Verena Spath, BSc

A dynamic network based approach for discovering new metabolic biomarkers in myocardial injury

MASTER'S THESIS

to achieve the university degree of

Diplom-Ingenieurin

Master's degree program: Biomedical Engineering

submitted to

Graz University of Technology

Supervisor

Univ.-Prof. Dipl.-Ing. Dr.techn. Christian Baumgartner

Institute of Health Care Engineering
with European Testing and Certification Body for Medical Devices

Graz, March 2017

Acknowledgements

First of all, I would like to thank my parents for providing me guidance and assistance throughout the course of my education and my whole life, and without whom I would have never been able to achieve my goals.

In addition, I owe my deepest gratitude to my finance Martin, who always supports me in any possible way. He provides me with encouragement and helps me to keep a differentiated view on my personal objectives.

Furthermore, I would like to thank my colleagues. I will always remember the time we spent together working on our master's theses. The constant interchange kept me motivated and focused, and helped me to accomplish my goals step by step.

Last but not least, special thanks to my friends who I can always count on, and to my brothers who are not only family but also friends to me.

Abstract

A dynamic network based approach for discovering new metabolic biomarkers in myocardial injury

Metabolic biomarkers play an important role in the discovery and classification of diseases. The process of discovering new metabolic biomarkers is a non-trivial task which includes a number of different steps such as experimental design and data generation, as well as data-mining tasks followed by the verification and validation of the findings. This highly complex process requires an interdisciplinary collaboration of various fields of expertise such as medicine, biochemistry and bioinformatics.

In this thesis, a new dynamic network based approach for the visualization of putative biomarker candidates is realized in order to provide an additional tool for the identification of early metabolic biomarkers in cardiovascular diseases. All experiments and algorithms of this work are implemented in `R` based on the provided data containing longitudinal information about putative metabolic biomarker candidates in myocardial injury. A novel measure called Paired Biomarker Identifier is used for both feature selection and graph inference. The graphs are constructed for different static and dynamic threshold values, with the resulting network graphs being visualized by using different representation tools in order to highlight various aspects of the given data. An exemplary verification of the findings, i.e. the outcome of each method proposed in this thesis, based on the implemented graphs is performed using `KEGG` database. The exemplary `KEGG` database research confirmed the principle used for the graph inference and the visualization approaches proposed in this thesis and provides useful assistance in the biomarker identification process.

Keywords: metabolic biomarker, discovery process, dynamic networks, visualization, myocardial injury

Ein dynamischer netzwerkbasierter Ansatz zur Identifizierung metabolischer Biomarker in Myokardschäden

Metabolische Biomarker spielen eine wichtige Rolle in der Entdeckung und Klassifizierung von Krankheiten. Der Prozess der Entdeckung neuer metabolischer Biomarker umfasst eine Vielzahl unterschiedlicher Schritte, wie etwa das Design eines Experiments, die Datenerzeugung, verschiedene Datenanalysemethoden und letztendlich die Validierung. Dieser hochkomplexe Prozess erfordert die interdisziplinäre Zusammenarbeit verschiedener Fachgebiete wie Medizin, Biochemie und Bioinformatik.

In dieser Arbeit wird ein neuer, dynamischer und auf Netzwerken basierender Ansatz für die Visualisierung potentieller Biomarker-Kandidaten realisiert, um ein zusätzliches Werkzeug zur Identifikation früher metabolischer Biomarker für kardiovaskuläre Erkrankungen zur Verfügung zu stellen. Alle in dieser Arbeit verwendeten Experimente und Algorithmen sind in `R` implementiert und basieren auf den Daten einer longitudinalen Studie über metabolische Biomarker in Myokardverletzungen. Sowohl für die Merkmalextraktion als auch für die Netzwerkerstellung wurde eine neue Berechnungsmethode, genannt „Paired Biomarker Identifier“, verwendet. Für verschiedene statische und dynamische Schwellwerte wurden zudem Graphen erstellt. Die sich daraus ergebenden Netzwerke wurden mithilfe verschiedener Repräsentationsmethoden visualisiert, um unterschiedliche Aspekte der Daten hervorzuheben. Die Ergebnisse, d. h. die Resultate jeder der in dieser Arbeit vorgeschlagenen Methoden, werden unter Verwendung der `KEGG` Datenbank exemplarisch verifiziert. Diese exemplarische Recherche bestätigte sowohl das für die Berechnung der Graphen angewandte Prinzip, als auch die vorgeschlagenen Visualisierungsmethoden, die Hilfestellung im Biomarker-Identifikationsprozess geben sollen.

Schlüsselwörter: metabolische Biomarker, Entdeckungsprozess, dynamische Netzwerke, Visualisierung, Myokardverletzungen

AFFIDAVIT

I declare that I have authored this thesis independently, that I have not used other than the declared sources/resources, and that I have explicitly indicated all material which has been quoted either literally or by content from the sources used. The text document uploaded to TUGRAZonline is identical to the present master‘s thesis.

Date

Signature

Contents

List of Abbreviations

AMI	acute myocardial infarction
AMP	adenosine monophosphate
AMPK	AMP-activated protein kinase
ATP	adenosine triphosphate
CAP	catabolite activator protein
CK	creatine-kinase
CK-MB	creatine-kinase muscle and brain subunits
CMP	cytidine monophosphat
CT	computer tomography
CV	coefficient of variation
DA	discriminantory ability
ECG	electrocardiogram
GCP	Good Clinical Practice
GO	Gene Ontology
H-FABP	heart fatty acid-binding protein
HOCM	hyperthrophic obstructive cardiomyopathy
KEGG	Kyoto Encyclopedia of Genes and Genomes
MeSH	Medical Subject Heading
MI	myocardial infarction
MRI	magnetic resonance imaging
MS	mass spectrometry
NCBI	National Center for Biotechnology Information
NIH	National Institutes of Health
pBI	paired Biomarker Identifier
PMI	planned myocardial infarction
PPI	Protein-Protein Interactions
RFM	Random Forest Models
ROC	Receiver Operating Characteristics
SMI	spontaneous myocardial infarction
SVM	Support Vector Machine
UMP	uracil monophosphate

1 Introduction

Due to new innovations in high-throughput technologies, the role of metabolic biomarkers in disease discovery and classification has become more and more important in the last few decades. The discovery processes performed in order to identify novel biomarker candidates generate huge amounts of data. Thus, both the analysis and selection of putative biomarkers from the data provided require an interdisciplinary collaboration of various fields of expertise as well as the integration of bioinformatics. There exist various feature-selection tools for the extraction of highly-discriminatory biomarker candidates. Nevertheless, further research is still necessary in order to enhance the existing methods and develop new tools for the data analysis. Therefore, the bioinformatic-driven search process for new biomarker candidates is under continuous development and comprises new innovations in the fields of data-mining and the visualization of the generated data. [1–3]

A commonly used approach in various fields of expertise is the network creation and analysis. Especially within the fields of biology and medical research, does network representation of data play an important role for the prediction of functionality and for the identification of correlations within the data. The determination of protein or gene functions and interactions as well as drug target identification or the diagnosis of various diseases pose potential applications of network analyses. Thus, protein-protein interactions, metabolic pathways or signal transduction are often modelled as network graphs. [4]

The approach of representing study results as network graphs is also used by Netzer et al. [2]. Here, metabolic data of a study cohort exposed to physical stress is analyzed in terms of an inferred network graph. The network is constructed based on a novel feature selection tool which can be applied to paired data. The graph visualization is performed in order to identify putative biomarker candidates.

The method proposed by [2] is adapted for the graph construction and visualization in the present thesis. In the present case, graphs are created based on metabolic data of patients with myocardial injury. For a convenient way of analyzing this data, different graph representation methods are implemented and compared.

The thesis in hand aims at revealing and discussing advantages and disadvantages of the implemented methods with regard to the identification of potential biomarkers. The purpose of the proposed methods is to enable a compact and interactive representation of the given data. In addition, it is of particular interest to examine the dynamics, i.e. the temporal evolution and changes of data generated by longitudinal studies.

2 Assignment

The aim of this thesis is to develop a dynamic approach for the network based analysis and visualization of newly discovered putative metabolic biomarkers in myocardial injury. For this purpose, the thesis is subdivided into a theoretical and a practical part.

In order to introduce the basic concepts of the proposed topic and provide an overall understanding, the general background of the biomarker discovery process is investigated. First of all, a literature review covering all main tasks of biomarker identification and including the accomplishment of biomarker discovery studies as well as preprocessing and data mining tasks (e.g. feature selection) is performed. Further literature research then comprises the study design's basics including the longitudinal study design and the study execution. In addition to the biomarker identification, a biological interpretation of identified biomarker candidates will also be taken into consideration. In order to obtain an overview of the state of art in the field of biological interpretation of putative biomarkers, the analysis is performed with the aid of metabolic pathway databases.

After having covered these principal concepts, theoretical preparations for the practical part of the thesis are necessary. The basis for this data is formed by data generated in a study which investigated potential biomarker candidates in early phases after myocardial injury [1]. The provided data is analyzed with regard to the theoretical literature research. This includes a scientific discussion of the study's structure and execution as well as the applied data mining tasks. Following the analysis, the given data can then be used in the practical part of the present master thesis.

Subsequently, a suitable network based method is developed by reviewing literature focusing on metabolic networks and their structure, as well as by taking a closer look at dynamical networks and their representations. The review comprises the basic concepts of network theory, different types of network representations as well as basic network measures which can be applied to compare different networks.

In the practical part of this thesis, networks based on the provided metabolic data can then be implemented. This should develop a method which represents the dynamic changes of the given longitudinal metabolic data over time. Following the network inference, a suitable visualization tool can be applied on the networks in order to ensure a simple and straightforward interpretation of the visualized data. If possible, this task should be accomplished in a manner that allows re-usability of the implemented method.

The entire literature review as well as the implemented methods and network visualizations need to be summarized in the master theses in hand.

3 Biomarker

3.1 Biomarker Definition

The denotation biomarker – a compound of the words "biological marker" – was first introduced as a Medical Subject Heading (MeSH) term in 1989. According to the National Center for Biotechnology Information (NCBI), biomarkers are defined as: "Measurable and quantifiable biological parameters [...] which serve as indices for health- and physiology-related assessments [...]." [5].

In 2001, another definition was released by the Biomarkers Definitions Working Group of the National Institutes of Health (NIH). They defined biological markers as objectively measurable and quantifiable indicators of normal biological or pathogenic processes, or as responses to a specific treatment [6].

Taking into account these definitions, biomarkers can be described as biological parameters or medical signs which allow the characterization of different health and/or pathological states of an organism, e.g. different disease states or disease traits. Biomarkers can be biological parameters measured in biosamples such as blood or tissue tests, recordings of processes of human bodies including blood pressure and electrocardiogram (ECG), or results of imaging tests such as computer tomography (CT) scans or magnetic resonance imaging (MRI). [6, 7]

According to a definition by the International Programme on Chemical Safety led by the World Health Organization (WHO), biomarkers can be chemicals, metabolites of chemicals, enzymes and other biochemical substances which can be measured in tissue or body fluids. From a broader perspective, the term biomarker comprises almost all measurements that reflect an interaction between a biological system and potential chemical, biological or physical threats. These measurements' responses could be molecular interactions (e.g. as biochemicals at a cellular level), or they could be of functional and physiological nature. [8]

3.1.1 Applications of Biomarkers

There are various applications for biomarkers, including their use as precise clinical measurement tools for disease detection and monitoring. This allows a categorization according to their specific applications in clinical use. It can be distinguished between antecedent biomarkers used to identify the risk of developing an illness, diagnostic biomarkers used for the detection of abnormal conditions in patients, biomarkers used to differentiate various states and extends of diseases, prognostic biomarkers used as indicators for disease prognosis and biomarkers that predict or monitor responses to different therapeutic interventions. [6, 7]

Depending on the type of application, the desired properties of a biomarker may change. Thus, screening biomarkers require high predictive values as well as a high specificity and sensitivity, whereas diagnostic biomarkers of acute diseases need to be detectable fast and need to show a correlation between their concentration and the extent of the disease. However, basic features such as accuracy, reproducibility, a high specificity for the outcome to identify and an easy interpretation are similar for the evaluation of the clinical benefit for each new biomarker. [7]

Biomarkers can also be used in clinical trials to provide a foundation for the designing. Moreover, they can act as a substitute for clinical endpoints that are defined as the outcome of a trial. Per definition, a clinical endpoint is a measurable parameter or characteristic that reflects a patient's condition, e.g. their body functions or survival, as a response to a therapeutic intervention within the scope of a randomized clinical trial. In this context, biomarkers are useful tools to assess the clinical impact of therapeutic interventions as well as their benefits and risks. If biomarkers are intended as a substitute for such a clinical endpoint, they are referred to as surrogate endpoints. Clinical endpoints, such as survival or mortality, occur very infrequently – sometimes only after many years of treatment – which makes their application in clinical trials a poor choice. In contrast, biomarkers as surrogate endpoints allow to collect data in shorter time while still being able to permanently monitor the safety and efficiency of a treatment. This immediately provided data ensures a rapid validation of a certain treatment and, consequently, enables researchers or clinicians to stop a potentially harmful intervention without having to wait for clinical data. Furthermore, biomarkers allow to design smaller as well as more specific and efficient studies due to their specificity. [6, 7, 9]

However, not all biomarkers are useful for substituting a clinical endpoint. There has to be a profound preselection in order to decide which biomarkers suffice to serve as surrogate endpoints. Therefore, there exist guidelines as proposed by A. B. Hill [10] which define criteria for evaluating an association between a disease and environmental features so as to determine the cause. These guidelines can also be used to assess an association between a biomarker and a disease. Hill proposes to use the strength, consistency, specificity and other features of associations as criteria. When a biomarker meets these criteria, it is more likely to be a useful surrogate endpoint. Yet, there is no absolute guarantee for a biomarker being a useful substitute even if it fulfils the above criteria. [10, 11]

Be that as it may, in order to start the debate about the use of biomarkers as surrogate endpoints, they have to be discovered first. Thus, the identification process that leads to the detection of new putative biomarker candidates is described in detail in the following sections.

3.2 Biomarker Discovery Process

The search and discovery of new biomarkers require a collaboration of different fields of expertise including biology, analytical- and biochemistry, medicine, biostatistics and bioinformatics as well as knowledge and innovations in the field of high-throughput technologies. Therefore, this interdisciplinary research results in a complex, multi-stage procedure which may be resource intensive, in particular with regard to financial resources. [3]

However, the process of identifying new biomarker candidates can be subdivided into several single process phases and comprises the discovery of potential biomarker candidates as well as their verification and clinical validation. These major steps of the identification process are depicted in figure 3.1 and described in detail in the following sub-sections.

