICD - blue). The three biological classes form coloured patterns to which discriminant metabolites are associated. Graph modules strongly correspond to known biological groups.

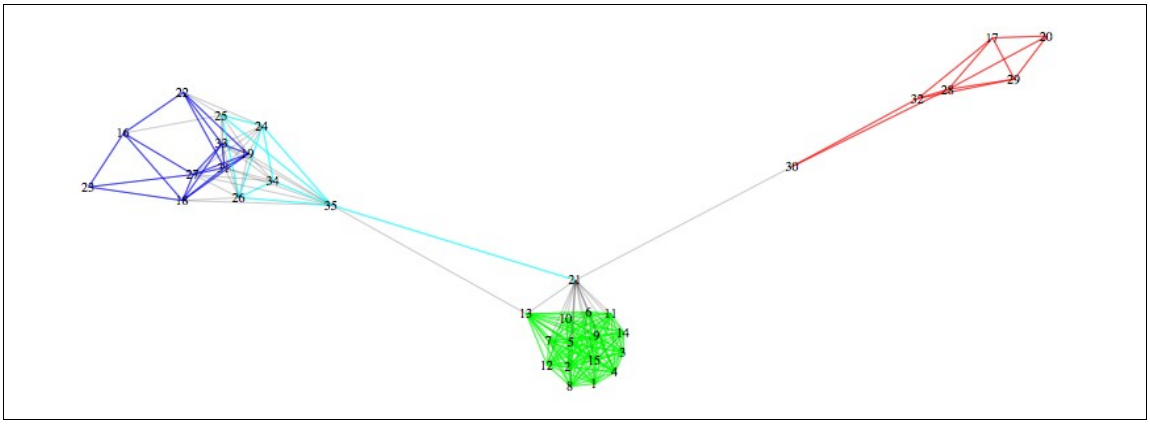

Figure 5.6B: Co-intensity network produced through constrained optimisation (model 2) with local search for all three classes. The three biological classes are clustered with precision via the graph's community structure partition. Discriminant metabolites are associated to the three graph modules.

5.5.5 Supervised and semi-supervised learning experiment

We performed the typical supervised classification procedure for cross-validation of dividing the raw dataset into 30% for training and 70% for testing over the 35 columns of the mass-sample matrix. Model 1 was optimised over the the training set in order to produce a trained solution vector of 1090/18783 and 555/18783 for the 3-class and 2-class scenarios, respectively. This trained vector, which can be regarded as a predictor variable, was applied on all training, test, and full data sets, in order to filter the least important masses before the construction of a co-intensity network. Figure 5.7A shows the application of the trained vector on the training set of the 3-class scenario. As expected, 3 disconnected subgraphs are forming flawlessly, indicating that the training on the 30% of the data was successful. Figure 5.7B displays the co-intensity network resulting from the application of the same trained vector on the test set, i.e. 70% of the raw data. We observe the formation of three distinct modules and a derived sample classification of 100% biological accuracy. Naturally, the same classification accuracy is observed in the network of figure 5.7C, which was created by the trained vector's application on 100% of the data. Since trained data is included for classification, network clusterability is higher while biological groups are forming very distinct modules and disconnected subgraphs (though classifying this dataset is of minor predictive interest). The same procedure was followed for the 2-class scenario, where the predictor was trained over two classes (HH, ICD) and then applied on the training, test, and full data sets alike. The resulting co-intensity networks are displayed in figures 5.8A, 5.8B, and 5.8C, respectively. It is noteworthy that an overall high classification accuracy and clusterability is observed. An almost flawless classification is performed over the 70% and 100% data sets (figures 5.8B, 5.8C) while we observe the modular formation of the class of samples that was ignored in training (blue module). Even more remarkably, in the case of the predictor created by semi-supervised learning over one single class (ICD-red, figure 5.9A), we observe a flawless classification in the 70% test set, with all untrained classes (HH-green, CCD-blue) appearing clearly as distinct modules (figure 5.9B). Surprisingly, the modules of the untrained classes (blue and green) appear to be much more densely connected than the trained one (red). The less important classification on

the full dataset (figure 5.9C), is the one with the highest number of misclassified objects, though still with a very good performance considering the amount of information that was used in training.

Different objective functions were tried out, each optimising for a different number of biological groups. Despite the dataset's small size (only 35 samples), the classification of the test set was correct even in the case of the semi-supervised training over class ICD. Other objective functions yielded also satisfactory results. In order to associate the known biological groups of the training set with the community structure (graph-based clusters) of the test set, one can examine the graph-theoretical properties of the corresponding subgraphs between the two phases of the machine learning process. Specifically, I noticed that the subgraph of the module with the highest average degree in the training set's graph corresponds to the subgraph of the highest average degree in the test set's graph (in this case the HH module). Similarly, the subgraphs of the modules with the second higher average degrees in the training set's graph correspond to the subgraphs of the second higher average degree in the test set's graph (ICD and CCD). CCD in particular does not at all appear in the training set's graph, it is however classified successfully in the test set. That implies that the information of this class was stored in the predictor even though it was not included as an optimisation criterion within the objective function during the training process.

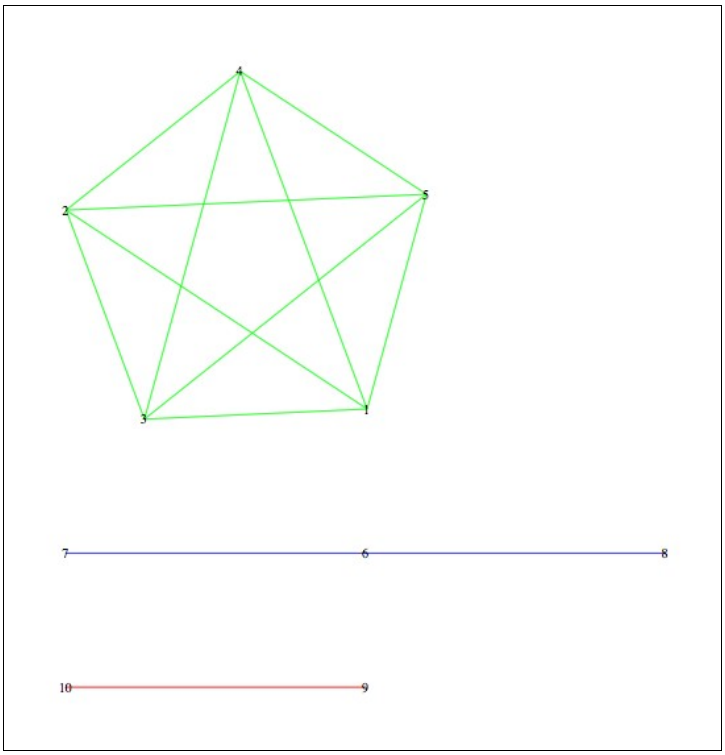

Figure 5.7A: Co-intensity network produced by applying predictor (binary solution vector) on the training set (30% of the raw dataset). The predictor was produced through mass difference optimisation on the training set for all three classes. Green, blue, and red edges link samples belonging to the same biological group, HH, CCD, ICD, respectively.

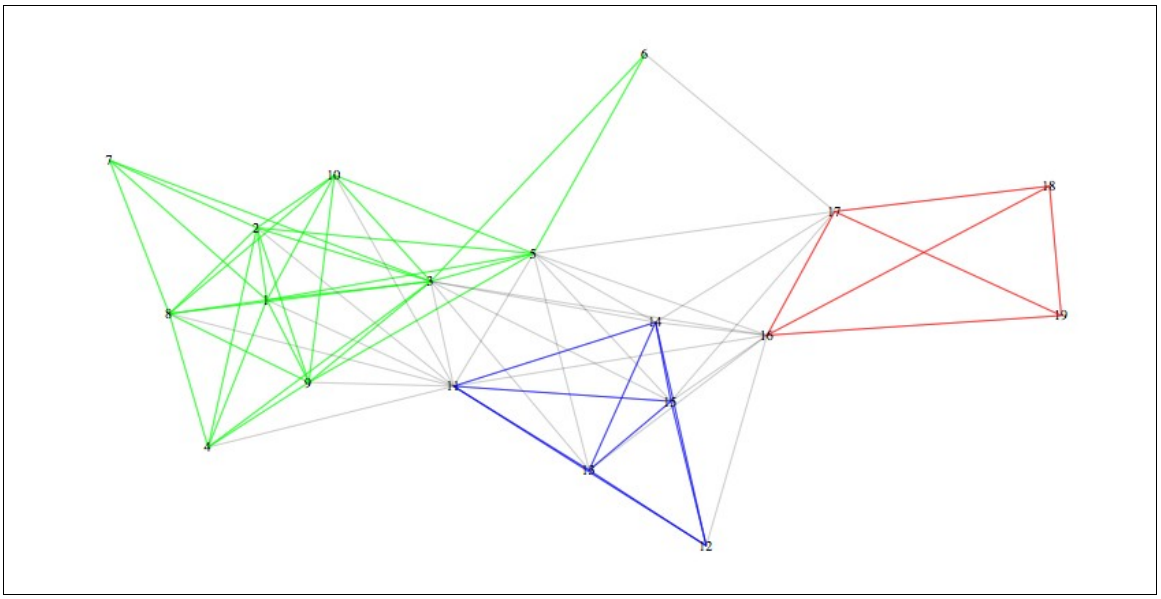

Figure 5.7B: Co-intensity network produced by applying 3-class predictor on test set (70% of the raw dataset)**.** *Green, blue, and red edges link samples belonging to the same biological group, HH, CCD, ICD, respectively.*