3.2.1 Biomarker Discovery

In the beginning of each biomarker discovery procedure, the framework of the study has to be defined, including the definition of research hypotheses and if necessary ethical approval has to be obtained. Subsequently, study execution with sample collection can be started. The obtained samples undergo a thorough bio-analytical process and the collected data is processed

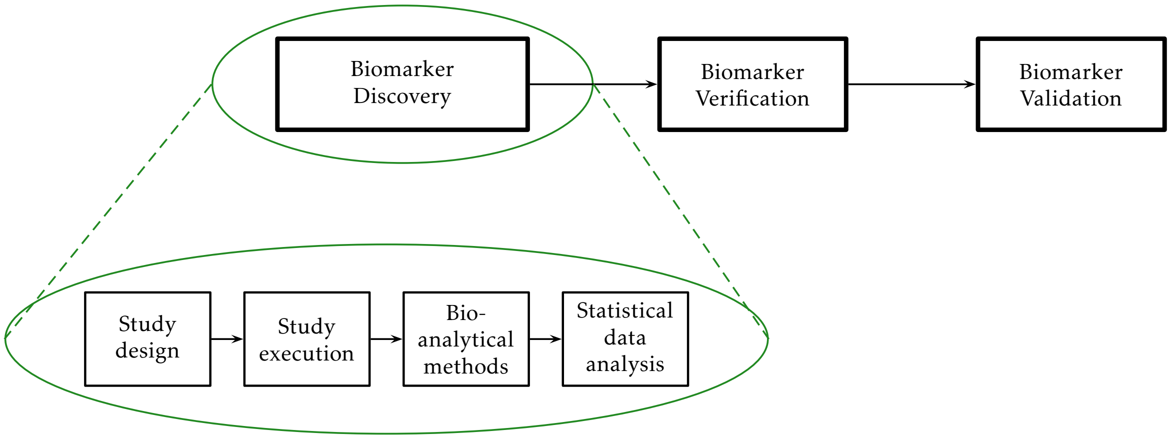


Figure 3.1: Schematic representation of the biomarker discovery process.

and investigated using data mining and various statistic tools in order to extract new potential biomarker candidates with clinical value in humans. The aforementioned biomarker discovery pipeline is described in detail in the following. [12]

Experimental Design

First of all, the general conditions and parameters of an experiment have to be defined. Therefore a variety of different experimental designs can be used in human biomarker discovery studies. An appropriate design is chosen depending on the nature of the question asked and the connected hypothesis. It can be distinguished between so-called observational or discovery based studies and hypotheses based or experimental studies. [13, 14]

Observational studies are applied in order to get initial insights, for example in pathogenic mechanisms, and to identify potential biomarker candidates related with these processes. Furthermore, the relationships of biomarkers to clinical outcomes can be examined. Therefore one or more groups of subjects are observed by the investigator with regard to defined characteristics. Four main types of observational studies can be distinguished. First of all, there is the case-series design, which mainly results in a simple report of interesting characteristics observed in a group of patients. Another type associated with observational studies is the cross-sectional or epidemiological study design. Data of patients is collected at one time and the focus lies on what happens in a very short period of time. In addition, there are cohort and case-control studies. Same as with cross-sectional design, they are also defined by the time frame of observation. Unlike as with cross-sectional design, they involve an extended period of time and therefore are often referred to as longitudinal studies. However, cohort and case-control studies differ regarding the direction of investigation. While cohort studies are forward looking, case control designs are based on retrospective analysis, where patients, for instance, with specific medical states are compared to a control group with similar phenotypes and characteristics, who do not show this medical states. Cohort studies are based on a group of patients sharing some characteristics, who are observed over a period of time investigating the effect of certain interventions. This study design provides the advantage that each patient may serve as his own biological control and hence no additional control group is needed. This results in a reduction of noise due to the variability of compared individuals. These designs are often used for the initial investigation of new biomarkers. [3, 13, 14]

Experimental study designs are focused on the effect of an intervention, such as a particular treatment or a specific drug, on the subjects. These studies are often called clinical trials and can be classified into controlled clinical trials and trials without control. Controlled clinical trials involve the comparison of a specific treatment or an experimental drug with another treatment or drug, which may be previously accepted or a placebo. They serve the purpose to determine whether an intervention makes a difference or not. Consequently, they are often applied in order to verify and validate a potential biomarker candidate which has been detected in an observational study. [3, 14, 15]

Study Execution

After an appropriate study design has been chosen, several factors such as disease of interest, study population, sample size and type as well as the frequency of measurement have to be considered. The design of an experiment is followed by the study execution, where the sample preparation, collection, and the initial analysis steps, for example mass spectrometry (MS) analysis, take place. During this phase of an experiment it is crucial to supervise all steps in order to ensure a quality controlled, standardized study execution and to meet the requirements of a Good Clinical Practice (GCP). [3, 13, 16]

Data Mining

If all these steps have been carried out, the main process of identifying clinically relevant biomarker candidates takes place, combining data processing and data mining tasks. During this part of a biomarker discovery process, the generated data is preprocessed, e.g. transformed, normalized and outliers are removed. This is followed by an extended analysis and different data mining tasks including the identification of potential biomarker candidates by data reduction (e.g. feature selection) and classification. [3]

For the purpose of reducing the dimensionality of data and therefore an improved understanding of the complex biological datasets, several techniques can be applied. One method to achieve data reduction is by clustering. It is attempted to group data according to certain criteria, such as related expression pattern or similar concentration values. The clusters obtained may reveal which data elements or features are responsible for good class segregation. In further analysis, these elements are regarded with particular care, while other elements may be neglected in the first step. In other words, it may be helpful to pick a subset of data samples which characterizes the segregation best. Consequently, using only this subset simplifies analysis due to a reduced dataset. In general, it can be differentiated between attribute based (e.g. k-means clustering) and similarity based (e.g. hierarchical clustering) methods. [15, 17] Another commonly used method for data reduction is feature selection. The task of feature selection is crucial to identify significant variables which form the initial pool of discriminatory features and hence, reduce the amount of data. This processes can be classified into supervised, semi-supervised and unsupervised methods, depending on whether the study cohort is clearly phenotyped and the study is well defined or not. A study cohort normally consists of several data instances, e.g. patients. A variety of independent variables, so called features, as well as different response values, e.g. disease stages, are assigned to each instance. If these response values are all known, the feature selection method is called supervised. If some or all response values are unknown, the process is termed semi-supervised or unsupervised feature selection. [3, 18]

After data reduction, prediction of class memberships, so-called classification is performed. This task can either be implemented with the aid of regression tools, support vector machines, artificial neural networks or by combining feature selection and classification. However, various other classification tools are available. [15]

In the framework of combining feature selection and classification, three main categories can be distinguished: The first category are filter based algorithms, which perform feature extraction independently of the classification process. Variable selection is performed once by only taking into account intrinsic attributes of the data. Afterwards, the classification algorithm is applied to the resulting feature subset and a model is built. A possible approach for a filter based strategy is to rank all features according to a defined quality criterion (e.g. paired Biomarker Identifier (pBI) as described in 4.3) and select the top k features out of it. Another example is null hypotheses testings, an univariate filter method based on the calculation of the p value, a sample dependent measurement which serves as evaluation measure for the discriminatory ability of variables. It can be applied on unpaired samples and dependent samples. [3]

Secondly, there are wrapper based methods. These types of algorithms generate different subsets of features based on a previously defined search procedure and evaluate them, using the extracted features to train and test a specific classification model. [3, 19]

The third class are the so-called embedded algorithms. Same as the wrapper based approaches, the embedded ones include a profound interaction between the search for a feature subset and the classification model. Support Vector Machine (SVM) or Random Forest Models (RFM) rank among them. SVM is a family of classifiers which perform a transformation of the input sample into a high dimensional space, the feature space, by applying a linear or kernel function. Then a linear hyperplane is drawn in order to separate two classes mapped in the feature space. [3, 12, 19]

All of the methods previously described can be assigned to the category of data-driven approaches for biomarker identification. They aim to investigate the underlying structure of high throughput datasets using various statistical, data mining or classification tools. However, another approach for biomarker identification is termed knowledge-driven and makes use of existing knowledge bases in order to facilitate the understanding of disease and underlying biochemical processes. Commonly used examples of knowledge-driven approaches are Protein-Protein Interactions (PPI), pathway analysis or text mining. [15]

PPI, non-random physical contact between two or more proteins, play an important role in the understanding of biological processes at a molecular level. It is of high interest to identify, interpret and modulate the activity of PPIs in order to predict protein and domain interactions or to explain important functional modules within proteins. PPIs are often modeled in networks. [12, 15]

The analysis of pathways provides another form of knowledge to be integrated in biomarker identification. Series of chemical reactions in cells as well as involved proteins can be linked and recorded as a pathway map. These pathway informations are usually stored in huge databases such as the Kyoto Encyclopedia of Genes and Genomes (KEGG) [20]. However, there is a huge variety of databases and pathway-related resources to choose from. [12, 15]

3.2.2 Biomarker Verification

The biomarker discovery pipeline, described in section 3.2.1, usually produces a huge amount of putative candidates. These identified biomarker candidates may in fact discriminate two classes of interest (e.g. diseased and healthy patients) and therefore meet the requirement of the initial hypothesis, but they may not all pass the validation step. For reasons of costs, time and resources it is of high interest to attenuate this pool of biomarker candidates before validation for example in a clinical trial is performed. In order to ensure that only the most promising biomarker candidates continue in the discovery process and therefore are validated, a thorough analysis of the identified candidates has to be performed. [16]

The verification task may comprise various individual steps, including a preselection of the biomarker candidates based on additional information on the individual metabolites. Therefore, a metabolite database, such as the KEGG database [20], can be used. An example is given by Lewis et al. [1]. After the identification of putative biomarker candidates for a myocardial infarction (MI) according to the pipeline depicted in 3.1, the identified metabolites are prioritized by eliminating already known biomarkers which play a role in the investigated disease. In addition, patients undergoing catheterization without the indication of a MI were examined and the resulting biomarker candidate pools were compared. Putative biomarkers occurring in both of the resulting candidate pools are eliminated for validation step. After the reduction of the putative biomarker candidates, validation is performed. [1]

3.2.3 Biomarker Validation

Subsequent to the identification and verification of putative biomarker candidates, the findings have to be validated in order to be approved as biomarkers for clinical use. Often, experimental studies are performed in order to validate the results and findings of an observational study [14]. The results of the initial and the validation study are compared. Common objective evaluation methods to determine the quality and the power of identified biomarker candidates include statistical measures such as sensitivity and specificity or the area under the Receiver Operating Characteristics (ROC) curve. However, a profound evaluation of a methods performance is enabled if larger sample sets which cover a broad range of patients or populations are used. Usually not all study frameworks allow to have a second independent group, due to for example financial budged or complicated measurement conditions. In this case evaluation has to be performed on a single patient group. Therefore an approach, referred to as cross validation can be used. The basic dataset is split into two or more groups depending on the cohort size. One group is used for training the method, and the other one for testing it afterwards. If there are more than two groups, this process is performed multiple times, whereas each time one subset is used to test and the remaining subsets are used to train the method. However, study design, e.g. independent or paired samples, always has to be considered when a statistical test is performed. [3, 15, 21]

3.3 Biological Interpretation of Identified Biomarkers

One challenging task in the field of biomarker discovery is constituted by the biological and biochemical interpretation of the identified biomarker candidates. This task comprises the functional annotation of the experimental findings in order to identify the most likely pathways the biomarker candidates are involved in and therefore allow to draw conclusions about their roles in pathological processes. There are a lot of biological databases available, covering data about pathway structures as well as details on molecular functions, structures and interactions. Information can be retrieved from these biological knowledge bases with the aid of different explorer tools (e.g. cPath [22]). The KEGG database [20] or the Gene Ontology (GO) [23] provide examples of integrative biological databases. [3, 12]

A commonly applied database for mapping the most likely pathways of putative biomarkers and for their biochemical interpretation is the KEGG. This database is used for systematic analysis of chemical substances, biological pathways, genomes, diseases and drugs. It currently consists of 16 main databases linked to each other, which can be classified into the four categories systems information, including `KEGG PATHWAY`, genomic information, chemical information, including `KEGG COMPOUND` and health information, including `KEGG DISEASE`. An overview of the KEGG resource is provided in table 3.1 [24]. Molecular networks comprising molecular interaction, reaction and relation networks, provide a basis for the interpretation of systemic functions of single cells and whole organisms. These knowledge is manually added to the database and can be depicted in pathway maps, whereas the map fields can be linked to other database entries. [12]

Table 3.1: The overview of the KEGG integrated database resource consisting of 16 databases [24]

Category	**Database**	**Content**
Systems information	KEGG PATHWAY	KEGG pathway maps
	KEGG BRITE	BRITE hierarchies and tables
	KEGG MODULE	KEGG modules
Genomic information	KEGG ORTHOLOGY (KO)	Functional orthologs
	KEGG GENOME	KEGG organisms (complete genomes)
	KEGG GENES	Genes and proteins
	KEGG SSDB	GENES sequence similarity
Chemical information	KEGG COMPOUND	Small molecules
	KEGG GLYCAN	Glycans
	KEGG REACTION	Biochemical reactions
	KEGG RCLASS	Reaction class
	KEGG ENZYME	Enzyme nomenclature
Health information	KEGG DISEASE	Human diseases
	KEGG DRUG	Drugs
	KEGG DGROUP	Drug groups
	KEGG ENVIRON	Health-related substances

3.4 Biomarkers in Cardiovascular Diseases

Cardiovascular diseases, including the MI remain one of the major causes of mortality in western countries [25]. Therefore, an early diagnosis of cardiac symptoms is crucial in order to ensure a successful treatment and hence reduce the mortality rate. Biomarkers, such as various serum proteins or the imaging technique ECG play an important role in the diagnosis of cardiac diseases. [26]

The process leading to an acute myocardial infarction (AMI) can be classified into several conditions, starting with endothelial cell injury and inflammation, leading to plaque formation and thrombogenesis and subsequently to an imbalance in the blood circulation. This results in an ischemia which, when prolonged, causes cell necrosis and consequently a MI. [27, 28]

As cell necrosis is the direct antecedent step in this pathogenesis, biomarkers which tag this condition are throughout valuable in the diagnosis of AMI and a lot of research has been done in this field. Commonly used metabolic myocardial necrosis marker include creatine-kinase muscle and brain subunits (CK-MB) and cardiac troponin (troponine I). The metabolite creatine-kinase (CK) is an enzyme occurring mainly in the skeletal and cardiac muscles. The enzyme has three isomers, the skeletal muscle fraction, the brain and the cardiac muscle fraction CK-MB. [28]

The metabolite troponin is a protein complex which is located on thin skeletal muscle filaments as well as on cardiac muscle fibres. It consists of three subunits, troponin C, the calcium binding component, troponin T, the tropomyosin binding unit and troponin I, the inhibitory component. Considering these three subunits, troponin C has the identical structure for skeletal muscle and for cardiac muscle tissue and thus is not specific for cardiac tissue necrosis. Troponin T and I show different isoforms for skeletal and cardiac muscles and therefore may be useful for the indication of a MI. However, troponin I, referred to as cardiac troponin, has not yet been isolated from skeletal muscles and is therefore uniquely specific for cardiac muscle tissue. [28, 29]

In general, cardiac troponin tends to be the marker of choice, because of its higher myocardial tissue specifity than CK-MB. Cardiac troponin is released into the blood stream 6 to 8 hours after a MI and CK-MB can be detected in the blood after about 4 hours. They both reach their peek values within 12 to 24 hours after AMI. However, an increased level in both of the mentioned proteins indicates an AMI and often both serum levels of the two proteins are assessed for the diagnosis of a MI. [28, 30]

Another protein which may serve as indicator of cardiac disease is the heart fatty acid-binding protein (H-FABP). It is a low molecular weight protein located in cardiac tissue as well as in brain, kidney and skeletal muscle in lower concentrations. This protein is released early into the cytosol after a MI and cell necrosis. Consequently, an increased concentration can be measured as early as 30 minutes after cardiac injury and concentration peaks occur at 6 to 8 hours after a MI. However, even though it allows to draw conclusions about a MI, if cardiac tissue is still in a pre-necrotic state, it is only used in combination with other markers for diagnosis of AMI. [26, 28]

However, the mentioned biomarkers do all have relatively late peaks and are not suitable for the early diagnosis of AMI. In 2008 Lewis et al. conducted a study for the identification of early markers for myocardial injury. They investigated changes in metabolites circulating in the blood and identified several of them to be thoroughly involved in metabolic pathways triggered by a MI. Alternations in purine and pyrimidine metabolisms as well as in tricarboxylic acid cycle (citrat cycle) and pentose phosphate pathway were observed. Some of the identified metabolic changes were transient, including the changes in malonic acid and alanine concentrations, while others were persistent over the measurement time frame including the changes in hypoxanthine, acontic acid, trimethylamine N-oxide and threonine. In addition, a transmyocardial enrichement pattern was detected for metabolites related to myocardial anaerobic metabolism, such as lactic acid, succine acid and adenosine triphosphate (ATP) degradation products hypoxanthine and adenosine monophosphate (AMP). The study cohort included patients undergoing planned myocardial infarction (PMI). In addition tests on patients undergoing catheterization were performed in order to verify the specificity of the identified biomarker candidates. The metabolites tryptophan, tyrosine and phenylalanie were found to be associated with cardiac catheterization and hence are not specific for a MI. [1]

4 Methods

4.1 Identification of Early Biomarkers in Cardiovascular Disease

The data used for all calculation and analysis steps in this thesis was collected by Lewis et al. [1] in 2008. The study that was performed in order to generate the data dealt with the identification of early metabolic biomarkers in myocardial injury.

The study investigated metabolite level changes in a cohort of 36 patients undergoing an alcohol septum ablation in order to treat symptomatic hyperthrophic obstructive cardiomyopathy (HOCM). This treatment induces the same metabolic processes in a body as a spontaneous myocardial infarction (SMI). The study cohort was divided into a deviation and a validation group, whereby data from 17 patients from the deviation cohort is used for further analysis in the present master's thesis. Within the study, blood was drawn from the patients at different points in time: first, directly prior to the treatment to get the baseline values (at t_0=0), then ten minutes after starting the ablation (t_{10}=10), and then at one, two, four and twenty-four hours (t_{60}=60, t_{120}=120, t_{240}=240, t_{1d}=1440) after the treatment. This serial sampling design allowed each patient to serve as his own biological control. In total, a group of 210 metabolites – carefully selected with regard to biological relevance and diversity as well as biophysical characteristics in order to ensure a broad coverage – was investigated in the blood samples by applying the MS analysis. The resulting metabolic biomarkers were compared to those of patients with SMI. In a further verification step, blood samples from patients undergoing catheterization without having the induction of myocardial infarction are examined and metabolite concentration levels are determined. This allows to verify the specificity of the putative biomarker candidates. [1]

4.2 Data Preprocessing

The data provided by Lewis et al.'s study is available as a table containing the metabolite concentration data of the examined patients for the measurement time t_i, with $i = \{0, 10, 60, 120, 240, 1440\}$ providing the minutes of sample drawing after the initial treatment. Time t_0 indicates the reference measurement performed shortly before the treatment and thus allows to reduce the effect of intervariabilty between the patients. The data is preprocessed according to [1] and contains no outliers.

For the data analysis performed in [1], concentrations of 245 metabolites from 17 patients from the deviation cohort are considered with regard to all measurement times. This data is ordered with respect to a predefined structure in order to fit the analysis algorithm. An example of the input structure is shown in table 4.1. The column `Classlabel` indicates the measurement time, with labels 1, 2, 3, 4, 5, and 6 corresponding to the times t_0, t_{10}, t_{60}, t_{120}, t_{240}, t_{1d}. The column `Names` includes the patient ID, the measurement times and the patient number p_i with $i = \{1, 2, ...17\}$, while the capital letter D indicates the deviation cohort. The ID number is again provided in column `ID` and has the values represented by i. All further columns present the different metabolites with their concentration values, respectively.

Table 4.1: Schematic representation of the structure of the input table used within the calculation and visualization algorithm. The column `Classlabel` takes values from one to the maximum number of measurements and indicates the groups of measurements which belong to one point in time. The column `Names` consists of a text string providing information about the cohort the sample belongs to. In this case, D stands for deviation cohort. Furthermore, the table gives an overview of the measurement times, indicated by t_i, where i states the time of the sample drawing after a medical intervention. In addition, the string contains the patient identification number p_i. In the column `ID`, the patient identification number is again given as single number. All further columns contain the measured metabolites and their concentration values, respectively.

Classlabel	**Names**	**ID**	**Metabolite 1**	**Metabolite 2**	**...**
1	D_t0_p1	1	1.34E+4	7.07E+3	...
1	D_t0_p2	2	1.26E+4	4.03E+3	...
...	...	...	...	...	...
1	D_t0_p17	17	5.00E+3	9.76E+3	...
2	D_t10_p1	1	1.28E+04	3.78E+03	...
2	D_t10_p2	2	1.56E+04	4.45E+03	...
...	...	...	...	...	...
2	D_t10_p17	17	1.23E+4	7.99E+3	...
3	D_t60_p1	1	1.32E+4	3.78E+3	...
3	D_t60_p2	2	1.12E+4	6.83E+3	...
...	...	...	...	...	...

The dataset used for the identification of potential biomarker candidates, the network visualization and the subsequent analysis include 245 metabolites in total. To extract a manageable size of metabolites for a further analysis and graphical representation, which in addition show an increased potential to serve as a biomarker for the given purpose, several preprocessing steps are necessary.

A main task in this process is the handling of missing values. Due to the already performed removal of outliers, not all measured datasets are complete. It may be the case that several metabolite concentration values of a patient at a certain point in time are not present, or that some metabolites at certain times are entirely missing. To overcome this problem, metabolites at all measurement points of all patients showing more than a defined percentage of missing values are completely excluded from the dataset. The remaining missing values are replaced by the median of the present concentration values of the respective metabolite at the respective time. This preprocessing step results in a table that includes a different setup of metabolites for each time.

In order to compare the datasets and, subsequently, the metabolites at a given time, the metabolites presented in the tables at all measurement points have to be the same. For this reason, each metabolite which does not occur in the tables at every measurement point is removed. The applied preprocessing results in tables with a reduced number of metabolites to be analyzed for all points in time.

4.3 Feature Selection by the Paired Biomarker Identifier

In order to extract potential biomarker candidates out of all investigated preprocessed metabolites and to ensure a proper and well recognizable network visualization, feature extraction and classification has to be performed. The aim is to reduce the number of the preprocessed metabolites. Therefore a feature selection method is applied.

In the study performed by Lewis et al. [1], a model to extract and prioritize features as proposed by Baumgartner et al. [31] was used. Since the study makes use of a longitudinal design which comprises sample drawing from one patient at different times and is done for multiple patients, the samples are paired. Therefore, the main feature extraction algorithms which are commonly used in diagnostic tests cannot be used in this case.

The feature extraction and classification method proposed by Baumgartner et al. [31] is called pBI and demonstrates a univariate feature selection method developed to identify metabolites that can be used as biomarker candidates due to their high predictive value. It provides a categorization tool to classify metabolites (e.g. in blood samples) into three classes of weak, moderate and strong predictors with respect to their ability to predict physiological or pathogenic states of a body. To achieve a biologically feasible prioritization of metabolites, the pBI value for a paired test problem is based on the following four statistical determinants: discriminantory ability (DA), magnitude, variance and the direction of changes. [31]

First, the DA is defined as the percent change of metabolite levels in a group in one direction versus the baseline. The DA can take on values in the range of $[0.5, 1]$. For further calculation, these values are rescaled to meet a range of $[0, 1]$. Second, the magnitude and variance are represented by the biological effect term which is calculated as the median percent change in metabolite concentrations at a certain time t_i versus the baseline, divided by the coefficient of variation $\sqrt{|\Delta_{change}|/|CV|}$. Last, the direction of the changes is determined by the `sign()` function. [31]

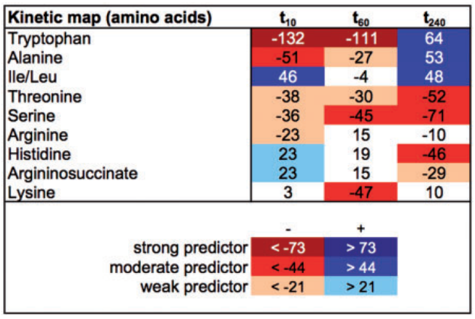
The pBI is defined as follows [31]:

$$\begin{aligned} pBI &= \lambda \cdot DA^* \cdot \sqrt{\frac{|\Delta_{change}|}{|CV|}} \cdot sign(\Delta_{change}) \\ \Delta_{change} &= \begin{cases} \Delta & \text{if} \Delta \geq 1 \\ -\frac{1}{\Delta} & \text{else} \end{cases} \end{aligned} \tag{4.1}$$

hereby, a coefficient of variation (CV) is set to 1 by default if CV > 1 in order to solely interpret datasets with a smaller variance than one as a positive biological effect.

For the classification of the identified metabolites, Baumgartner et al. [31] defined three different cutoffs based on the DA measure and displayed them in a 2D pseudocolor plot. The cutoffs classify the metabolites into three categories according to their pBI scores. This allows a quick visual evaluation of the predictive value of the identified metabolites. An example is displayed in figure 4.1. [31]

In contrast to [31], this thesis uses the pBI score measure to preselect putative biomarker candidates from all identified metabolites. Thus, the pBI measure is applied to the preprocessed data in a second intervention step. The metabolites that remain after the removal and substitution of missing values are reduced by calculating the pBI scores at each measurement. Afterwards, the metabolites are ranked with respect to their scores at each point in time and a threshold level is defined to include only those metabolites in the tables that have values higher than said threshold. The metabolites with scores higher than the defined cutoff at each measurement are



Kinetic map (amino acids)	t_{10}	t_{60}	t_{240}
Tryptophan	-132	-111	64
Alanine	-51	-27	53
Ile/Leu	46	-4	48
Threonine	-38	-30	-52
Serine	-36	-45	-71
Arginine	-23	15	-10
Histidine	23	19	-46
Argininosuccinate	23	15	-29
Lysine	3	-47	10

	-	+
strong predictor	< -73	> 73
moderate predictor	< -44	> 44
weak predictor	< -21	> 21

Figure 4.1: Example of a kintec map of amino acids on PMI data at 10, 60 and 240 minutes after the myocardial injury using the pBI score. Red color increments indicate decreasing levels and blue indicate increasing levels. [31]

combined and ranked from highest to lowest. A parameter is defined which shows the total number of metabolites that should be used for further calculations and the analysis. Subsequently, only a defined number of metabolites with top-ranked pBI scores is included in the resulting dataset.

In order to obtain an overview of the pBI scores of the metabolites at the different measurements, barplots are created for each time. The bars show the metabolites ranked by the absolute values of their respective pBI scores. The blue bars indicate metabolites with negative value scores while the red bars denote metabolites with positively valued scores. An example of such a barplot is depicted in 4.2, including the top 20 metabolites at one measurement time. With the help of the barplot diagrams of the pBI scores of the measured metabolites, putative biomarker candidates can be preselected.

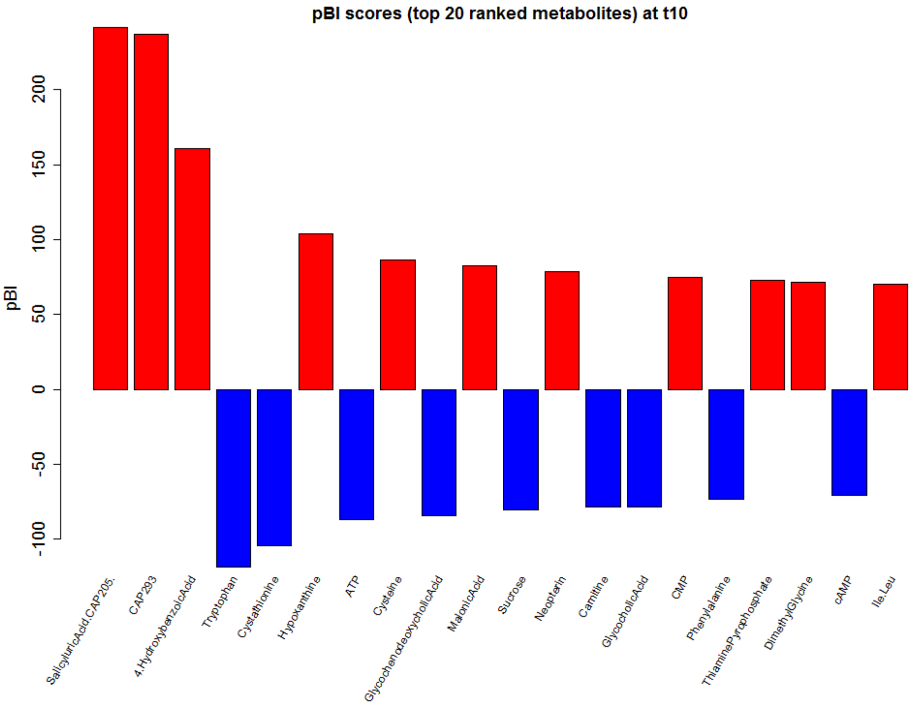


Figure 4.2: Exemplary representation of a barplot of pBI scores at measurement time t_{10}. The red bars indicate metabolites with positive pBI score values and the blue bars show the scores of metabolites with negative score values. All are ranked according to the absolute value of their pBI scores.

4.4 Network Representation

To model the relations between the identified putative biomarker candidates, a network-based approach is chosen. This allows to construct a network graph by means of the given metabolites. For this, the basics of graph theory have to be considered and they are thus described in the following subchapter.