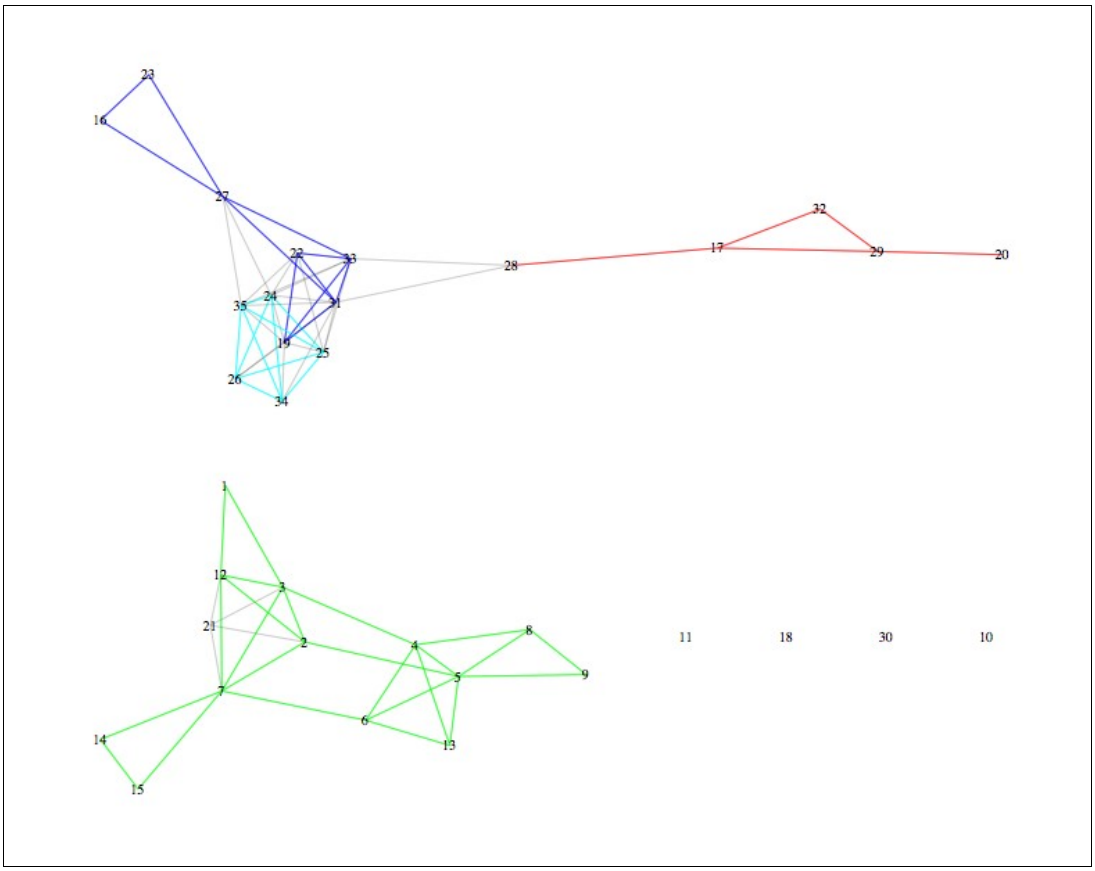

Figure 5.7C: Co-intensity network produced by applying 3-class predictor on full dataset. Green, blue, and red edges link samples belonging to the same biological group, HH, CCD, ICD, respectively. Graph modules strongly correspond to known biological groups, showing high predictive capability.

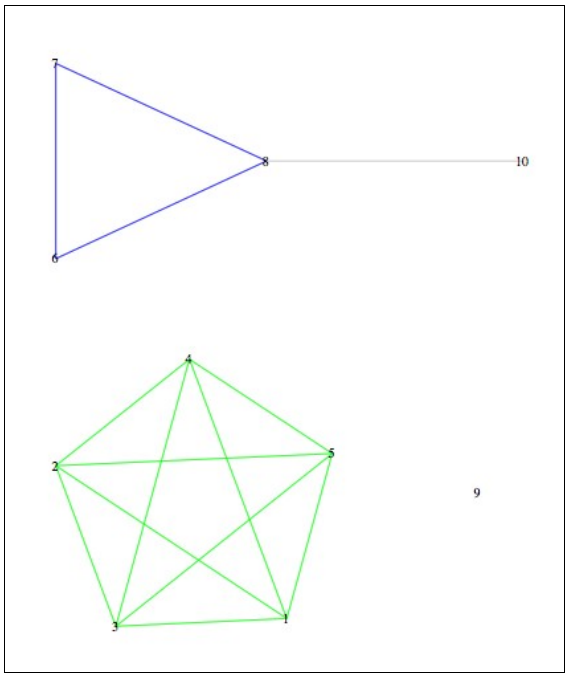

Figure 5.8A: Co-intensity network produced by applying predictor (binary solution vector) on the training set (30% of the raw dataset). The predictor was produced through mass difference optimisation on the training set for two out of three classes (green and blue).

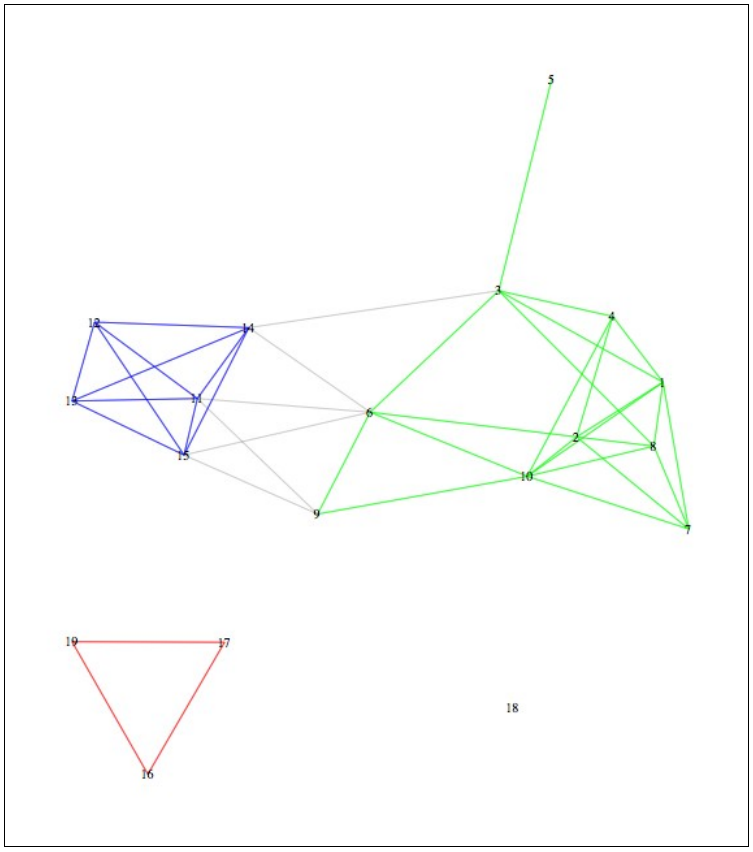

Figure 5.8B: Co-intensity network produced by applying 2-class predictor on test set (70% of the raw dataset). Green, blue, and red edges link samples belonging to the same biological group, HH, CCD, ICD, respectively. Graph modules largely correspond to known biological groups, showing moderate predictive capability.

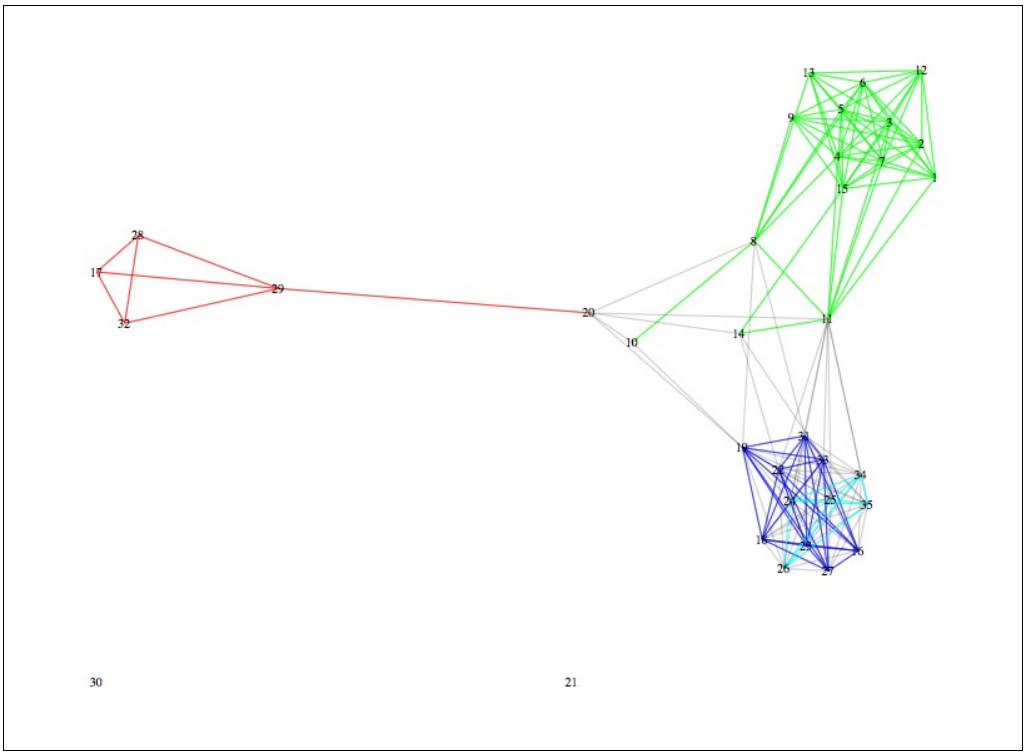

Figure 5.8C: Co-intensity network produced by applying 2-class predictor on full dataset. Green, blue, and red edges link samples belonging to the same biological group, HH, CCD, ICD, respectively. Graph modules strongly correspond to known biological groups, showing high predictive capability.

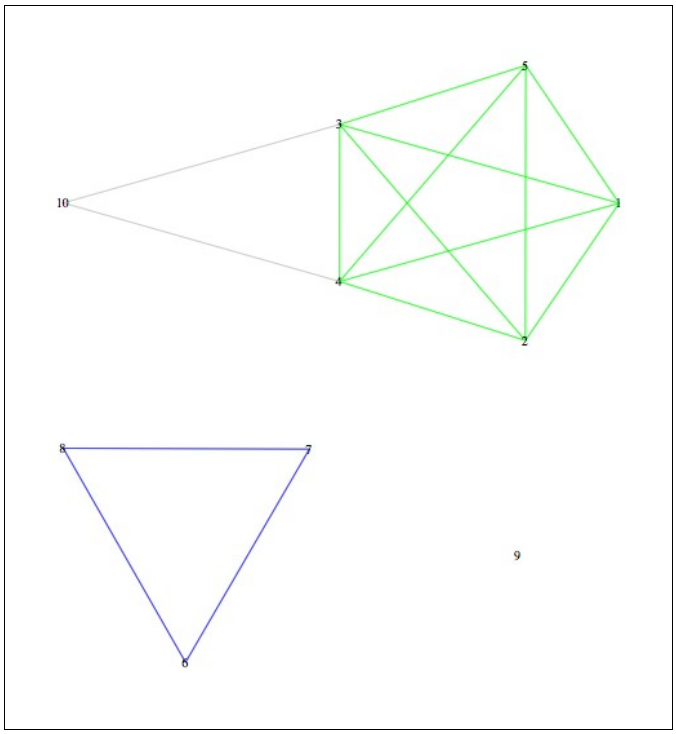

Figure 5.9A: Co-intensity network produced by applying predictor (binary solution vector) on the training set (30% of the raw dataset). The predictor was produced through mass difference optimisation on the training set for one out of three classes (red - ICD).