4.4.1 Graph Theory and Network Representations

A graph $G = (V, E)$ is a mathematically definable object that consists of a finite set of elements called vertices $V = \{v_1, v_2, ..., v_n\}$ or nodes together with a set of elements called edges E, whereby each edge element represents a connection between a pair of vertices (v_i, v_j). Graph objects can be used to model dependencies and relations between the elements of a system. They can be depicted in a diagram, whereby the nodes can be represented as points in a plane and the edges as lines joining two points (see figure 4.3). [32]

Figure 4.3 depicts an undirected, unweighted simple graph $G = (V, E)$ with five vertices and six edges, whereby $V = \{v_1, v_2, v_3, v_4, v_5\}$ and $E = \{(v_1, v_2), (v_1, v_3), (v_1, v_4), (v_2, v_4), (v_3, v_5), (v_4, v_5)\}$. Each edge joins two vertices and is denoted by specifying its two vertices.

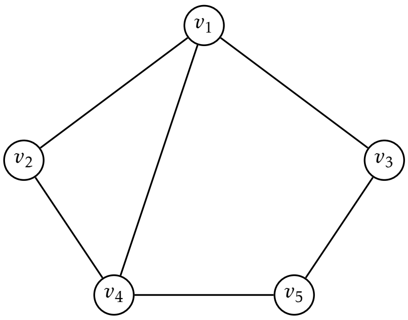


Figure 4.3: Schematic representation of an undirected, unweighted graph $G(V, E)$. The graph consists of five vertices $V = \{v_1, v_2, v_3, v_4, v_5\}$ and six edges $E = \{(v_1, v_2), (v_1, v_3), (v_1, v_4), (v_2, v_4), (v_3, v_5), (v_4, v_5)\}$ connecting these vertices.

When discussing graph theory, some specific terms are required to describe the relations and properties of the nodes and edges. To begin with, adjacent nodes are nodes that are connected by at least one edge. Node v_1 and node v_4 in figure 4.3, for example, can be termed adjacent nodes. Two vertices that are joined by an edge can be called endpoints of this edge, e.g. nodes v_1 and v_3 are the endpoints of edge $\{v_1, v_3\}$. Moreover, an edge $\{v_i, v_j\}$ is incident with the vertices v_i and v_j. Two or more edges joining the same pair of vertices are called multiple edges and, if an edge joins a node to itself, this is called a loop. A graph with no multiple edges or loops is called a simple graph. The degree of a vertex v_i is defined as the number of edges incident with v_i. If all vertices of a graph have the same degree $deg(v_i)$, the graph is called regular. [32, 33]

Considering a graph as a set of vertices and edges, it can be classified into several common types. First, one can distinguish weighted and unweighted graphs, where weights (or no weights) are assigned to the edges. Furthermore, the edges can be directed, which means that they can only be traversed in a specific direction, or undirected, where traverse in either direction is possible. [4]

The networks created in this thesis are partially unweighted while others have weighted edges. However, all of them are undirected.

Concerning the network representation, one can distinguish static and dynamic representations. The main static representations to store network graphs are adjacency matrices and edge list representations. An adjacency matrix A of a graph $G(V, E)$ is defined as a $V \times V$ matrix with

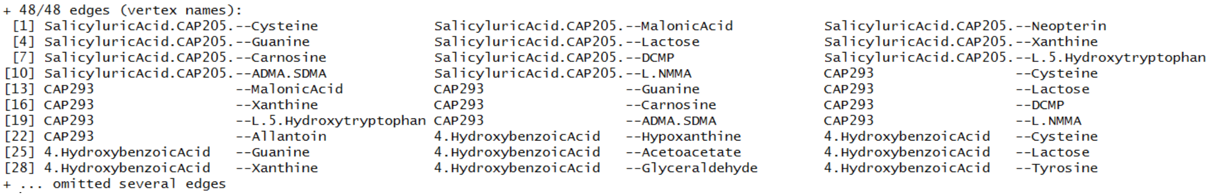
$$A(i,j) = \begin{cases} 1 & \text{if } (i,j) \in E(G) \\ 0 & \text{if } (i,j) \notin E(G), \end{cases} \tag{4.2}$$

where each entry a_{ij} represents the number of edges connecting the vertices v_i and v_j. [33, 34]

A schematic representation of the structure of an adjacency matrix of a metabolic network is depicted in equation 4.3, where the rows and columns of the matrix are given by the same metabolites, in this case M_1 - M_6. These metabolites represent the nodes of the graph. Each entry a_{ij} indicates whether there is a connection between two metabolites. When $a_{ij} = 1$, the nodes are connected; otherwise $a_{ij} = 0$ indicates that there is no edge between the two nodes M_i and M_j. The adjacency matrix depicted in 4.3 is symmetrical around the main axis, where each entry a_{ij} for $i = j$ takes the value zero. Consequently, the underlying graph has no loops and neither multiple nor weighted edges.

$$A = \begin{array}{c} \\ M_1 \\ M_2 \\ M_3 \\ M_4 \\ M_5 \\ M_6 \end{array} \!\!\begin{array}{c} \begin{array}{cccccc} M_1 & M_2 & M_3 & M_4 & M_5 & M_6 \end{array} \\ \begin{pmatrix} 0 & 1 & 0 & 1 & 1 & 1 \\ 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 1 & 0 & 1 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 \end{pmatrix} \end{array} \tag{4.3}$$

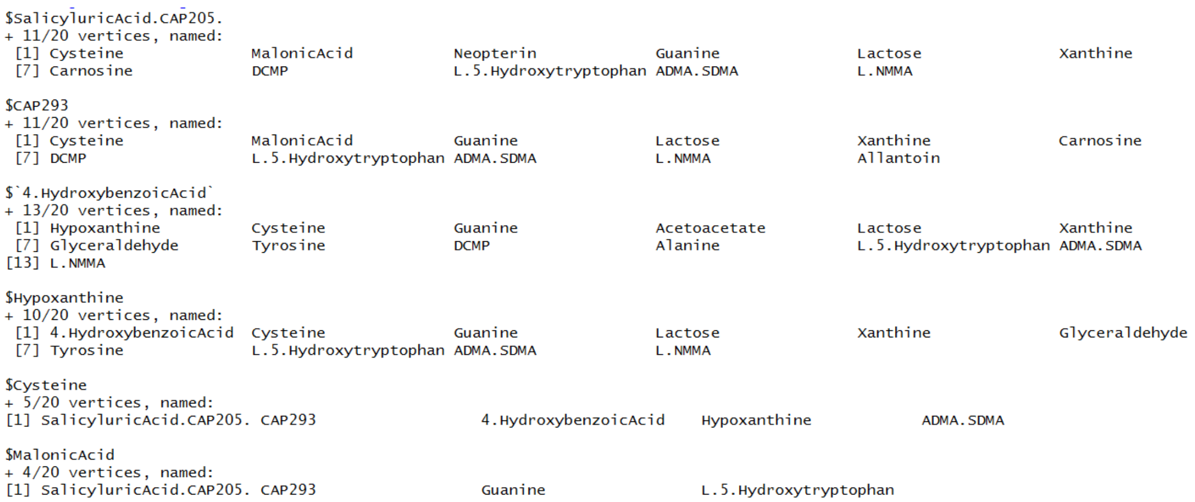
Yet another way of representing a network graph is by creating an edge list. This is a list of all edges of a graph $G(V, E)$ represented by unordered vertex pairs [33, 34]. Figure 4.4 shows an example of such an edge list. Here, the nodes are represented by the respective metabolite names and a connection between two nodes is indicated by the symbols --. The number at the top of the figure gives the total number of edges present in the respective graph, in this case there are 48 edges.



```
+ 48/48 edges (vertex names):
 [1] SalicyluricAcid.CAP205.--Cysteine              SalicyluricAcid.CAP205.--MalonicAcid           SalicyluricAcid.CAP205.--Neopterin
 [4] SalicyluricAcid.CAP205.--Guanine               SalicyluricAcid.CAP205.--Lactose               SalicyluricAcid.CAP205.--Xanthine
 [7] SalicyluricAcid.CAP205.--Carnosine             SalicyluricAcid.CAP205.--DCMP                  SalicyluricAcid.CAP205.--L.5.Hydroxytryptophan
[10] SalicyluricAcid.CAP205.--ADMA.SDMA             SalicyluricAcid.CAP205.--L.NMMA                CAP293                 --Cysteine
[13] CAP293                 --MalonicAcid           CAP293                 --Guanine               CAP293                 --Lactose
[16] CAP293                 --Xanthine              CAP293                 --Carnosine             CAP293                 --DCMP
[19] CAP293                 --L.5.Hydroxytryptophan CAP293                 --ADMA.SDMA             CAP293                 --L.NMMA
[22] CAP293                 --Allantoin             4.HydroxybenzoicAcid   --Hypoxanthine          4.HydroxybenzoicAcid   --Cysteine
[25] 4.HydroxybenzoicAcid   --Guanine               4.HydroxybenzoicAcid   --Acetoacetate          4.HydroxybenzoicAcid   --Lactose
[28] 4.HydroxybenzoicAcid   --Xanthine              4.HydroxybenzoicAcid   --Glyceraldehyde        4.HydroxybenzoicAcid   --Tyrosine
+ ... omitted several edges
```

Figure 4.4: Schematic representation of an edge list. Number at the top gives the total number of edges present in the respective graph. Metabolite names indicate nodes. The sign -- represents an edge between the two nodes at either side of the sign.

An example of a dynamic network representation is the adjacency list representation, whereby for each vertex v_i of $V(G)$, a list of vertices which are adjacent to v_i is given [34]. An example of an adjacency list is depicted in figure 4.5. Each list element can be subdivided into three parts. First, it provides the name of the regarded node. In the second line, the total number of nodes (out of all nodes present in the network) to which the considered node is connected is given. The third part consists of the names of all nodes to which the respective node is connected.



```
$SalicyluricAcid.CAP205.
+ 11/20 vertices, named:
 [1] Cysteine              MalonicAcid           Neopterin             Guanine               Lactose               Xanthine
 [7] Carnosine             DCMP                  L.5.Hydroxytryptophan ADMA.SDMA             L.NMMA

$CAP293
+ 11/20 vertices, named:
 [1] Cysteine              MalonicAcid           Guanine               Lactose               Xanthine              Carnosine
 [7] DCMP                  L.5.Hydroxytryptophan ADMA.SDMA             L.NMMA                Allantoin

$`4.HydroxybenzoicAcid`
+ 13/20 vertices, named:
 [1] Hypoxanthine          Cysteine              Guanine               Acetoacetate          Lactose               Xanthine
 [7] Glyceraldehyde        Tyrosine              DCMP                  Alanine               L.5.Hydroxytryptophan ADMA.SDMA
[13] L.NMMA

$Hypoxanthine
+ 10/20 vertices, named:
 [1] 4.HydroxybenzoicAcid  Cysteine              Guanine               Lactose               Xanthine              Glyceraldehyde
 [7] Tyrosine              L.5.Hydroxytryptophan ADMA.SDMA             L.NMMA

$Cysteine
+ 5/20 vertices, named:
[1] SalicyluricAcid.CAP205. CAP293                  4.HydroxybenzoicAcid    Hypoxanthine            ADMA.SDMA

$MalonicAcid
+ 4/20 vertices, named:
[1] SalicyluricAcid.CAP205. CAP293                  Guanine                 L.5.Hydroxytryptophan
```

Figure 4.5: Schematic representation of an adjacency list. Each list element consists of the name of the respective vertex together with the number of vertices it is connected to and the names of each vertex it is connected to.

4.4.2 Network Creation

In order to infer a network graph that represents the identified metabolic biomarker candidates of the present data and their correlations to each other, the software `R` [35] is used and the package `BiomarkeR` [2] is adapted. For the graph construction, the nodes are defined as the given analytes M, while the edges represent the chemical interaction of analyte pairs in the network, i.e. the ratios R between the metabolite concentrations M. These ratios are computed as $r_{ij} = |log_2(\frac{m_i}{m_j})|$ with $i > j$, and $m \in M, r \in R$. Subsequently, the pBI scores $s_{ij},\ s \in S$ of the logarithmic ratios R, in the following referred to as pBI*, are computed and a graph G is constructed. Here,

$$G_{ij} = \begin{cases} 1 & \text{if } |s_{ij}| > \tau \\ 0 & \text{else,} \end{cases} \tag{4.4}$$

for $i, j \in 1, ..., |M|$. Analyte pairs with ratios greater than the threshold τ are represented as edges in the graph G. [2]

A network graph is inferred for each time a blood sample was drawn from the patient. Therefore, the interactions of the metabolites, i.e. the binary logarithm ratios, are computed for each point in time. Subsequently, the pBI score measure, which is utilized as feature selection tool (pBI) and – in combination with a defined threshold value – as classification tool for metabolite-to-metabolite interactions (pBI*), is calculated for each of these ratios. For the calculation of the pBI* scores, two points in time are taken into consideration (t_i whereas $(i \neq 0)$ and t_{i-1} with $i = \{0,\ 10,\ 60,\ 120,\ 240,\ 1d\}$) in order to be able to illustrate the changes over time. The threshold τ is set to different values so as to specify the network edges for different representations. There are two different approaches for defining τ: a dynamic and a static one.

For determining the threshold τ in a dynamic way, a defined quantile is calculated for the data at each of the respective measurement points. Therefore, all pBI* scores of the logarithmic ratios are considered and the value specifying the defined quantile is calculated. The resulting value works as a threshold for the graph inference and differs for each time.

In order to specify the threshold τ as static value, a boxplot diagram of all pBI* scores of the logarithmic ratios is created. For the data at each time point, a threshold value based on a defined quantile is calculated. The mean of all of these threshold values defines a static threshold value for the graph inference at each time point.

4.5 Network Visualization

In fact, different approaches to visualize various aspects of the given data are implemented in order to ensure a thorough analysis of the provided data. The visualization methods comprise different representations of network graphs and are thus based on graphs constructed as described in section 4.4. The following sections discuss the most important graph representations realized in this thesis.

4.5.1 Graph Representation

The construction of the graphs as well as the visualizations of the resulting network graphs are both implemented in `R`. The package `igraph` [36] is used to store the resulting network graphs as graph objects and enable further analysis as well as different visualization methods. First of all, a graph is constructed for each measurement time, as described in 4.4.2 for different dynamic and static thresholds. The constructed graphs are then depicted in a simple network representation using a circle layout. The metabolites as nodes are ordered in a circle and are connected by unweighted or weighted edges according to the respective classification criterion based on their relations.

Additionally, an overview graph is constructed for different threshold values based on network inference, as described in 4.5.3. Therefore, all metabolites present in the dataset after removing the missing values are depicted in a network graph. The node and label sizes are adapted to the respective node degree by normalizing the degree value with the maximum degree value. Labels are only displayed for nodes with high degree values in order to improve readability.

4.5.2 Heatmap Representation

Besides adjacency lists as described in 4.4.1, an additional possibility of representing the connection between individual nodes – or rather the connections originating from one specific node to others in the network – is the representation as a heatmap. All values arranged in a matrix are represented as colors in order to emphasize their specific value and, consequently, identify potential patterns in the data.

In order to represent the connections originating from one node, a heatmap is constructed for each of the metabolites. Therefore, all pBI* scores of the logarithmic ratios associated with the metabolite of interest are considered. These ratios are classified into weak, moderate and strong associations according to predefined threshold values. The three different classes are then displayed in different colors, whereby the direction of the pBI* score is also taken into consideration. This results in a color scheme of light-blue and light-red for weak connections, changing to clear blue and red for moderate connection values, and resulting in dark-blue and dark-red for strong connections. The color blue indicates a negative score value, while red denotes a positive one. Figure 4.6 shows an example of the color key used for a heatmap construction based on the different threshold values.

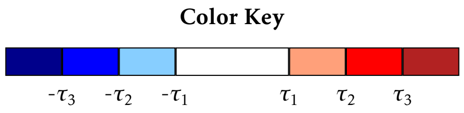


Figure 4.6: Schematic representation of a color key used for heatmap visualization. The color sections are specified by the calculated threshold values $\tau_1 = \tau_{weak}$, $\tau_2 = \tau_{moderate}$ and $\tau_3 = \tau_{strong}$, indicating either weak, moderate or strong connections between metabolites.

The threshold values for the classification are defined by different quantiles: the whole range of the pBI* score values of the logarithmic ratios is considered for all measurement times. Not only are the scores calculated for each of the points in time, but they are also averaged by computing the mean for each of the thresholds over the time of the measurement.

The pBI* score values are ordered in a matrix format, in which the rows represent all metabolites in the network without the metabolite of interest. The columns show the different measurement times t_i with $i = \{10, 60, 120, 240, 1d\}$. Each matrix entry a_{ij} represents the pBI* score of the logarithmic ratio between the metabolite of interest and another metabolite. In addition to the coloring, the values are displayed in the heatmaps.

4.5.3 Combined Network Plot

A further possibility of visualizing the changes in the networks over time and thereby highlight the dynamics of the longitudinal measurement is to combine several networks of different measurement times. Here, one considers networks created out of a defined number of metabolites and by defining a specific dynamic or static threshold at a particular measurement time for these combined network plots. The nodes of each of the networks are layered upon each other and all edges occurring in a network at a specific time are included in the combined graph. To improve visibility and allow a better interpretation of the resulting network, the strength of the edges in the network, i.e. the metabolite-to-metabolite connections, are highlighted. Therefore, weights are assigned to all edges in the graph.

There are two different approaches to calculate the weights for a combined network plot. The first concept is simply based on combining the adjacency matrices of all graphs. Each entry in an adjacency matrix represents the presence of a connection between two nodes in a binary manner. An entry in these matrices takes the number zero if there is no connection and a value of one if there is an edge between two nodes. To get an idea of how often two metabolites are connected over different measurement times, all adjacency matrices are summed up. This results in a combined adjacency matrix, in which each entry a_{ij} can take values in the range of zero to the number of networks combined. Thus, if n graphs of n different times are combined, an entry a_{ij} can take the discrete value x with $x = \{0, 1, 2, ..., n-1, n\}$. These entry values are rescaled to yield a value in the range of [0,1] in order to ensure the weight distribution is independent from the total number of graphs combined. The weight values are assigned as edge weights to the combined graph.

For the visualization of the combined plot, the edge weights are displayed as lines with different widths in the network graph. The edges with the smallest width indicate a connection that only occurs in one of the combined networks. In contrast, those edges with the highest weights are displayed as the thickest line. Hence, the thickness of an edge serves as an indicator for the frequency of its occurrence in all of the combined network graphs.

The second approach for calculating the weights of the combined plot is to dynamically weight the edges based on the pBI* scores of the logarithmic ratios. In order to ensure comparability between the different points in time and to enable a combination of the dynamically weighted edges, each pBI* score value is normalized by the maximum score of each measurement time, before it is assigned as weight to the corresponding edge. This results in weights which can take values in the range from [0,1]. This makes them comparable between the graphs at different measurement times. Similar to the procedure using binary weights, continuous weights are entered in the respective adjacency matrix and summed across time. Thus, if n networks are combined, each matrix entry can take a continuous value $y \in [0, n]$. These values y are again rescaled so that they are in the range of $[0, 1]$ in order to achieve edge weights which are independent from the number of combined graphs. Afterwards, the values are assigned as edge weights to the combined graph.

The edge weights are displayed as lines with different line widths in the graph plot: here, the line widths rise in accordance with increasing values. Similar to discrete weights, the thickness of an edge serves as an indicator for its frequency of occurring in all of the combined network graphs.

4.5.4 Interactive 3D Network Visualization

In order to achieve an interactive 3D representation of the inferred network graphs, the `R` package `rgl` [37] is used. A network of one measurement time is always displayed with its nodes arranged in a circle in the x-y plane. The networks of different measurement points which should be included in the 3D plot are located at different times at the z-axis (which denotes the time).

The resulting 3D plots can be investigated as interactive 3D plots in `R` or exported as standard file format, e.g. as png (portable network graphics) file. In addition, they can be saved as gif (graphics interchange format) file in order to save and use the representation as an animation.

4.6 Biological Verification and Interpretation

The networks constructed provide information about the connection and, therefore, the interaction between two metabolites. In order to check and verify the findings based on the network graphs, the KEGG database [20] is used. The respective metabolites and putative biomarker candidates are used as search term in the `KEGG Compound` database and the linked `KEGG Pathway` entries are investigated with respect to their interactions with other metabolic pathways and, finally, diseases associated with them. Some exemplary results are summed up in a table.

5 Results

5.1 Experimental Setup

Before performing calculations and applying the visualization tools described in chapter 4, the setup used for the different network visualization methods has to be defined. For this reason, all parameters used for the network visualization as well as the choice of threshold values for the metabolite preselection and network inference are described in this section.

The first preprocessing step includes the handling of missing values in the initial dataset. In a first step, metabolites containing more than a defined percentage of missing values are excluded from the dataset. This threshold percentage is defined as 60% for further calculations. Thus, metabolites with more than 60% of their values missing are completely removed from the dataset.

The second parameter defined is the threshold value for the metabolite preselection based on the pBI scores at each point in time. Figure 5.1 shows the distribution of the pBI scores for each point in time. The thick black line in the middle of a box indicates the q50 threshold while the upper limit of a box presents the q75 threshold. The preprocessing only includes those metabolites which are considered moderate predictors. Therefore, the threshold value q75 is defined as a dynamic threshold, selecting only those metabolites with pBI scores higher than the dynamically calculated cutoff τ_2 at each time for further processing.

The third parameter defined is the number of metabolites that remain in the dataset after the preprocessing pipeline. This parameter is set to 20 for further calculation and analysis steps for reasons of visualization and simplicity. Consequently, the total number of metabolites is reduced from 245 metabolites before the intervention to 20 after the preprocessing.

5.2 Visualization of the Paired Biomarker Identifier Scores

As described in the previous section, the pBI scores are used to identify and classify potential metabolic biomarker candidates at each measurement. For the computation of the pBI scores, the metabolite concentration data for all times t_i, with $i = \{0, 10, 60, 120, 240, 1d\}$ is considered separately and each dataset is compared to a reference dataset at t_{i-1} with $i \neq 0$.

The distributions of the absolute values of the pBI scores at all points in time are depicted as a boxplot diagram in figure 5.1. The cutoffs for the dynamic classification are set to q50, q75 and q90 for weak, moderate and strong predictors, respectively, whereby the quantile for the given percentage value is calculated and used as cutoff value. These values provide the basis for calculating the static threshold values depicted as colored horizontal lines in the boxplot. The green line shows the static threshold for weak predictors with $\tau_1 = 33.64$, as determined by calculating the mean of all dynamically computed 50% quantile values at the different points in time. The orange line shows the moderate static threshold with $\tau_2 = 52.21$, giving the result of the average value of all 75% quantiles. The red line depicted in the diagram denotes the strong threshold with $\tau_3 = 70.29$, determined by considering all 90% quantiles for calculating the mean. Keeping in mind that a high pBI score indicates a good biomarker candidate, these thresholds should provide a basis for the preselection and classification of metabolites which may serve as biomarkers for myocardial injury. In this thesis, the moderate static cutoff τ_2, which presents the mean value of the different moderate dynamic cutoffs, is applied as threshold value for the metabolite preselection.

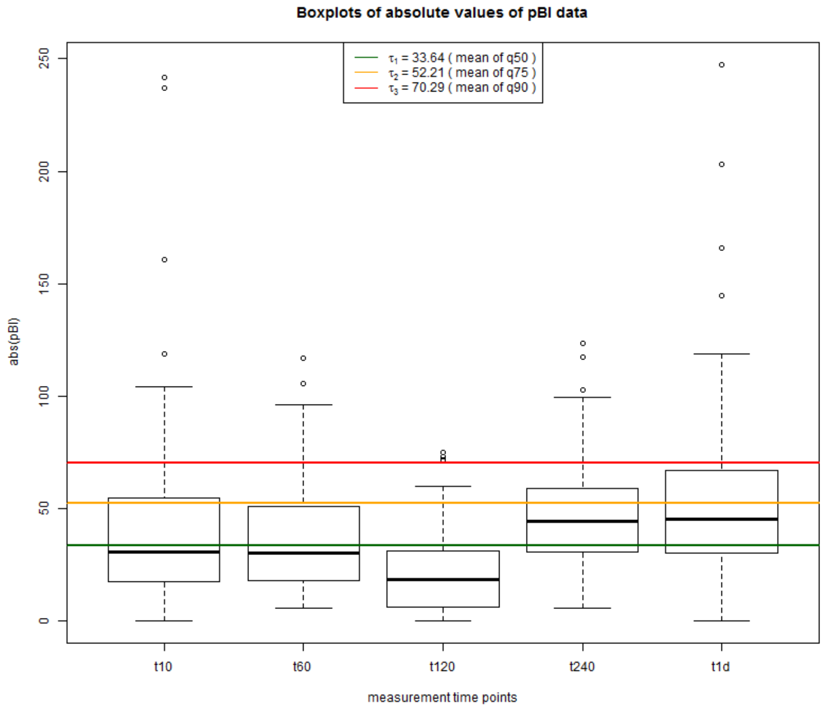


Figure 5.1: Boxplot of absolute pBI scores for measurement times t10 minutes to t1d. The box indicates the range of quantile 25 to quantile 75, the median is depicted as thick black line in the middle of the box. Threshold τ_1 gives the mean of the median values (quantile 50) of each dataset as low static cutoff. The moderate static threshold τ_2 is calculated by averaging the third quartile q75 and the strict static threshold τ_3, the mean of quantile 90 of each dataset is depicted as red line.

Figures 5.2 and 5.3 depict bar plots of the metabolites with the highest pBI scores for each of two selected times. For the sake of proper visibility, only those 20 metabolites which yielded the highest scores are shown in the plots. There are no pBI score values for the measured data at t0 because this measurement acts as a reference dataset for the measurement at t10 and, consequently, no separate pBI scores were calculated. The bar plot diagrams for t120, t240 and t1d are shown in the appendix A.1.

Positively valued pBI scores are depicted as red and negatively valued scores as blue bars, whereas positive and negative values indicate an increase and decrease in concentration from the reference time to the measurement time. It can be seen that for measurement t10, the metabolites salycyluric acid and catabolite activator protein (CAP) 293 yield the highest pBI scores with score values higher than 200 in a positive range. In contrast, at t60, the metabolites guanine and AMP are top-ranked with absolute score values in the range of 100 to 150, both in the positive and negative direction, respectively. From these observations, the conclusion can be drawn that for measurement at different times, different metabolites are ranked important and can be classified as potential biomarker candidates based on pBI as the feature selection tool.

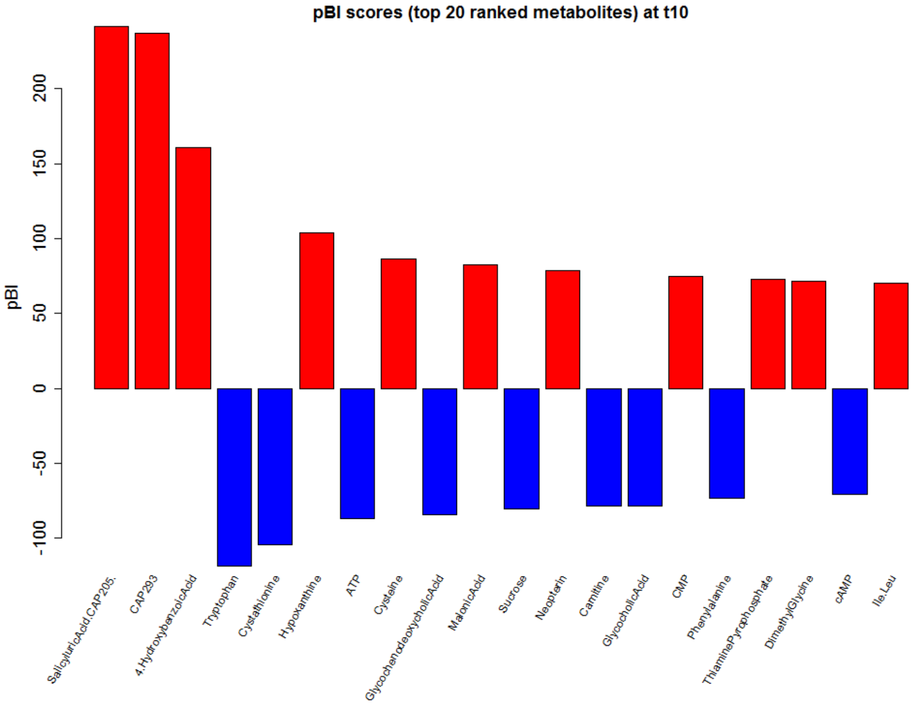


Figure 5.2: Bar plot of the top 20 metabolites, ranked according to their respective pBI score values at measurement t10. The red and blue bars indicate a positive and negaive pBI score value, respectively.