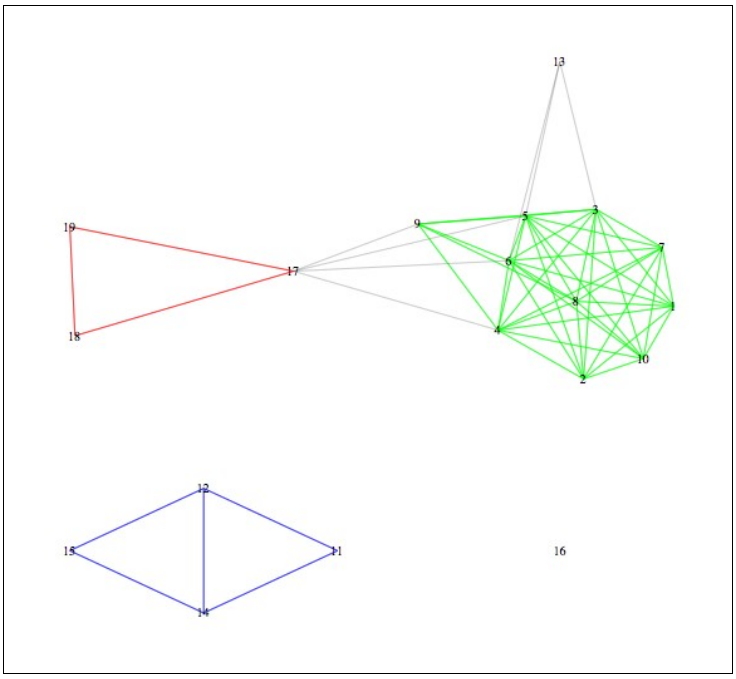

Figure 5.9B: Co-intensity network produced by applying 1-class predictor on test set (70% of the raw dataset). Green, blue, and red edges link samples belonging to the same biological group, HH, CCD, ICD, respectively. Graph modules largely correspond to known biological groups, showing high moderate capability.

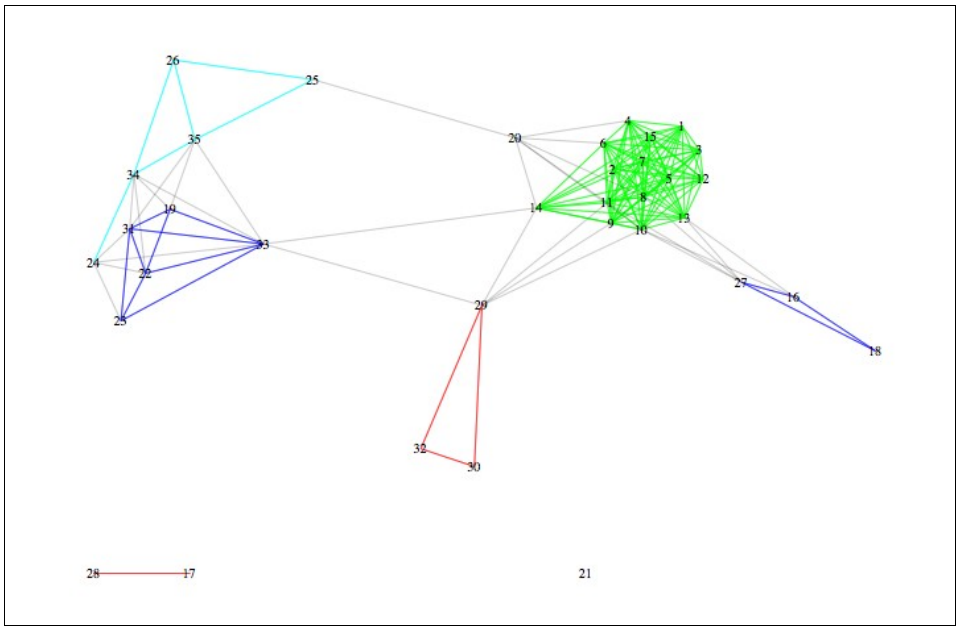

Figure 5.9C: Co-intensity network produced by applying 1-class predictor on the whole dataset. Green, blue, and red edges link samples belonging to the same biological group, HH, CCD, ICD, respectively. Graph modules largely correspond to known biological groups, showing moderate predictive capability.

5.6 Case study: Insulin resistance data

In this section, I put to a further test the methods presented in this chapter by applying *model-2* optimisation on the insulin resistance data; a ICR-FT-MS metabolomics dataset of very high complexity [75]. Any unsupervised classification algorithm applied on this dataset would fail to detect any sort of clusters or patterns. I applied my own semi-supervised models in order to see how they perform in a more complex scenario.

5.6.1 Experimental background

The study behind this dataset concerns the relation between insulin sensitivity and diabetes in the context of ICR-FR-MS non-targeted metabolomics [75]. Decline in insulin sensitivity measured by the Matsuda formula [76] implies a high risk of developing type 2 diabetes. The article in question uses multivariate statistics in order to establish a threshold on the Matsuda index values between the regions of high and low risk [75]. A total of 46 plasma samples from non-diabetic subjects exhibiting high to low sensitivities were analysed by means of ICR-FT-MS, producing 12413 metabolites [75].

5.6.2 Empirical results

Each sample in the Insulin dataset is associated to an ESI Matsuda value, ranging from 2.48 to 41.47. In [75], samples have been divided into three classes depending on the three selected sub-ranges for low, intermediate, and high Matsuda values. The class of high values is regarded as the “risk” region ($\geq$15) while the class of low values as the “non-risk” (<8.5). Red, blue, and green edges, represent the connections between samples of the same classes, i.e. red, green, and blue, for high (risk), intermediate, and low (non-risk) regions, respectively.

Figure 5.10 shows how an efficient unsupervised algorithm such as *co-intensity network clustering* fails to classify the dataset since no distinct graph modules and hardly any visual patterns can be detected in the network. I applied a model-2 type of optimisation

over the raw Insulin dataset by considering only two out of the three biological groups (high and low ESI Matsuda values). As it is seen in 5.11, the high and low (red and blue) classes are formed very distinctively as separate graph modules, while the intermediate class (green) appears remarkably right in the middle of the two, testifying for the biological significance of the results.

A similar treatment was applied using model-2 constrained optimisation over the raw dataset. In the co-intensity network of 5.12, we observe a less well formed clusterability which, nonetheless, follows the same pattern of the two biological groups appearing as separate modules with the intermediate group being in the middle of the two. The solution vector leading to this result contains 74 out of the total 12413 metabolites. Due to a lack of information on the short-listed metabolites of the concerned study [75], I was not able to calculate the statistical significance of those results as I did in the Crohn's disease dataset of the previous section.

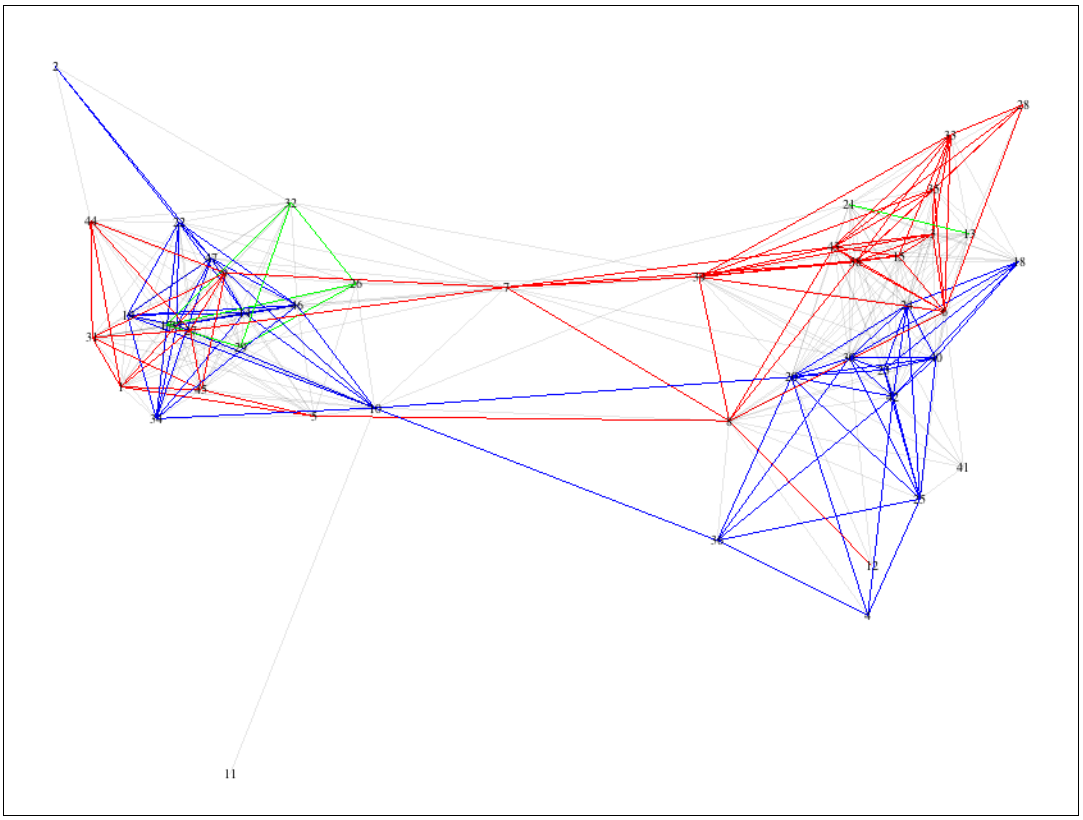

Figure 5.10: Co-intensity network created out of the raw Insulin resistance dataset (non-optimised solution: 12413/12413 masses). Red and blue coloured edges link samples belonging to the same biological group, high (risk) and low (non-risk) ESI Matsuda values, respectively. Green edges represent the intermediate zone of ESI values. The graph shows only scarce patterns but yields no biological clusters.