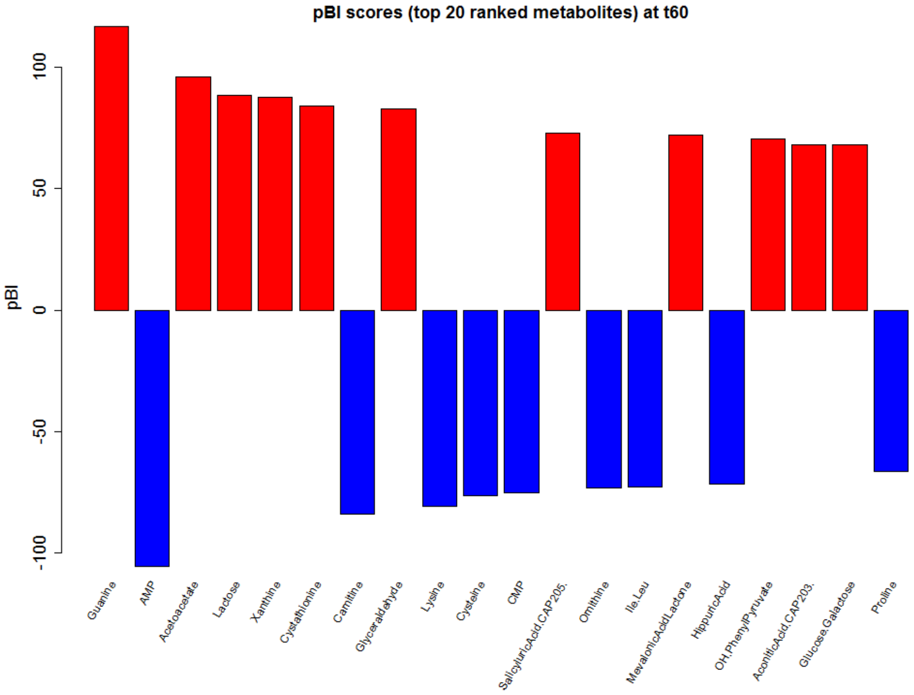


Figure 5.3: Bar plot of the top 20 metabolites, ranked according to their respective pBI score values at measurement t60. The red and blue bars indicate a positive and negaive pBI score value, respectively.

In order to provide an overview of the data, a graph containing all metabolites which remain in the dataset after the removal of the missing values is constructed, i.e. the parameter which defines the number of metabolites to be included is set to maximum. Therefore, networks of all metabolites created for different points in time with a threshold level of q90 are combined as described in 4.5.3. The resulting graph is depicted in figure 5.4 based on the force-directed Fruchterman-Reingold layout [38]. The size of each node depends on the respective node degree, i.e. the number of edges connected to a node. Nodes with higher degrees are depicted with bigger node sizes and larger node labels.

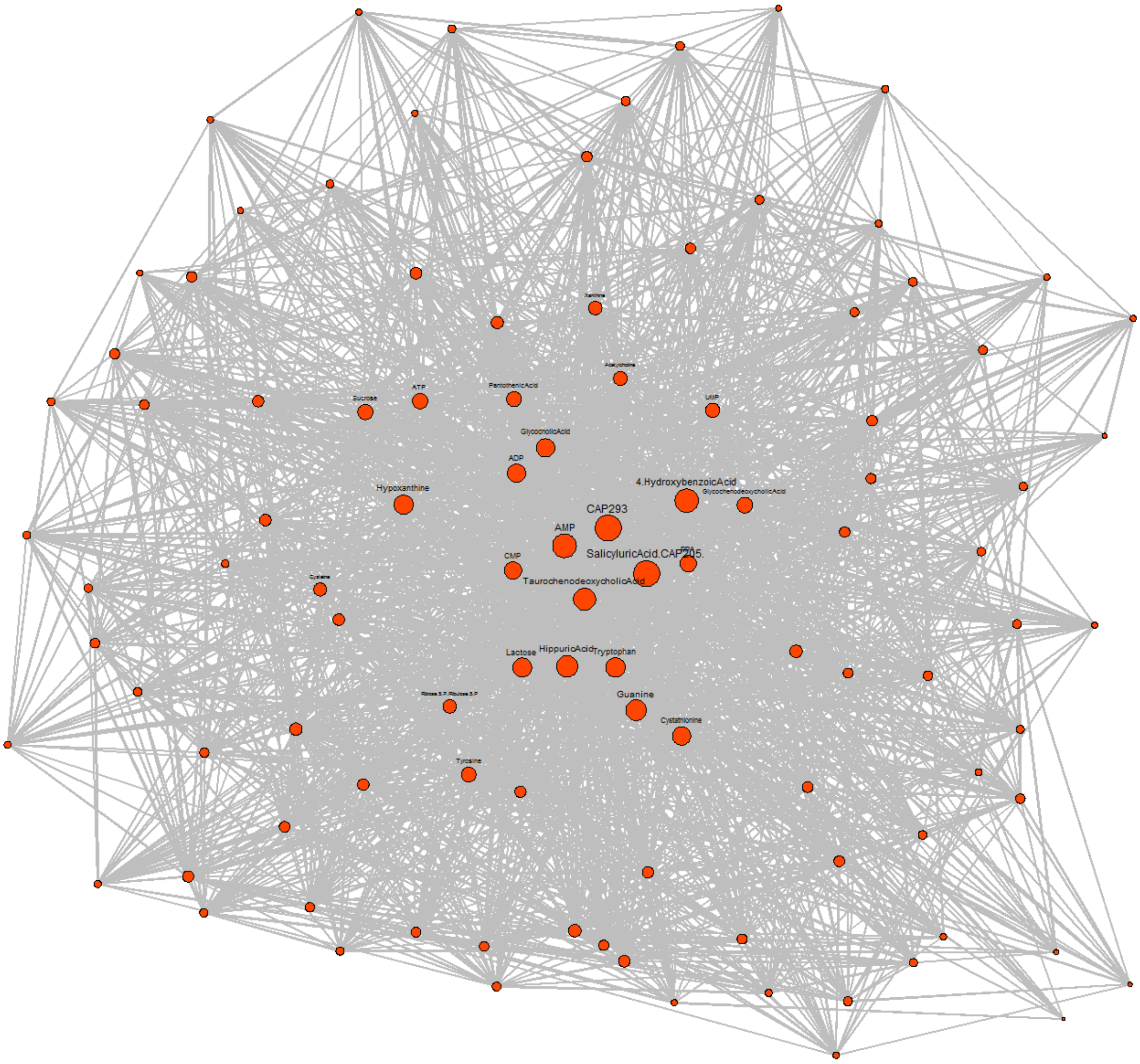


Figure 5.4: Combined graph of networks at all points in time constructed with threshold q90. The network contains all metabolites of the given dataset. The node size and node label size are adapted to the respective node degree.

5.3 Network Visualization

In the following sections, results of the different network visualization tools, as described in section 4.5, are presented. The networks are inferred using two different approaches. The first method applies dynamic thresholds for deciding whether an edge between two nodes should be created, while the second uses static threshold values.

The dynamic threshold values are defined with q50, q75 and q90, thereby specifying the quantiles which determine the cutoff value. The three cutoff values are calculated for each of the five different measurement points. Subsequently, 15 dynamic cutoff values are determined in total. Table 5.1 provides an overview of all dynamic threshold values calculated for the given dataset.

The static threshold values for the network generation are based on the dynamically calculated cutoff values. In order to determine the static cutoffs, the mean of the dynamic cutoff values is calculated for each time, resulting in three different static thresholds for q50, q75 and q90. The resulting static threshold values are illustrated in table 5.1.

Table 5.1: Summary of dynamic and static threshold values at three different cutoffs, based on the quantiles q50, q75 and q90. The dynamic thresholds are given for each measurement time and each of the cutoff quantiles. The static thresholds are calculated as the mean values of the dynamic thresholds of all time points for each of the cutoff quantiles.

quantile		q50	q75	q90
cutoff		τ_1	τ_2	τ_3
dynamic	t10	42.61	76.57	104.10
	t60	31.26	58.60	79.66
	t120	20.56	43.17	56.58
	t240	55.86	80.38	109.81
	t1d	53.48	80.58	115.01
static	mean	40.75	67.86	93.03

For all metabolite-to-metabolite ratios yielding a pBI* score higher than the respective cutoff value, an edge between the two metabolites is created.

Figure 5.5 depicts the distributions of the absolute pBI* score values of the binary logarithm ratio of the metabolites. Each box represents the score values at one of the measurements, which serves to compare the value range over time. The thick black line inside a box represents the dynamic 50% quantile threshold while the upper bound of a box denotes the dynamic 75% quantile threshold at each of the measurements. The horizontal colored lines represent the three static threshold values τ_1, τ_2 and τ_3. The depicted values are summarized in table 5.1.

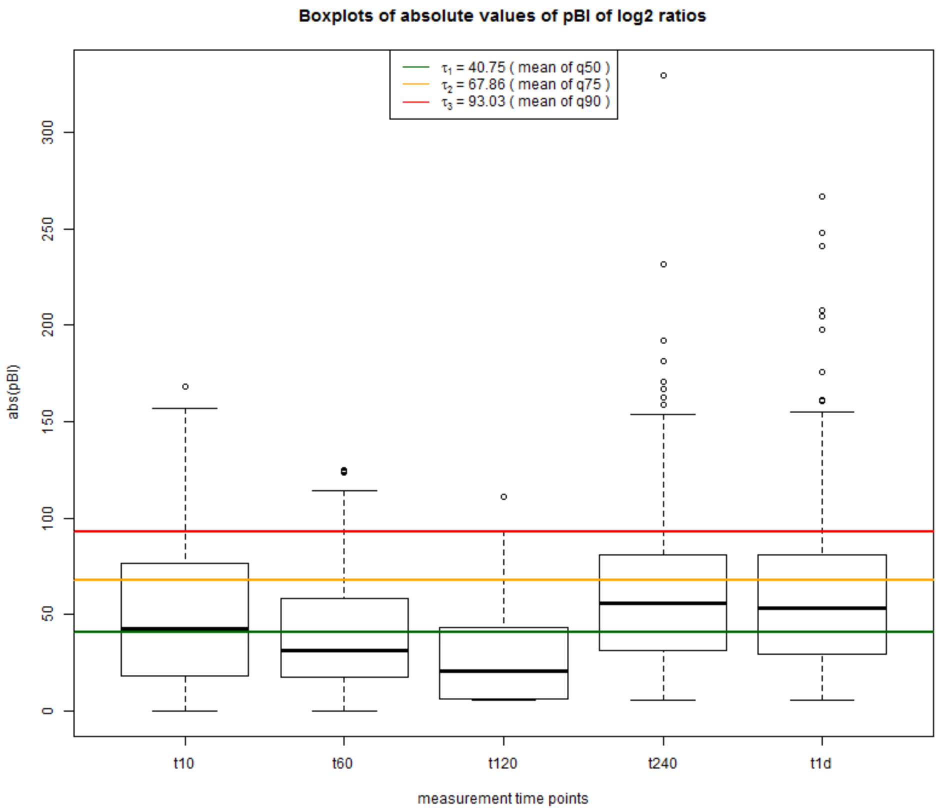


Figure 5.5: Boxplot of absolute pBI scores of binary logarithm ratios of metabolite concentrations at different measurement points. The boxes represent the values at each of the respective times. The horizontal lines represent different static threshold values given by the mean of the different dynamic cutoff values at each point in time for each threshold level. τ_1, τ_2 and τ_3 represent the mean of the dynamic threshold values at q50, q75 and q90, respectively.*

5.3.1 Heatmap Representation

In order to have an indicator for the strength of the connection between two metabolites given as nodes in a network, the pBI* scores of the binary logarithm of the metabolite-to-metabolite ratios are considered.

For each of the metabolites representing the nodes in the network graph, a heatmap is constructed. The edges with their perspective score values (representing the connections to other nodes of the network) are depicted for each point in time. Figures 5.7 and 5.8 show examples of a heatmap representation of two metabolites present in the networks. Each heatmap shows the strengths of the connections of one metabolite compared to a set of other metabolites present in the network graph. The classification of the strength of a connection is based on the pBI* score measure of the logarithmic ratios between metabolite concentrations. Further examples of heatmaps representing the connections originating from other metabolites are included in the appendix A.2.

The color key in a heatmap plot visualizes the three different classes the values can be assigned to depending on their respective pBI* score. These classes are defined by the static threshold values as given in table 5.1. Then, these values are applied both on negative and positive scores, resulting in a color scheme from light to dark blue for negative scores and light to dark red for positive scores (always depending on the respective static threshold value). The color key that indicates the strength of the connection between two metabolites based on the calculated cutoff values is depicted in figure 5.6. A positive score value colored in one of the red color grades indicates that the metabolite considered in the heatmap has a higher concentration value than the one it is connected to. In reverse, it can be seen that a negative score value indicates that the metabolite has a smaller concentration value than that to which it is compared. Consequently, the color scheme allows a quick assessment of the relation of two metabolites' concentration values at different measurement times.

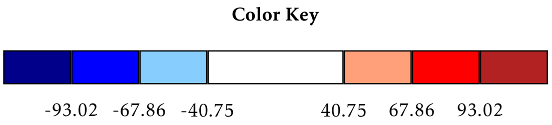
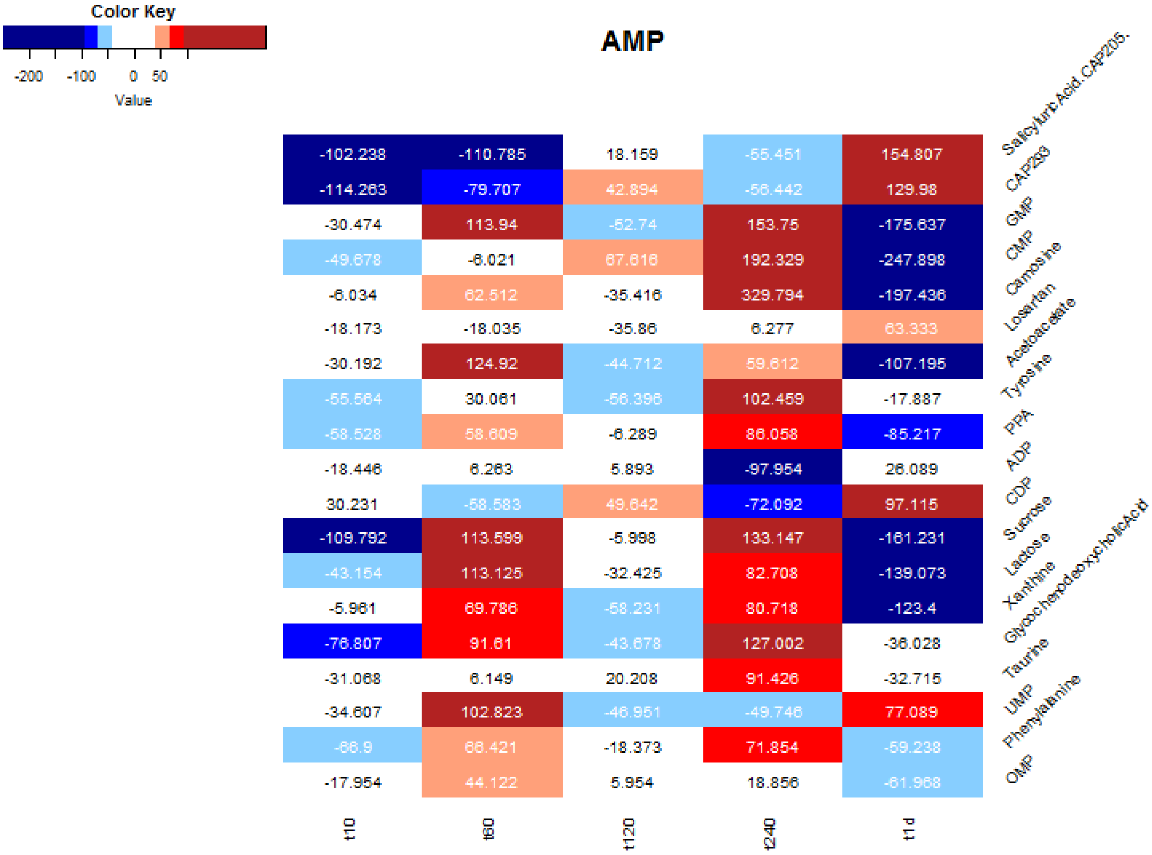


Figure 5.6: Color key for a heatmap visualization with static threshold values for weak, moderate and strong connections. The color sections are specified by the calculated threshold values τ_{weak} = 40.74, $\tau_{moderate}$ = 67.86 and $\tau strong$ = 93.02, and applied in both negative and positive direction.

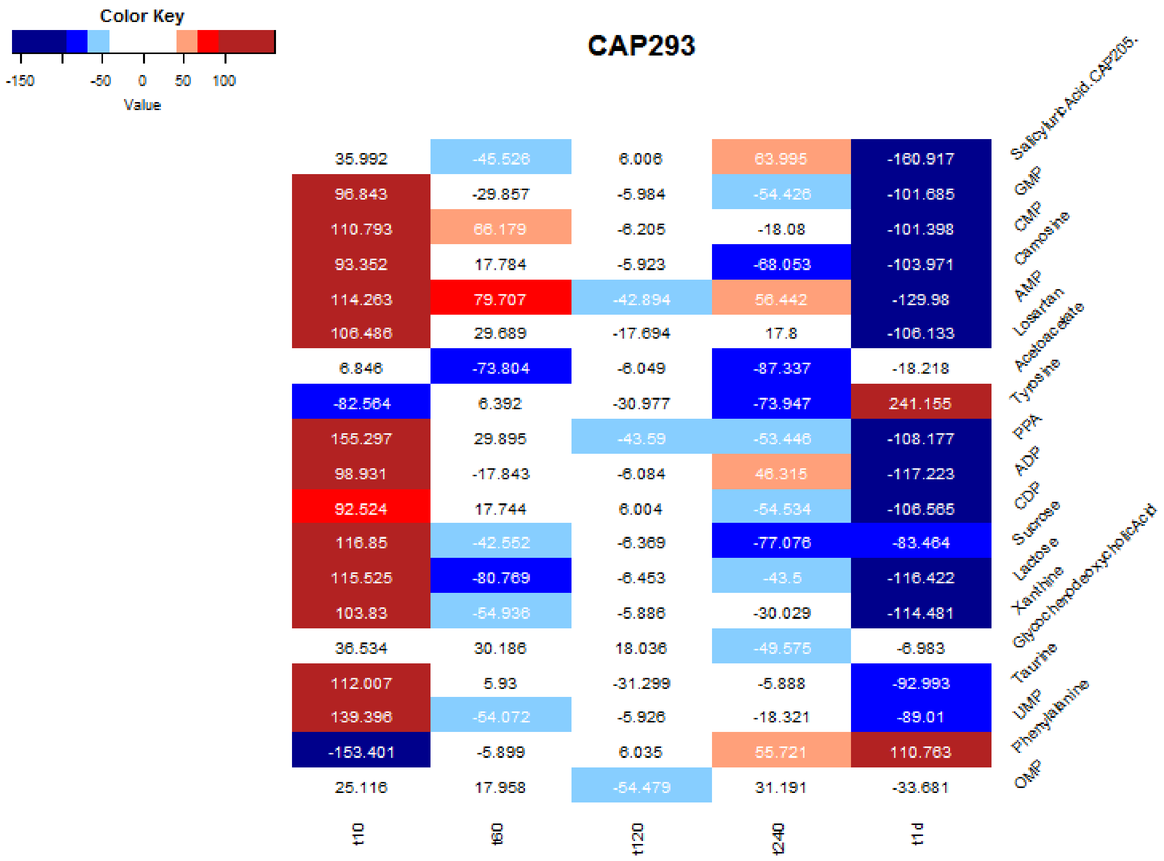
Figure 5.7 shows the edges originating from node AMP for the different networks at different times. Darker colors in the heatmap indicate higher pBI* score values. The latter are provided as numbers in the respective field for each connection at a measurement point. The majority of scores classified as strong occurs at t240 and t1d, whereas early stages after the treatment do not yield as high values.



	t10	t60	t120	t240	t1d
SalicyluricAcid.CAP205.	-102.238	-110.785	18.159	-55.451	154.807
CAP203	-114.263	-79.707	42.894	-56.442	129.98
GMP	-30.474	113.94	-52.74	153.75	-175.637
CMP	-49.678	-6.021	67.616	192.329	-247.898
Carnosine	-6.034	62.512	-35.416	329.794	-197.436
Losartan	-18.173	-18.035	-35.86	6.277	63.333
Acetoacetate	-30.192	124.92	-44.712	59.612	-107.195
Tyrosine	-55.564	30.061	-56.396	102.459	-17.887
PPA	-58.528	58.609	-6.289	86.058	-85.217
ADP	-18.446	6.263	5.893	-97.954	26.089
CDP	30.231	-58.583	49.642	-72.092	97.115
Sucrose	-109.792	113.599	-5.998	133.147	-161.231
Lactose	-43.154	113.125	-32.425	82.708	-139.073
Xanthine	-5.961	69.786	-58.231	80.718	-123.4
GlycochenodeoxycholicAcid	-76.807	91.61	-43.678	127.002	-36.028
Taurine	-31.068	6.149	20.208	91.426	-32.715
UMP	-34.607	102.823	-46.951	-49.746	77.089
Phenylalanine	-66.9	66.421	-18.373	71.854	-59.238
OMP	-17.954	44.122	5.954	18.856	-61.968

Figure 5.7: Heatmap representing the metabolite connections originating from metabolite AMP. Numbers in the map field give the pBI scores of the binary logarithmic ratios between AMP and the respective metabolites. Colors indicate the strength of the connection directly associated with the absolute value of the score, whereby a high score indicates a strong connection and is represented by a dark color.*

Figure 5.8 shows the connections originating from metabolite CAP293. In contrast to the connections classified as strong of metabolite AMP (see figure 5.7), the strong connections of metabolite CAP293 are concentrated at t10, thus indicating a metabolite activity in an early stage after the treatment. In addition, 13 out of 19 connections are classified as strong at t1d.



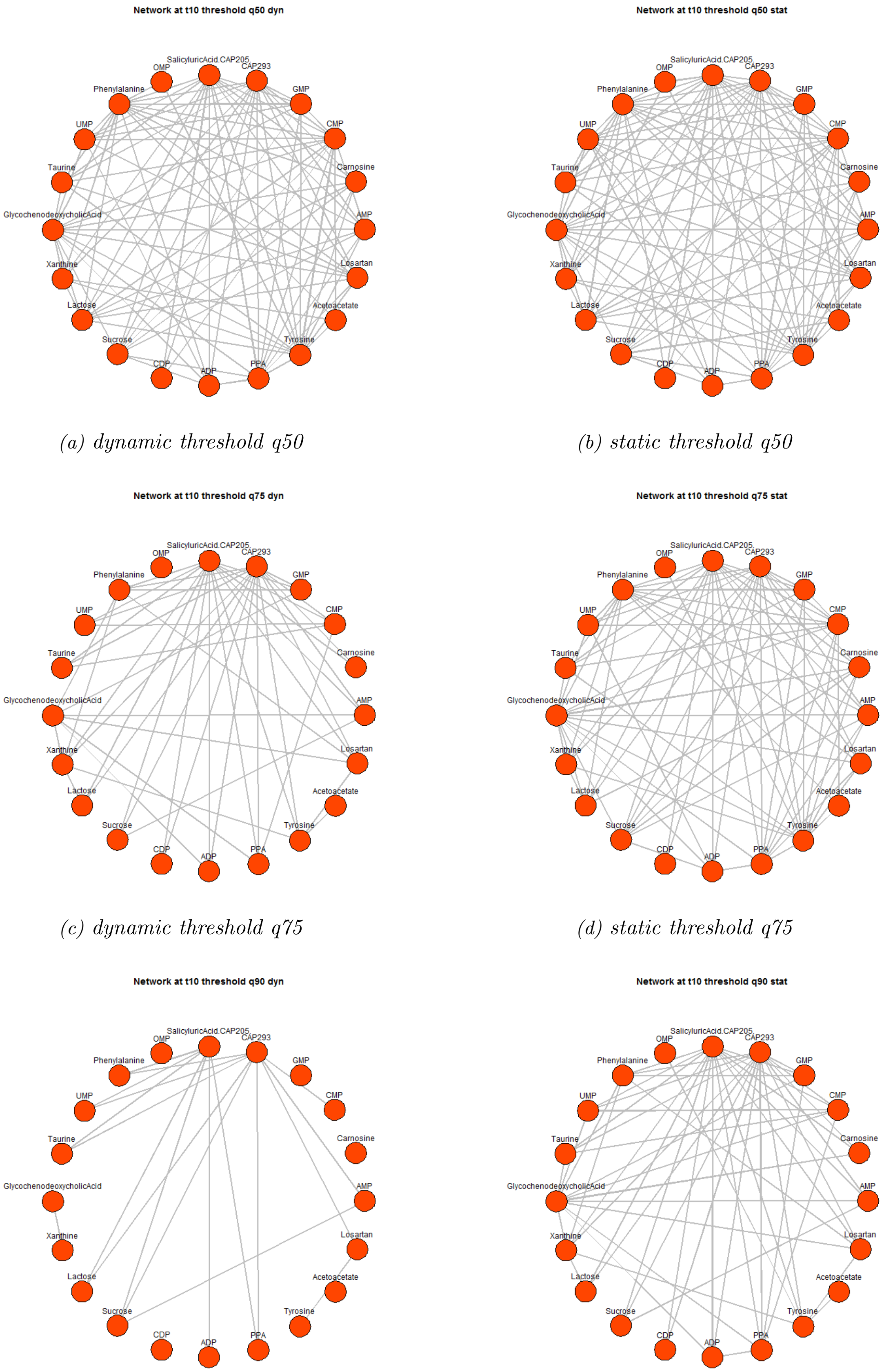
t10	t60	t120	t240	t1d	
35.992	-45.526	6.006	63.995	-160.917	SalicyluricAcid.CAP205.
96.843	-29.857	-5.984	-54.426	-101.685	GMP
110.793	66.179	-6.205	-18.08	-101.398	CMP
93.352	17.784	-5.923	-68.053	-103.971	Carnosine
114.263	79.707	-42.894	56.442	-129.98	AMP
106.486	29.689	-17.694	17.8	-106.133	Losartan
6.846	-73.804	-6.049	-87.337	-18.218	Acetoacetate
-82.564	6.392	-30.977	-73.947	241.155	Tyrosine
155.297	29.895	-43.59	-53.446	-108.177	PPA
98.931	-17.843	-6.084	46.315	-117.223	ADP
92.524	17.744	6.004	-54.534	-106.565	CDP
116.85	-42.552	-6.369	-77.076	-83.464	Sucrose
115.525	-80.769	-6.453	-43.5	-116.422	Lactose
103.83	-54.936	-5.886	-30.029	-114.481	Xanthine
36.534	30.186	18.036	-49.575	-6.983	GlycochenodeoxycholicAcid
112.007	5.93	-31.299	-5.888	-92.993	Taurine
139.396	-54.072	-5.926	-18.321	-89.01	UMP
-153.401	-5.899	6.035	55.721	110.763	Phenylalanine
25.116	17.958	-54.479	31.191	-33.681	OMP

Figure 5.8: Heatmap representing the metabolite connections originating from metabolite CAP293. Numbers in the map field denote the pBI scores of the binary logarithmic ratios between CAP293 and the respective metabolites. Colors indicate the connection strength that is directly associated with the absolute value of the score: a high score indicates a strong connection and is thus represented by a dark color.*

5.3.2 Comparison of Different Dynamic and Static Thresholds

In this section, network graphs constructed with different threshold values, as summarized in table 5.1, are presented and compared.

In figure 5.9, the network graphs with dynamically calculated (left side) and static threshold values (right side) for the different cutoff values q50, q75 and q90 at t10 after the treatment are depicted. All of the networks are displayed in a circle layout with the nodes representing the same metabolites. The networks constructed with dynamic thresholds (a), (c) and (e) show a decreasing number of edges for higher threshold values. Networks created with the same static threshold values (b), (d) and (f) yield more edges in each of the graphs for time t10 in comparison to the dynamic threshold values.



Network at t10 threshold q50 dyn

Network at t10 threshold q50 stat

Network at t10 threshold q75 dyn

Network at t10 threshold q75 stat

Network at t10 threshold q90 dyn

Network at t10 threshold q90 stat

(a) dynamic threshold q50

(b) static threshold q50

(c) dynamic threshold q75

(d) static threshold q75

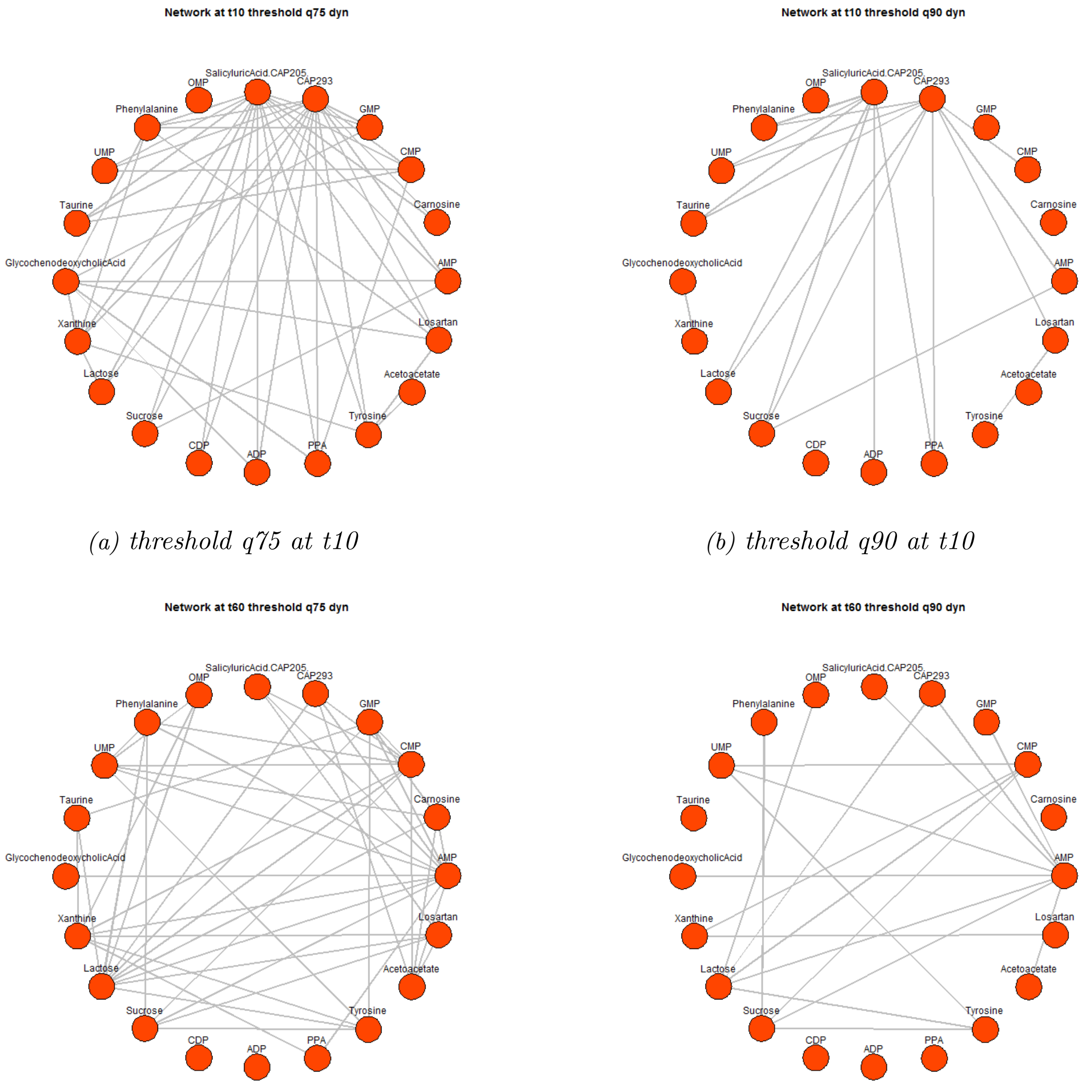
(e) dynamic threshold q90

(f) static threshold q90

Figure 5.9: Network plots in a circle layout at different dynamic and static threshold values, with (a), (c) and (e) representing the different dynamically calculated thresholds and (b), (d) and (f) representing the different statically calculated threshold values at time t10.