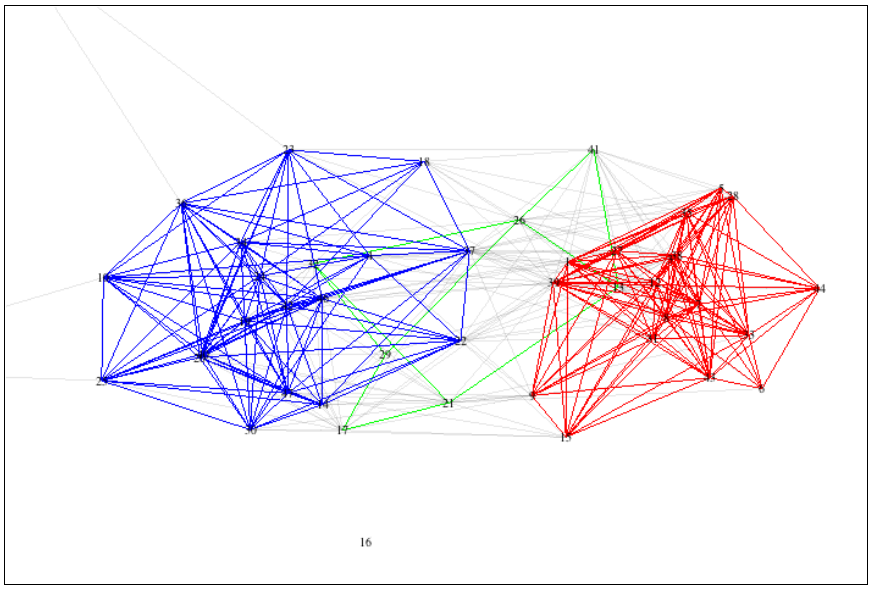

Figure 5.11: Co-intensity network (981/12413 masses) produced by optimising for two out of three biological classes (2-class optimisation for high and low ESI values). The graph depicts very high clusterability with modules that correspond to the two main biological groups (blue and red). The intermediate group (green), which was not considered in the optimisation, can be observed appearing naturally at the intersection of the two modules, reflecting an important biological pertinence.

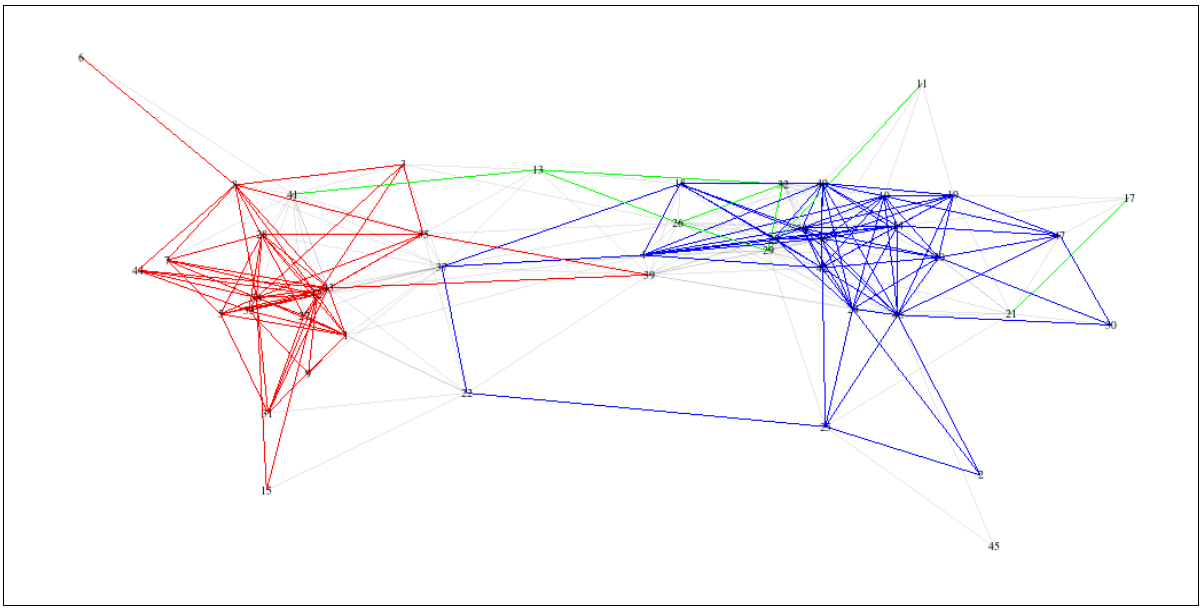

Figure 5.12: Co-intensity network produced through constrained optimisation (model 2) with local search for the two main classes only (red and blue). The three biological classes form coloured patterns and modules to which 74 discriminant metabolites are associated. Graph modules strongly correspond to known biological groups.

5.7 Conclusion

In this chapter I presented the theory and application of a novel computational framework that can be used as an alternative to standard multivariate statistical approaches for the purposes of data classification and biomarker identification in ICR-FT-MS metabolomics. The framework models a biological scenario in the form of a combinatorial optimisation problem and solves it by means of metaheuristic search. I developed principally two mathematical models for classification and discriminant signal selection, both of which use biological clusterability as the fitness criterion of the optimisation process. My measure of clusterability is calculated via co-intensity network clustering.

Model 1 represents the problem in the form of a binary vector which, once trained, can be used as:

i. a semi-supervised classifier of unlabelled data points (samples) in the same dataset (i.e. the train set used in optimisation),

ii. a semi-supervised classifier for the non-targetted discovery of unknown biological groups in the same dataset,

iii. a predictor for supervised learning and classification in unknown datasets.

The application and cross-validation of the method yielded positive and biologically pertinent results on two distinct datasets.

Model 2 represents the problem in the form of a base-*n* vector (for *n* classes) which, after training, leads to discriminant signal (biomarker) identification as well as association to one of the know biological groups. Its results are directly comparable to the ones of PLS-regression and were deemed statistically significant after hypothesis testing.

CHAPTER VI

An adapted combinatorial learning model for the study of discriminant chemical reactions in mass difference networks

This chapter deals with the last main topic of this work, which happens to be the link between the two previous main topics, namely the quantitative methods of *mass difference networks* and *combinatorial learning*. I describe an enhanced combinatorial learning model which takes into account mass difference information and yields a new indicator of biological interest in mass spectrometry bioinformatics and metabolomics.

6.1 Abstract

The use of Fourier Transform Ion Cyclotron Resonance Mass Spectrometry (ICR-FT-MS) in non-targeted metabolomics offers ultra-high mass accuracy whose full potential can be explored computationally through the mass difference network model and the Netcalc method [1][2]. As described in a different section of this work, one of the main objectives in non-targeted metabolomics is the identification of discriminant signals and potential biomarkers. A novel quantitative and computational framework that combines machine learning and operational research was suggested in chapter V as an alternative approach to this task. To date, all computational and statistical analysis on ICR-FT-MS data is focused on the instrument's direct output (namely exact masses and intensity values), however, the application of the Netcalc method has shown the importance of

experimental mass difference information in structural networks. The quantitative advances described in this thesis give rise to the question of whether - in addition to the detection of discriminant metabolite masses - there is a need for the detection of *discriminant exact mass differences*. I developed yet another approach in order to deal with this new question that I refer to as the *discriminant mass difference problem*. The approach is a merge of the methods developed during this thesis and described in earlier chapters of this manuscript, notably mass-different networks and combinatorial learning (chapters III,V). I refer to this approach as the *mass difference optimisation model*.

6.2 Introduction

Mass differences can reflect structural, stoichiometric, and biochemical information in a system. Metabolic pathways constitute a specific sequence of reactions which (a) transforms one or many substrates into one or many products, (b) delivers ATP and reduction-equivalents for energy production, (c) stores energy in structures in the form of fatty acids. Metabolic pathways enable an organism, organ, or cell to respond to external stimuli (e.g. a disease, toxins, a change in nutrition) in an organized fashion. When two different phenotypic groups (e.g. healthy and unhealthy specimens) react in a specific but different manner to the same stimulus, the magnitude and direction of mass flux through metabolic pathways might be altered. A consequent change in metabolite patterns might result to a change in stoichiometric (i.e. mass differential) patterns, indicating that a mass spectral data filter, which omits data that has not been associated to a phenotype specific subset of mass differences, should result in a set of m/z values that are more likely to be discriminative in that given context. Motivated by this hypothesis, I devised a method which leads to the extraction of such relevant mass differences from a biochemical system in the form of a structural mass difference network.

Known techniques, such as ‘gene set enrichment analysis’, ‘pathway enrichment analysis’ or the in-house developed adaptation ‘mass difference enrichment analysis’, indicate that an optimized set of mass differences might improve biological data discrimination. Based on my pre-defined measure of biological clusterability, the mass

difference optimisation approach searches for the most discriminant reactions in a biochemical system. This is achieved by combining the structural network approach of chapter III with the combinatorial optimisation framework of chapter V. The transformation list used for the mass difference network reconstruction consists of 175 mass differences selected from KEGG reaction entries, which are incident to the four most important metabolites/coenzymes denoted in [37] : Coenzyme A, Pyruvate, Glycine, Glutamate. In order to improve Netcalc coverage, the entire homologous series from C2 to C16 fatty acids and from C2 to C10 dicarboxylic acids were added. The setup of this mass difference list intends to be as close as possible to the metabolome.

6.3 Methods, models, and algorithms

6.3.1 Method overview

In order to detect discriminant mass differences, I have combined structural mass difference networking with the combinatorial learning framework for biomarker identification. The idea is to use the framework of chapter VI in combination with a mass difference network (chapter III) in order to detect the mass differences that "characterise" the sample in question. The biological assumption, therefore, is that there exists such a group of *discriminant mass differences* and the associated computational problem consists of obtaining them. The key to this approach is the introduction of a new binary vector which models an arbitrary problem solution in respect to the transformation list used in the mass difference network reconstruction.

6.3.2 Combinatorial problem modelling

Model 3: Mass difference solution vector (binary encoding)

In an $n \times m$ *mass-sample intensity matrix* $A = [a_{ij}]$ (as described in section 5.4.1), we associate a vector Z of size n, representing the list of n exact masses that correspond to the row-vectors of A. We apply *Algorithm 1* on vectors Z and W to output a mass difference network $G = \langle I, J, K \rangle$ (section 3.2.1), where W represents the list of k pre-

chosen chemical transformations and the tuple <*I, J, K*> is the sparse information output. At this state, we can define the new model vector ${}^{md}S = [{}^{md}s_i]_k$ where $s_i \in \{0,1\}, \forall\, i$, representing a feasible solution to the mass difference optimisation problem. The size of this *mass difference solution vector* ${}^{md}S$ is equal to *k*, i.e. the number of mass differences in the transformation list, which is the size of vector *W*. Every mass difference vector solution ${}^{md}S$ is associated to its corresponding mass difference network *G* = <*I, J, K*> by using the nonzero values of ${}^{md}S$ in order to filter the indices of nodes in <*I, J*>.

6.3.2 Problem resolution

In order to use the *objective function 1:* Φ_o (section 5.4.5), we need to convert the mass difference solution vector ${}^{md}S$ to a *model 1* type of mass-solution vector *S*. Function Φ_o, from the classical optimisation framework, will quantify the clustering of sampled objects using a co-intensity network produced from a *model 1* vector as described in section 5.4.4, *Algorithm 3*. Therefore, the principal difference between the MD-optimisation model and the classical metabolic optimisation framework (chapter V), is that the former applies optimisation on the mass difference solution vector that represents the absence or presence of entries in a transformation list at a given state (modelled as binary information). The input to the objective function Φ_o is a *model 1* vector, therefore, the MD-solution vector ${}^{md}S$ has first to be converted into a mass-vector before being passed as an argument to Φ_o. The *model 1* solution is generated by assigning the value of one only to the bits whose masses are participating in the structural mass difference network deriving from the MD-solution vector ${}^{md}S$ in combination with the known <*I, J, K*> information produced by *algorithm 1*. The optimisation process maximises:

$$\Phi_o\left(f_{m\leftarrow md}\left({}^{md}S\right), < I, J, K >\right)$$

where $f_{m\leftarrow md}$ is a function which converts an MD-solution vector to a *model 1* vector. The exhaustive search space is 2^k, calculated by the binary variable domain (value 2) to the power of *k* (number of bits in ${}^{md}S$). The metaheuristic algorithm permutes the values of bits in ${}^{md}S$, with each bit-change causing nodes to appear and disappear in <*I, J, K*> and consequently row-vectors to be removed and added from the mass-sample intensity

matrix *A*, affecting thus the *model 1* vector and the output value of the objective function.

6.4 Empirical results

Model 3 optimisation was applied on the Crohn's dataset of 18480 masses using 175 chemical reactions at 0.2 ppm. The resulting mass difference network of 5830 masses can be seen in figure 6.1. In a single optimisation run, the co-intensity network at thresholds $T = 0.23$ and $T = 0.50$ can be seen in figures 6.2A and 6.2B, respectively. 35 out of 175 reactions were deemed as discriminant at the algorithm's convergence. Crohn’s disease and inflammatory bowel disease are known to impair with the lack of Vitamin B6 (pyridoxal phosphate) [78]. 14 of the 35 reactions are potentially dependent on vitamin B6. Additionally a derivative of transamination (which is catalysed by vitamin B6) and reactions involving typtophan and its transamination product indolepyruvate were found to improve biological clusterability. Tryptophan was found by our group to be a discriminative metabolite in the same dataset [4]. The same study revealed Tyrosine to be of importance; a vitamin B6 dependent reaction of tyrosine as well as condensations with its transamination product 4-Hydroxyphenylpyruvic acid were found to improve the clustering output. Another pair of reactions which is linked via vitamin B6 is 2-Oxoarginine condensation and Arginine condensation on hydrogenated carbonyls. A further important transformation linking to vitamin B6 is thertiary N-methylation, which is found in the synthesis of Biotine and Choline. The transformations esterification with ‘phosphatidylcholine head group’ and ‘Phosphorylcholine’ were found to be important. Nitration, known to be a marker of inflammation, as well as thiolation were found to be important. Reactions which adhere to the context of mucosal injury are ‘decarboxylative condensation of Ornithine’ and ‘condensation of Ornithine’. Ornithine Decarboxylase is involved in the mucosa-protective formation of polyamines from Ornithine [79].

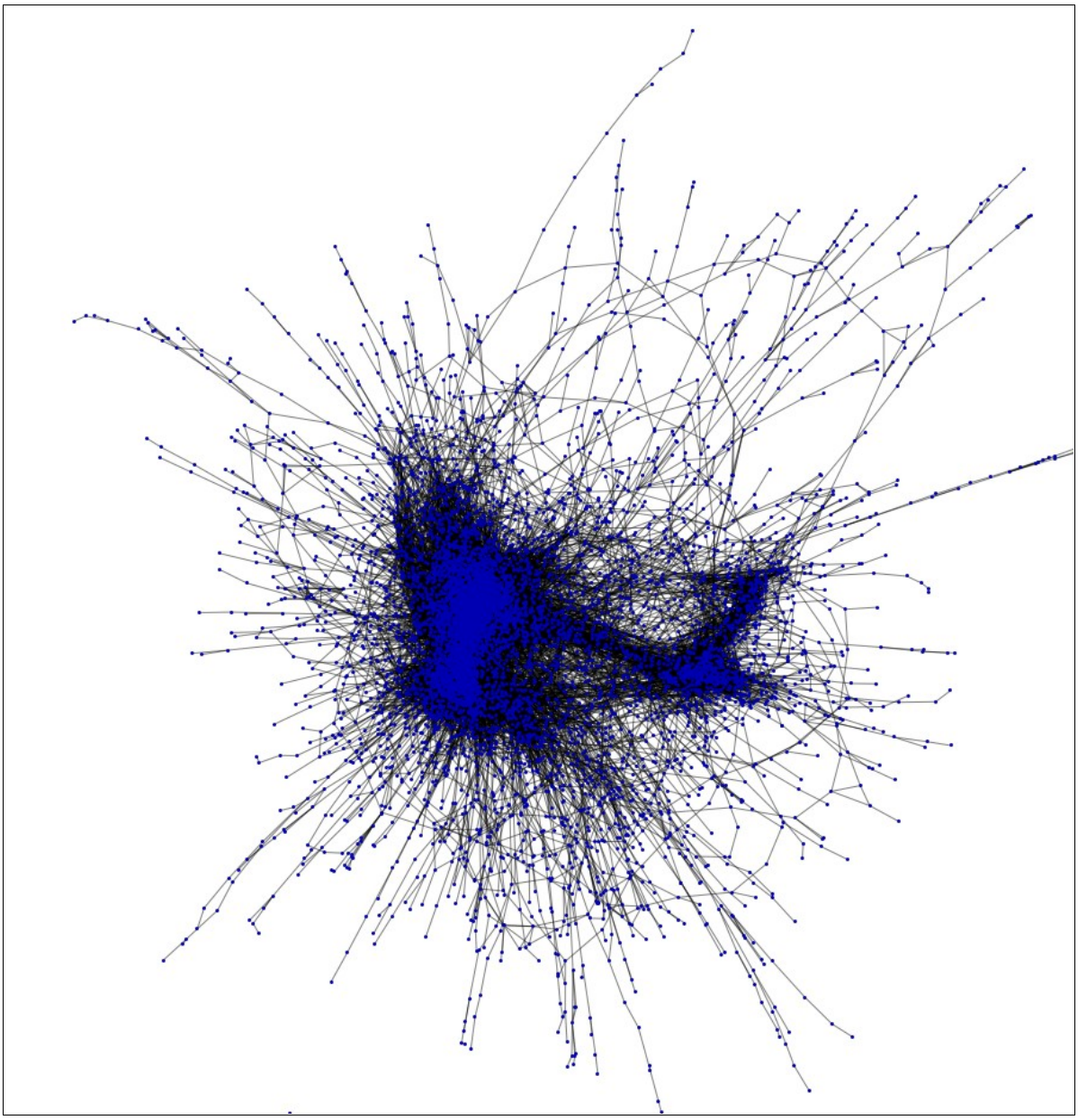

Figure 6.1: Structural mass difference network of the Crohn's dataset at 0.2 ppm (5830 masses - 175 reactions). The illustrated network was produced in the mass difference optimisation model for the purpose of discriminant mass difference identification. Despite its density, the network displays high modularity and scale-free architecture.

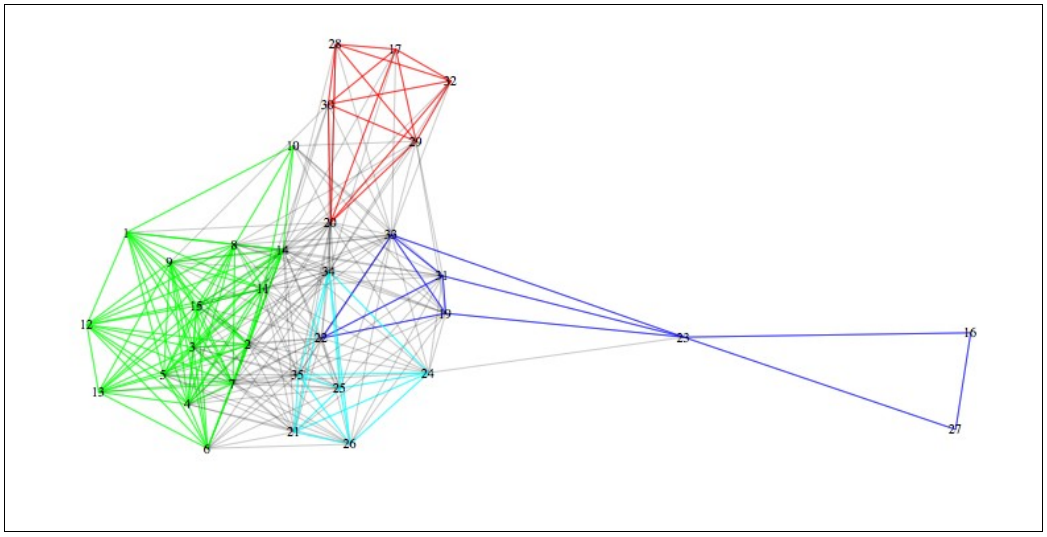

Figure 6.2A: Co-intensity network (at similarity threshold 0.23) produced by the masses mass difference optimisation. Green, blue, and red edges link samples belonging to the same biological group, HH, CCD, ICD, respectively. The network was constructed via the masses that took place in the mass difference network of figure 6.1. Graph modules in this co-intensity network largely correspond to known biological groups, showing moderate biological clusterability. The chemical reactions associated to this co-intensity network (vie the network of figure 6.1) are the resulting discriminant mass differences.

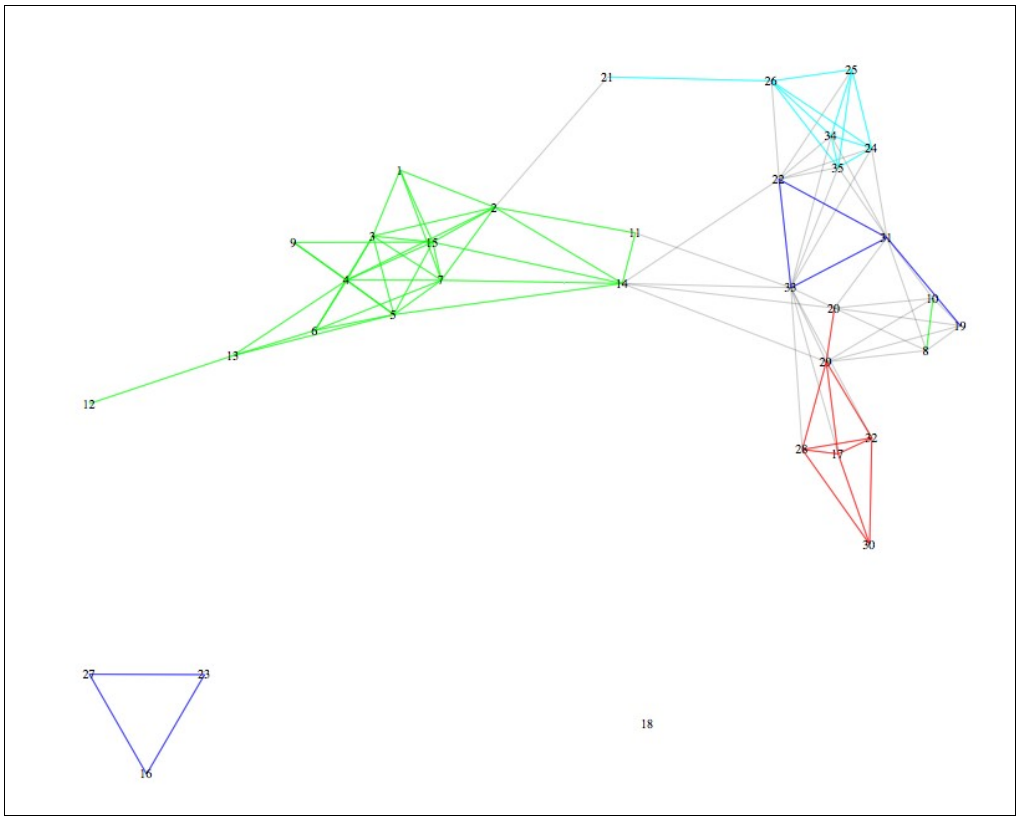

Figure 6.2B: Co-intensity network at threshold 0.50 produced by mass difference optimisation. Green, blue, and red edges link samples belonging to the same biological group, HH, CCD, ICD, respectively. The network was constructed via the masses that took place in the mass difference network of figure 6.1. Graph modules in this co-intensity network largely correspond to known biological groups, showing moderate biological clusterability. The chemical reactions associated to this co-intensity network (vie the network of figure 6.1) are the resulting discriminant mass differences.

6.5 Conclusion

In this chapter, we proposed, developed, and tested a new in silico method for determining discriminant mass differences (reactions) in a biochemical system based on the measure of biological clusterability that I presented in a previous chapter. The *mass difference optimisation* presented herein was materialised by merging the work seen in previous sections of this manuscript, notably structural mass difference networks, Netcalc, co-intensity network clustering, and the combinatorial optimisation framework for ICR-FT-MS data classification.

The results of the mass difference optimization adhere to the determinants of Crohn's Disease published by various scientific groups over time [4][79]. In addition, they confirm the medical praxis of handling Crohn's disease with a prescription of vitamin B treatment [78].

CHAPTER VII

Epilogue

This final chapter serves as a conclusion to my thesis. I review the main research topics of this work along with their corresponding experimental results and bring forward novel ideas for future work in this field.

7.1 Discussion

In this thesis, I have used my quantitative background in Artificial Intelligence (specialised in Machine Learning and Operational Research) in order to develop novel computational techniques specific to mass spectrometry bioinformatics. I have conceived and formalised a set of data mining methods and tools specifically adapted to high accuracy mass spectrometric data, produced by technologies such as ICR-FT-MS and Orbitrap. The work was focused on datasets of natural organic matter and metabolomics, treated by a 12T ICR-FT-MS instrument.

The first part of this manuscript focused on mass difference network reconstruction and the development of a network-based elementary formula calculation algorithm that we called 'Netcalc'. Traditional approaches to formula calculation are able to annotate a single exact mass by applying an exhaustive search on every possible combination of elemental composition that can correspond to that mass. However, the Netcalc algorithm

performs a biochemical network reconstruction of the entire sample and uses one known formula as a starting point in order to gradually annotate all connected masses in the network. Although in its infancy, Netcalc displayed superior results in respect to the number of correctly annotated masses and the computational time invested in the task. In addition, our enhanced structural mass difference network reconstruction model produces a data visualization which is far more informative than conventional methods such as Van Krevelen diagrams and Kendrick plots.

In the second part of this thesis, I focused on the development of novel machine learning techniques for the analysis of metabolomics samples. I first determined and adapted the most optimal clustering algorithm and similarity measure for the unsupervised classification of biological groups in ICR-FT-MS metabolomics datasets. I standardised the concept of co-intensity network clustering and demonstrated its applicability to ICR-FT-MS bioinformatics. I introduced the concept of biological clusterability which measures the quality of classification in terms of biological pertinence in a given sample, i.e. to what extent the in silico classes match with the real biological groups. I introduced a framework of combinatorial modelling which allows a biological scenario to be modelled as a discrete mathematical problem and solved computationally by means of combinatorial optimisation. I developed initially two models for this framework, one that is used for predictive/diagnostic clustering and semi-supervised learning, another for biomarker identification. The application of this model produced biologically pertinent and statistically significant results.

In the last section of this work I proposed, developed, and tested a new biological parameter in metabolomics research called 'discriminant mass differences'. The approach comes out as the natural merge of my work on combinatorial learning and structural mass difference networks. The first model of my optimisation framework was adapted and used with a new objective function in order to focus on finding discriminant mass differences in a sample. The results of the approach are in agreement with findings in recent literature.

7.2 Future work

The purpose of this work is to lay the foundations of new points of interest in computational and quantitative research of ICR-FT-FMS bioinformatics and mass spectrometry in general. The Netcalc approach, in particular, has advanced to the point of having its own distributable software application, regardless, it is considered to be in its infancy. There is lots of room for experimentation and improvement over aspects that I either formalised or simply tested, ranging from the selection of optimal ppm values and filtering methods to specialised search algorithms. Formula filtering is very crucial for the efficient validation of annotations while the algorithms involved can be affecting both speed and quality of search. My heuristic approach can be considered fast enough and sufficient for most tasks, yet there is room for further testing on exhaustive search methods. I experimented by programming algorithms for exhaustive and purely unsupervised search, which would build up their own reaction lists on the fly (mass difference mining) and use it to link and annotate all disconnected subgraphs of a mass difference network (source code not included in the manuscript). I did not deepen, however, my research into this direction and there is therefore much room for improvement. An interesting project would be to gather up all such alternative methods and options and integrate them into the Netcalc software tool, which would then have to be rewritten for efficiency in a general purpose programming language such as C++.

The community structure graph-clustering approach has proved itself to be optimal for unsupervised classification but, for what concerns cluster association to individual masses, I have discovered a big potential in Self-Organizing Maps (SOM); a type of Artificial Neural Network. In the case of supervised classification (prediction), the Perceptron family of Artificial Neural Networks was of particular interest as it was shown to be compatible with combinatorial optimisation modelling and Operational Research problem-solving. An integration of these methods with combinatorial learning could possibly yield the optimal tool for both unsupervised and supervised analysis.

The work on combinatorial learning and mass difference optimisation is by definition at an experimental stage. There is a vast parametrisation potential to these models and using

expert judgement to try different settings and obtain new results can be a separate topic of research on its own. I experimented a lot with different metaheuristic search algorithms with my focus being on Genetic Algorithms, Gradient Descent, and Simulated Annealing, as well as different objective functions that would calculate the *modularity* parameter in different ways. There are, however, many more things to be tested out, such as algorithms for the fitness of a solution (evaluation function), the speed of search, and the efficiency of clustering. Different filtering and cut-off techniques were also extensively tested, with a lot of work remaining to be done in these fields where parametrisation options are numerous.

As a last thought, I believe that a very interesting project would be the merge of all ICR-FT-MS techniques presented in this work into a complete software platform, where all complex parametrisation would be possible through a rich graphical user interface and its own scripting language. This would be undoubtedly a very big step towards the establishment of mass spectrometry data mining as a separate subfield of bioinformatics.

Bibliography

1. R. Breitling, S. Ritchie, D. Goodenowe, M.L. Stewart, M.P. Barrett, “Ab initio prediction of metabolic networks using Fourier transform mass spectrometry data”, Metabolomics 2, 155, (2006).

2. D. Tziotis, N. Hertkorn, P. Schmitt-Kopplin, “Kendrick-analogous network visualisation of ion cyclotron resonance Fourier transform mass spectra: improved options for the assignment of elemental compositions and the classification of organic molecular complexity”. Eur J Mass Spectrom (Chichester, Eng) 17, 415-421, (2011).

3. M. Haenlein, A.M. Kaplan, “A Beginner’s Guide to Partial Least Squares Analysis”, Understanding statistics, 3(4), 283–297, (2004).

4. J. Jansson, B. Willing, M. Lucio, A. Fekete, J. Dicksved et al., “Metabolomics Reveals Metabolic Biomarkers of Crohn's Disease”. PLoS ONE 4(7): e6386, (2009).

5. A.M. Turing, “Computing machinery and intelligence”, Mind 59 (236), 433-460, (1950).

6. J. McCarthy, “Programs with Common Sense”, Proceedings of the Teddington Conference on the Mechanization of Thought Processes, 756-91, (1959).

7. S. Cristoni, L.R. Bernardi, “Bioinformatics in mass spectrometry data analysis for proteomics studies”, Expert Rev Proteomics, 1(4):469-83, (2004).

8. A.G. Marshall, C.L. Hendrickson, G.S. Jackson, “Fourier transform ion cyclotron resonance mass spectrometry: a primer”, Mass Spectrom Rev 17, 1-35, (1998).

9. C. Mueller, “Metabolomics in host-pathogen interactions - An investigation of Chlamydia infected human cells”, Technische Universitaet Muenchen, (2012).

10. N. Hertkorn, M. Frommberger, M. Witt, B.P. Koch, Ph. Schmitt-Kopplin, E.M. Perdue, “Natural Organic Matter and the Event Horizon of Mass Spectrometry”, Anal. Chem., 80, 8908 (2008).

11. P. Pernot, N. Carrasco, R. Thissen, I. Schmitz-Afonso, “Tholinomics-Chemical Analysis of Nitrogen-Rich Polymers”, Anal. Chem., 82, 1371 (2010).

12. B.P. Koch, T. Dittmar, M. Witt, G. Kattner, “Fundamentals of Molecular Formula Assignment to Ultrahigh Resolution Mass Data of Natural Organic Matter”, Anal. Chem., 79, 1758 (2007).

13. E.B. Kujawinski, M.D. Behn, “Automated Analysis of Electrospray Ionization Fourier Transform Ion Cyclotron Resonance Mass Spectra of Natural Organic Matter”, Anal. Chem., 78, 4363 (2006).

14. S.G. Villas-Boas, J. Hojer-Pedersen, M. Akesson, J. Smedsgaard, J. Nielsen, “Global metabolite analysis of yeast: evaluation of sample preparation methods”, Yeast 22, 1155-1169, (2005).

15. R. Bhalla, K. Narasimhan, S. Swarup, “Metabolomics and its role in understanding cellular responses in plants”, Plant Cell Rep 24, 562-571, (2005).

16. H. Soh, M. Wasa, M. Fukuzawa, “Hypoxia upregulates amino acid transport in a human neuroblastoma cell line”, J Pediatr Surg 42, 608-612, (2007).

17. M.A. Ott, G. Vriend, “Correcting ligands, metabolites, and pathways”, BMC Bioinformatics 7, 517, (2006).

18. R.A. Dixon, D. Strack, “Phytochemistry meets genome analysis, and beyond”, Phytochemistry 62, 815-816, (2003).

19. N.M. Gruning, H. Lehrach, M. Ralser, “Regulatory crosstalk of the metabolic network”, Trends Biochem Sci 35, 220-227, (2010).

20. K.R. Patil, J. Nielsen, “Uncovering transcriptional regulation of metabolism”, Proc Natl Acad Sci, 102(8), 2685-9, (2005).

21. K. Saito, F. Matsuda, “Metabolomics for functional genomics, systems biology, and biotechnology”, Annu Rev Plant Biol, 61, 463-489, (2009).

22. J. Ihmels, R. Levy, N. Barkai, “Principles of transcriptional control in the metabolic network of Saccharomyces cerevisiae”, Nat Biotechnol, 22, 86-92, (2004).

23. D.G. Robertson, P.B. Watkins, M.D. Reily, “Metabolomics in toxicology: preclinical and clinical applications”, Toxicol Sci, 120, S146-170, (2011).

24. D.B. Kell, “Theodor Bucher Lecture. Metabolomics, modelling and machine learning

in systems biology - towards an understanding of the languages of cells", FEBS Congress and the 9th IUBMB conference in Budapest, FEBS, J 273, 873-894, (2006).

25. Cheriyadat, A., Bruce, L.M., Why principal component analysis is not an appropriate feature extraction method for hyperspectral data, Geoscience and Remote Sensing Symposium, 2003. IGARSS '03. Proceedings. 2003 IEEE International, 3420 - 3422 vol.6, (2003).

26. Renard, N., Bourennane, S., "Dimensionality Reduction Based on Tensor Modeling for Classification Methods", Geoscience and Remote Sensing, IEEE Transactions on (Volume:47 , Issue: 4), 1123 - 1131, (2009).

27. ALGLIB manual, "Principal Component Analysis", http://www.alglib.net/dataanalysis/principalcomponentsanalysis.php

28. MathWorks Matlab manual, "Partial Least Squares", http://www.mathworks.fr/fr/help/stats/partial-least-squares.html

29. mixOmics manual, "PLS Discriminant Analysis (PLS-DA)", Institut de Mathematiques de Toulouse, http://www.math.univ-toulouse.fr/~biostat/mixOmics/Methods_sPLSDA.html

30. A. Bondy and U.S.R. Murty, "Graduate Texts in Mathematics: Graph Theory", Springer, 1, 1-10, (2008).

31. A.L. Barabasi and Z.N. Oltvai, "Network biology: understanding the cell's functional organization", *Nat. Rev. Genet*. **5**, 101 (2004).

32. M.R. Carlson, B. Zhang, Z. Fang, P.S. Mischel, S. Horvath, et al., "Gene connectivity, function, and sequence conservation: Predictions from modular yeast co-expression networks", BMC Genomics, 7, 40, (2006).

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

34. E. Ravasz, A.L. Somera, D.A. Mongru, Z.N. Oltvai, A.L. Barabasi, "Hierarchical organization of modularity in metabolic networks", Science, 297, 1551, (2002).

35. J.A. Papin, J.L. Reed, B.O. Palsson, "Hierarchical thinking in network biology: the unbiased modularization of biochemical networks", Trends Biochem Sci, 12, 641-7, (2004).

36. C. Henegar, R. Cancello, S. Rome, H. Vidal, K. Clément, J.D. Zucker, "Clustering Biological Annotations and Gene Expression Data to Identify Putatively Co-regulated Biological Processes", J. Bioinformatics and Computational Biology, 4(4), 833-852 (2006).

37. R. Guimera, L.A. Nunes Amaral, “Functional cartography of complex metabolic networks”, Nature, 433(7028), 895-900, (2005).

38. P. Baldi, S. Brunak, “Bioinformatics: The Machine Learning Approach, Second Edition, (Adaptive Computation and Machine Learning)”, The MIT Press, (2001).

39. G. D. Stormo, T. D. Schneider, L. Gold, A. Ehrenfeucht, “Use of the perceptron algorithm to distinguish translational initiation sites in e. Coli”, Nucl. Acids Res., 10, 2997–3011, (1982).

40. N. Qian and T. J. Sejnowski. “Predicting the secondary structure of globular proteins using neural network models”, *J. Mol. Biol.*, 202, 865-884, (1988).

41. E. Alpaydin, “Introduction to Machine Learning”, The MIT Press, (2004).

42. MathWorks Matlab manual, “Hierarchical Clustering”, http://www.mathworks.fr/fr/help/stats/hierarchical-clustering.html

43. T. Weise, “Global Optimization Algorithms - Theory and Application”, (2008) http://www.it-weise.de/projects/book.pdf

44. U. Diwekar, “Introduction to Applied Optimization”, Springer Optimization and Its Applications, Vol. 22, (2008).

45. D.J. Wilde, C.S. Beightler, “Foundations of Optimization”, Prentice-Hall, 1, 1-10, (1967).

46. N. Gould, “An introduction to algorithms for continuous optimization”, (2006). http://www.numerical.rl.ac.uk/people/nimg/course/lectures/paper/paper.pdf

47. M. Minoux, “Programmation Mathematique, Theorie et Algorithmes”, Dunod Paris, (1983).

48. S. Geman, D. Geman, “Stochastic relaxation, Gibbs distributions and the Bayesian restoration of images”, IEEE Trans. Pattern Anal. Machine Intell., 6, 721– 741, (1984).

49. J. R. Koza. “Evolution of a computer program for classifying protein segments as transmembrane domains using genetic programming”, Proceedings of the Second International Conference on Intelligent Systems for Molecular Biology - AAAI Press, 244-252, (1994).

50. S. Handley, “Classifying nucleic acid sub-sequences as introns or exons using ge- netic programming”, “Proceedings of the Third International Conference on Intelligent Systems for Molecular Biology - AAAI Press, 162-169, (1995).

51. R. Parsons, M. E. Johnson, “DNA sequence assembly and genetic programming - new results and puzzling insights”, Proceedings of the Third International Conference on Intelligent Systems for Molecular Biology, AAAI Press, 277-284, (1995).

52. J. Meija, “Mathematical tools in analytical mass spectrometry”, Anal. Bioanal. Chem., 385, 486 (2006).

53. S.A. Visser, ”Application of Van Krevelen's graphical-statistical method for the study of aquatic humic material”, Environ. Sci. Technol., 17, 412, (1983).

54. A. Reinhardt, C. Emmenegger, B. Gerrits, C. Panse, J. Dommen, U. Baltensperger, R. Zenobi and M. Kalberer, “Ultrahigh mass resolution and accurate mass measurements as a tool to characterize oligomers in secondary organic aerosols”, Anal. Chem., 79, 4074, (2007).

55. C.A. Hughey, C.L. Hendrickson, R.P. Rodgers, A.G. Marshall, K. Qian, “Kendrick Mass Defect Spectrum: A Compact Visual Analysis for Ultrahigh-Resolution Broadband Mass Spectra”, Anal. Chem., 73, 4676, (2001).

56. P. Schmitt-Kopplin, Z. Gabelica, R.D. Gougeon, A. Fekete, B. Kanawati, M. Harir, I. Gebefuegi, G. Eckel, N. Hertkorn, “High molecular diversity of extraterrestrial organic matter in Murchison meteorite revealed 40 years after its fall”, PNAS, 107, 2763, (2010).

57. P. Schmitt-Kopplin, A. Gelencser, E. Dabek-Zlotorzynska, G. Kiss, N. Hertkorn, M. Harir, Y. Hong, I. Gebefuegi, “Analysis of the Unresolved Organic Fraction in Atmospheric Aerosols with Ultrahigh-Resolution Mass Spectrometry and Nuclear Magnetic Resonance Spectroscopy: Organosulfates As Photochemical Smog Constituents”, Anal. Chem., 82, 8017, (2010).

58. S. Kim, R.W. Kramer, P.G. Hatcher, “Graphical method for analysis of ultrahighresolution broadband mass spectra of natural organic matter, the van Krevelen diagram”, Anal Chem 75, 5336-5344, (2003).

59. D. Ohta, S. Kanaya, H. Suzuki, “Application of Fourier-transform ion cyclotron resonance mass spectrometry to metabolic profiling and metabolite identification”,

Curr Opin Biotechnol, 21, 35-44, (2010).

60. E.V. Kunenkov, A.S. Kononikhin, I.V. Perminova, N. Hertkorn, A. Gaspar, P. Schmitt-Kopplin, I.A. Popov, A.V. Garmash, E.N. Nikolaev, “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”, Anal. Chem., 81, 10106, (2009).

61. Y. Hu, “Efficient and high quality force-directed graph drawing”, Mathematica Journal 10, 37 (2006).

62. T. Grinhut, D. Lansky, A. Gaspar, N. Hertkorn, P. Schmitt-Kopplin, Y. Hadar and Y. Chen, “Novel software for data analysis of Fourier transform ion cyclotron resonance mass spectra applied to natural organic matter”, Rapid Commun. Mass Spectrom., 24, 2831, (2010).

63. T. Kind, O. Fiehn, “Seven Golden Rules for heuristic filtering of molecular formulas obtained by accurate mass spectrometry”, BMC Bioinformatics 2007, 8, 105, (2007).

64. K. Suhre, P. Schmitt-Kopplin, “Masstrix: Mass translator into pathways”, Nucleic Acids Research, 36, W481–W484, (2008).

65. Schmitt-Kopplin et al., “Ultrahigh resolution Fourier transform ion cyclotron (FTICR) mass spectrometry for the analysis of natural organic matter from various environmental systems”, book chapter in A.T. Lebedev, “Comprehensive Environmental Mass Spectrometry”, ILM Publications, p. 443-450.

66. J. Antón, M. Lucio, A. Peña, A. Cifuentes, J. Brito-Echeverría, F. Moritz, D. Tziotis, C. López, M. Urdiain, P. Schmitt-Kopplin, R. Rosselló-Móra, “High Metabolomic Microdiversity within Co-Occurring Isolates of the Extremely Halophilic Bacterium Salinibacter ruber”, PLoS ONE, 8(5), e64701, (2013).

67. B.J. Webb-Robertson, M.M. Matzke, J.M. Jacobs, J.G. Pounds, K.M. Waters, “A statistical selection strategy for normalization procedures in LC-MS proteomics experiments through dataset-dependent ranking of normalization scaling factors”, Proteomics, 11(24), 4736-41, (2011).

68. O. Yeniay, A, Goktas, “A comparison of partial least squares regression with other prediction methods”, Hacettepe Journal of Mathematics and Statistics, 31, 99-111, (2002).

69. R.D. Cramer, “Partial least squares (PLS): Its strengths and limitations”, Springer, Perspectives in Drug Discovery and Design, 1, 269-278, (1993).

70. J.K. Nicholson, E. Holmes, I.D. Wilson, “Gut microorganisms, mammalian metabolism and personalized health care”, Nature Rev Microbiol, 3, 431–438, (2005).

71. R. Rossello-Mora, M. Lucio, A. Pena, J. Brito-Echeverria, A. Lopez-Lopez, et al., “Metabolic evidence for biogeographic isolation of the extremophilic bacterium Salinibacter ruber”. ISME J, 2, 242–253, (2008).

72. J. Chen, X. Zhao, J. Fritsche, P. Yin, P. Schmitt-Kopplin, et al., “Practical approach for the identification and isomer elucidation of biomarkers detected in a metabonomic study for the discovery of individuals at risk for diabetes by integrating the chromatographic and mass spectrometric information”., Anal Chem, 80, 1280–1289, (2008).

73. Sean Luke, “Essentials of Metaheuristics”, Lulu, (2009), available for free at http://cs.gmu.edu/~sean/book/metaheuristics/

74. U. Brandes , D. Delling , M. Gaertler , R. Görke , M. Hoefer , Z. Nikoloski , D. Wagner, “On Modularity -- NP-Completeness and Beyond”, (2006).

75. M. Lucio, A. Fekete, C. Weigert, B. Wägele, X. Zhao, et al., “Insulin Sensitivity Is Reflected by Characteristic Metabolic Fingerprints - A Fourier Transform Mass Spectrometric Non-Targeted Metabolomics Approach”, PLoS ONE, 5(10), e13317, (2010).

76. M. Matsuda, R.A. DeFronzo, “Insulin sensitivity indices obtained from oral glucose tolerance testing: comparison with the euglycemic insulin clamp”, Diabetes Care, 22, 1462–1470, (1999).

77. F. Moritz, S. Forcisi, M. Harir, B. Kanawati, M. Lucio, D. Tziotis, Philippe Schmitt-Kopplin, “The Potential of Ultrahigh Resolution MS (FTICR-MS) in Metabolomics”, book chapter in Michael Lämmerhofer, Prof. Dr. Wolfram Weckwerth, “Metabolomics in Practice: Successful Strategies to Generate and Analyze Metabolic Data”, Wiley-VCH Verlag GmbH & Co., (2013).

78. Saibeni et al., “Low vitamin B6 plasma levels, a risk factor for thrombosis, in inflammatory bowel disease: role of inflammation and correlation with acute phase reactants”, American Journal of Gastroenterology, 98, 112-17, (2003).

79. G. Ricci, G. Stabellini, G. Bersani, G. Marangoni, P. Fabbri, G. Gentili, V. Alvisi, "Ornithine decarboxylase in colonic mucosa from patients with moderate or severe Crohn's disease and ulcerative colitis", EUR J GASTR, 11(8), 903-904, (1999).

Curriculum Vitae

Dimitrios Tziotis

05/2014 - present

Ernst & Young, Enterprise Intelligence and Data Analytics
Data Scientist - Quantitative Analytics Consultant (Senior)

04/2013 – 05/2014

Institut Pasteur (Paris), Neurobiology - Structural Bioinformatics
Biostatistician - Data Scientist

09/2009 - 01/2013

Helmholtz Zentrum Muenchen - Technical University Munich (TUM)
Data Scientist - Machine Learning engineer
PhD thesis: "Machine learning and network analysis using mathematical optimisation in Mass Spectrometry Bioinformatics".

2008-2009

Academy of Athens, Biomedical Research Foundation, Bioinformatics group
Bioinformatics Data Analyst

2007-2008

Université Paris-Descartes (Paris V)
MSc, Biomedical Informatics: Bioinformatics
Rank in the Master's programme: 1st

2005-2007

Université Pierre et Marie Curie (Paris VI)
MPhil, Artificial Intelligence: Operational Research and Machine Learning

2001-2004

Queen Mary, University of London
BEng, Electronic and Computer Engineering
Upper Second-Class Honours

List of scientific communication

Publications

1. D. Tziotis, N. Hertkorn, P. Schmitt-Kopplin, "Kendrick-analogous network visualisation of ion cyclotron resonance Fourier transform mass spectra: improved options for the assignment of elemental compositions and the classification of organic molecular complexity", Eur J Mass Spectrom (Chichester, Eng), (2011).

2. Philippe Schmitt-Kopplin, Liber-Belair G, Boris P. Koch, Flerus R, Gerhard Kattner, Mourad Harir, Basem Kanawati, Marianna Lucio, Dimitrios Tziotis, Norbert Hertkorn, Istvan Gebefugi, "Dissolved organic matter in sea spray: a transfer study from marine surface water to aerosols", Biogeosciences, (2011).

3. Michael Witting, Marianna Lucio, Dimitrios Tziotis, Philippe Schmitt-Kopplin, "Ultrahigh resolution mass spectrometry based non-targeted microbial metabolomics", book chapter for "Genetics Meets Metabolomics: book chapter from Experiment to Systems Biology" by Karsten Suhre, Springer, (2010).

4. Schmitt-Kopplin1,2, Ph., Harir1, M., Tziotis1 D., Gabelica3, Z., Hertkorn1, N., "Ultrahigh resolution Fourier transform ion cyclotron (FTICR) mass spectrometry for the analysis of natural organic matter from various environmental systems", book chapter for "Comprehensive Environmental Mass Spectrometry (Advanced Topics in Environmental Science)" by Albert T. Lebedev, ILM Publications, (2011).

5. G. Chong-Diaz, M. Ruíz-Bermejo, M. Harir , P. Schmitt-Kopplin, D. Tziotis , D. Gomez-Ortiz, et al., "Molecular preservation in halite and perchlorate rich hypersaline subsurface deposits in the Salar Grande basin (Atacama Desert, Chile): implications for the search of organics on Mars.", Journal of Geophysical Research - Biogeosciences, (2012)

6. C. Müller, I. Dietz, D. Tziotis, F. Moritz, J. Rupp, P. Schmitt-Kopplin, "Molecular cartography in acute Chlamydia pneumoniae infections--a non-targeted metabolomics approach", Anal Bioanal Chem, (2013).

7. B.S. Sixt, A. Siegl, C. Müller, M. Watzka, A. Wultsch, D. Tziotis, J. Montanaro, A. Richter, P. Schmitt-Kopplin, M. Horn, "Metabolic features of Protochlamydia amoebophila elementary bodies - a link between activity and infectivity in Chlamydiae", PLoS Pathogens, (2013).

8. J. Antón, M. Lucio, A. Peña, A. Cifuentes, J. Brito-Echeverría, F. Moritz, D. Tziotis, C. López, M. Urdiain, P. Schmitt-Kopplin, R. Rosselló-Móra, "High Metabolomic Microdiversity within Co-Occurring Isolates of the Extremely Halophilic Bacterium Salinibacter ruber", PLoS ONE, (2013).

9. Forcisi S, Moritz F, Kanawati B, Tziotis D, Lehmann R, Schmitt-Kopplin P, “Liquid chromatography-mass spectrometry in metabolomics research: mass analyzers in ultra high pressure liquid chromatography coupling”, J Chromatogr A, (2013).

10. F. Moritz, S. Forcisi, M. Harir, B. Kanawati, M. Lucio, D. Tziotis, Philippe Schmitt-Kopplin, “The Potential of Ultrahigh Resolution MS (FTICR-MS) in Metabolomics”, book chapter in Michael Lämmerhofer, Prof. Dr. Wolfram Weckwerth, “Metabolomics in Practice: Successful Strategies to Generate and Analyze Metabolic Data”, Wiley-VCH Verlag GmbH & Co., (2013).

11. C. Müller, M. Harir, N. Hertkorn, B. Kanawati, D. Tziotis, P. Schmitt-Kopplin, “Using ultrahigh resolution mass spectrometry to unravel the chemical space of complex natural product mixtures”, book chapter in “Natural Products Analysis: Instrumentation, Methods, and Applications”, edited by Vladimir Havlicek, In press, (2013).

12. K. Wörmann, A. Walker, F. Moritz, S. Forcisi, D. Tziotis, M. Lucio, S. S Heinzmann, J. Adamski, R. Lehmann, H.U. Häring, P. Schmitt-Kopplin, “Revolution in der Diabetesdiagnostik dank -omics: Biomarker mittels Metabolomics”, Diabetes aktuell, (2012).

<u>Erklärung</u> :

Ich erkläre an Eides statt, dass ich die der Fakultät für Wissenschaftszentrum Weihenstephan für Ernährung, Landnutzung und Umwelt der Technischen Universität München zur Promotionsprüfung vorgelegte Arbeit mit dem Titel:

“Machine Learning and Network Analysis using Mathematical Optimisation in Mass Spectrometry Bioinformatics” unter der Anleitung und Betreuung durch Priv.-Doz. Dr. Philippe Schmitt-Kopplin ohne sonstige Hilfe erstellt und bei der Abfassung nur die gemäß §6 Abs. 5 angegebenen Hilfsmittel benutzt habe.

(x) Ich habe die Dissertation in keinem anderen Prüfungsverfahren als Prüfungsleistung vorgelegt.

(x) Ich habe den angestrebten Doktorgrad noch nicht erworben und bin nicht in einem früheren Promotionsverfahren für den angestrebten Doktorgrad endgültig gescheitert.

Die Promotionsordnung der Technischen Universität München ist mir bekannt.

München, den 29.03.2013

Acknowledgement

Regarding the completion of this thesis, I would like to thank my boss and supervisor apl. Prof. Dr. Philippe Schmitt-Kopplin for offering me this opportunity, entrusting me with his ideas that have been the basis of my work, and having worked closely with me for over three years, during which l felt as an integral part of a team. A big thanks goes to Franco Moritz for being my closest collaborator, friend, and most enthusiastic supporter of my work, without whom the outcome of this thesis would have not been the same. I would additionally like to thank all colleagues who showed a sincere interest in my project and all people who stood by me as friends.

For what concerns the course of my academic career, I would like to thank the teachers, professors, and tutors, who inspired me, as well as the friends who offered me selfless and invaluable support during times of need.

Overall, I would like to thank all people, friends and family, who have been close to me. I would especially like to thank my parents, Angeliki and Theophilos Tziotis, for always believing in me. Their moral and material support has made everything possible.