5.3.3 Comparison of Networks at Different Measurement Times with Dynamic Thresholds q75 and q90

Figures 5.10 and 5.11 illustrate networks constructed by means of two different dynamically calculated threshold values at five different measurement times t10, t60, t120, t240 and t1d, respectively. On the left side, those networks with a dynamic threshold value q75 are shown, while the right depicts networks with a dynamic threshold q90. It can be seen, that networks created with a higher threshold q90 have less edges in the respective graphs. In addition, it can be observed that the edge connections between the nodes change over time for both threshold values. A comparison of the networks at different measurements with static threshold values q75 and q90 is illustrated in the appendix A.3.1.


Network at t10 threshold q75 dyn
Network at t10 threshold q90 dyn
Network at t60 threshold q75 dyn
Network at t60 threshold q90 dyn
SalicyluricAcid.CAP205
OMP
CAP293
Phenylalanine
GMP
UMP
CMP
Taurine
Carnosine
GlycochenodeoxycholicAcid
AMP
Xanthine
Losartan
Lactose
Acetoacetate
Sucrose
Tyrosine
CDP
ADP
PPA

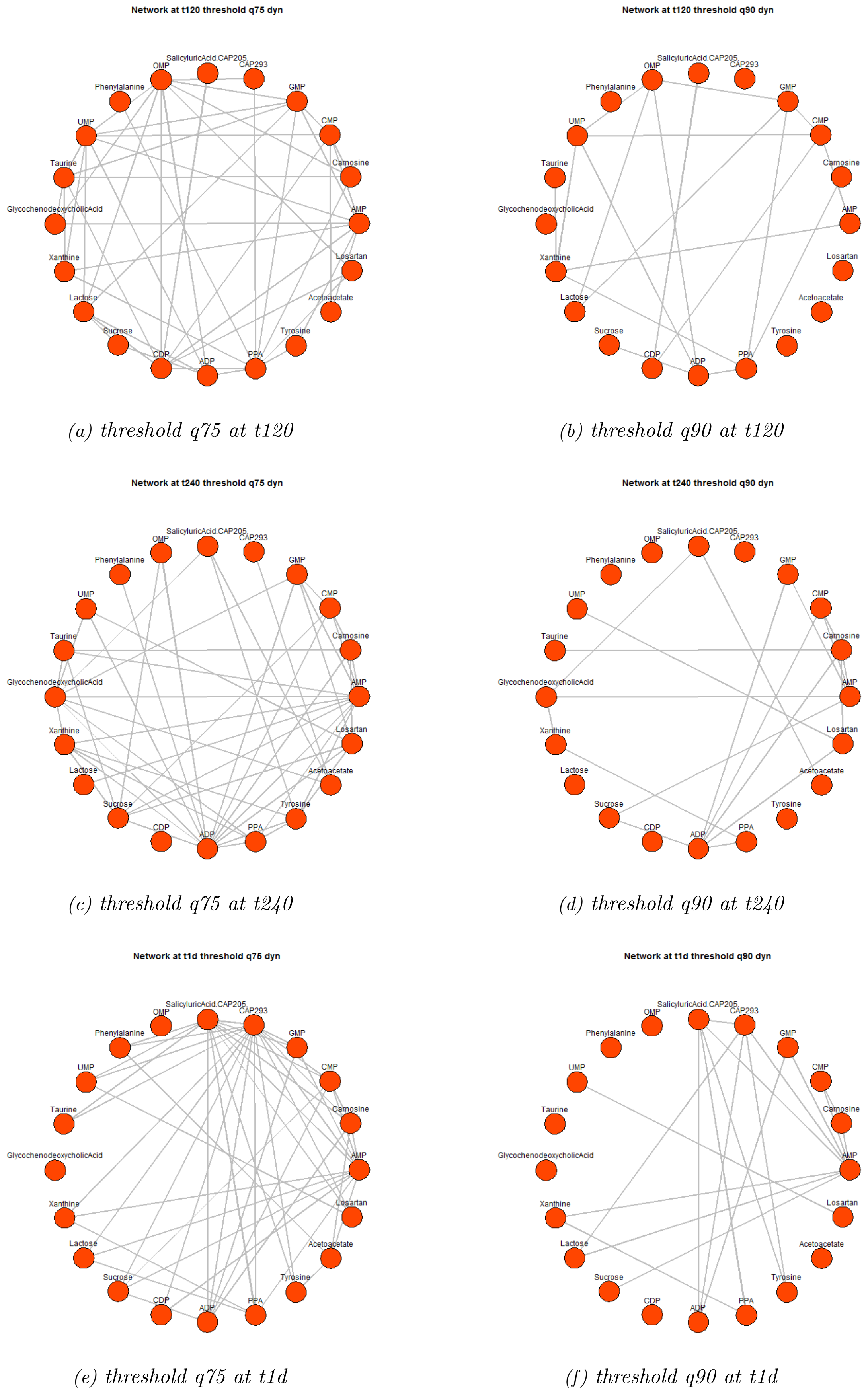
(a) threshold q75 at t10

(b) threshold q90 at t10

(c) threshold q75 at t60

(d) threshold q90 at t60

Figure 5.10: Network plots in a circle layout at dynamically calculated thresholds q75 and q90 at two different measurement times t10: (a), (b) and t60: (c), (d).



(a) threshold q75 at t120

(b) threshold q90 at t120

(c) threshold q75 at t240

(d) threshold q90 at t240

(e) threshold q75 at t1d

(f) threshold q90 at t1d

Figure 5.11: Network plots in a circle layout at dynamically calculated thresholds q75 and q90 at three different measurement times t120: (a), (b), t240: (c), (d) and t1d: (e), (f).

5.3.4 Comparison of Combined Network Plots

For the visualization of the combined network plots, all network graphs constructed for different measurement points as well as the respective threshold are considered. All edges present in any of the network graphs at any time are combined and weighted in order to represent the strength of the connection between two nodes over time.

Figure 5.12 illustrates the combined network plots for networks constructed with different dynamic thresholds: subplot (a) and (b) show the combined network plots created with q50, subplot (c) and (d) illustrate networks constructed with q75, and (e) and (f) are graphs created with a threshold value of q90. For each of the three threshold values, two different modes of edge weighting are applied. The left side of figure 5.12 depicts the combined graphs with discrete weighted edges. The edge width indicates its frequency of occurrence over the time of the measurement. To obtain the number of occurrences, the adjacency matrices of the combined networks are summed up. The resulting values are rescaled to fit a range from zero to one.

For reasons of visualization, the edges are weighted again using a fourth degree polynomial function in order to emphasize connections with high scores. To guarantee an appropriate choice of the line width, a range of $[0.25, 5]$ given in the unit of points (pt) is chosen. Mapping the value range to this line width range by using the simple forth degree polynomial

$$f(x) = ax^4 + c \tag{5.1}$$

results in the parameters $a = 4.75$ and $c = 0.25$.

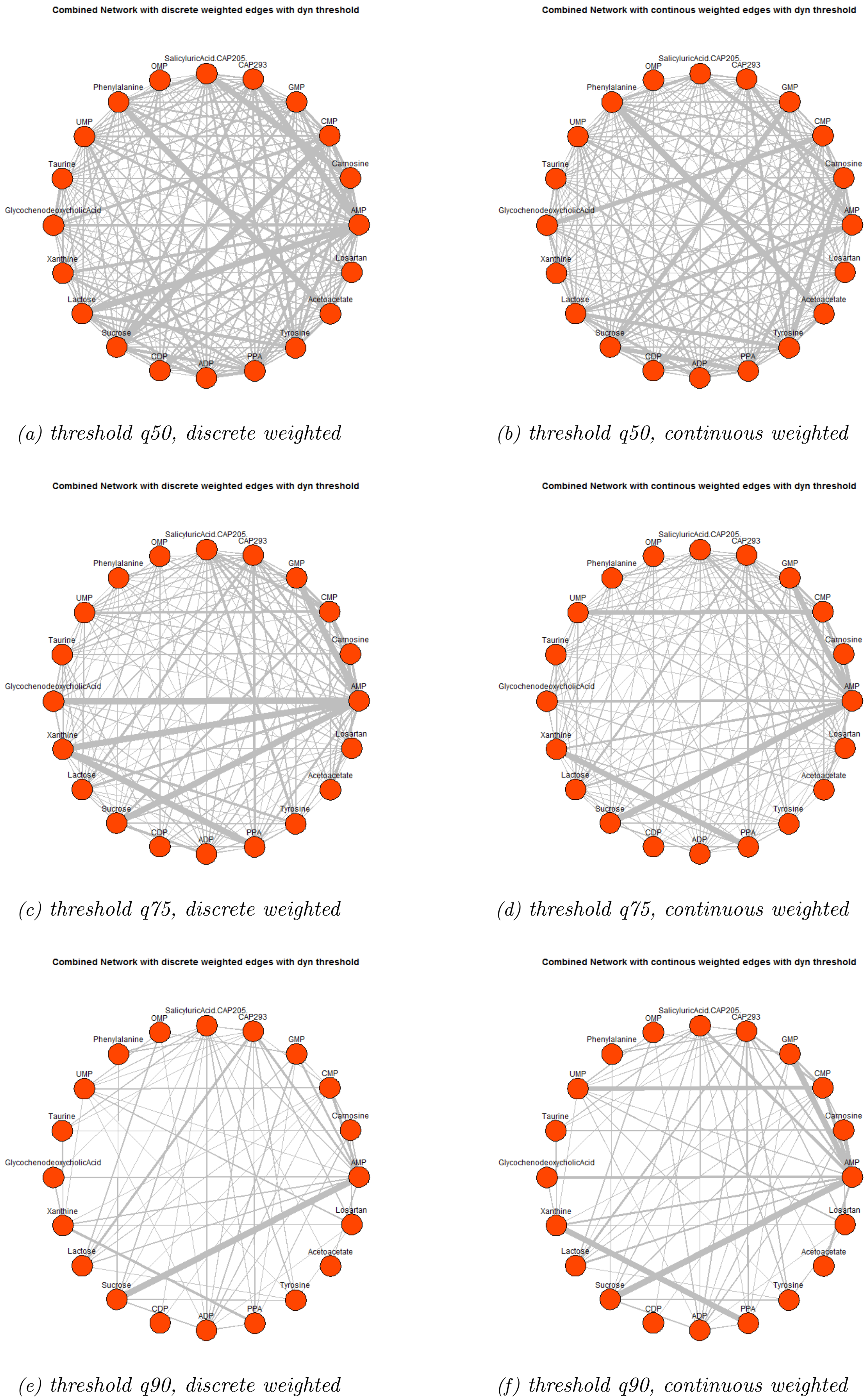
Subsequently, edges with thick line widths, as can be seen in subplot (a) of figure 5.12 with the edge connecting the nodes CAP293 and AMP, or salycyluric acid and carnosine indicate an occurrence of these connections in several of the graphs at different points in time considered for this combined plot. Edges with an occurrence in only one network of those combined feature a thin line width.

The right hand side of figure 5.12 illustrates the combined network plots for three different threshold values with edges of continuous weights. For calculating the edge widths of these graphs, the normalized pBI* score values of the ratios between the metabolites are considered and summed up. Subsequently, the resulting values are rescaled again to a range from zero to one, using equation 5.1. As a consequence, the variety of line widths used for the illustration of the combined graphs with continuous weights increases in comparison to the graphs with discrete weighted edges.

Comparing the two combined network plots for the threshold value q75 of figure 5.12, as depicted in subfigures (c) and (d), the difference between the two edge weighting modes is revealed. Subplot (c) on the left side highlights four edges as thick line, whereas in suplot (d) only two of the four edges are depicted as thick lines. A further difference is the edge connecting the nodes uracil monophosphate (UMP) and cytidine monophosphat (CMP): in subplot (c), it is present as a small line, whereas it is more dominant in subplot (d) with edge weights depending on calculated scores.

However, when the threshold values for the network construction increase, fewer edges are present in the combined network plots and less edges are depicted as thick line. Consequently, less edges are present at several points in time for higher threshold values.

The combined network graphs with discrete and continuous weighted edges for the network construction with static threshold values are depicted in the appendix A.3.2.


Combined Network with discrete weighted edges with dyn threshold
Combined Network with continous weighted edges with dyn threshold
Combined Network with discrete weighted edges with dyn threshold
Combined Network with continous weighted edges with dyn threshold
Combined Network with discrete weighted edges with dyn threshold
Combined Network with continous weighted edges with dyn threshold
SalicyluricAcid.CAP205
OMP
CAP293
Phenylalanine
GMP
UMP
CMP
Taurine
Carnosine
GlycochenodeoxycholicAcid
AMP
Xanthine
Losartan
Lactose
Acetoacetate
Sucrose
Tyrosine
CDP
ADP
PPA

(a) threshold q50, discrete weighted

(b) threshold q50, continuous weighted

(c) threshold q75, discrete weighted

(d) threshold q75, continuous weighted

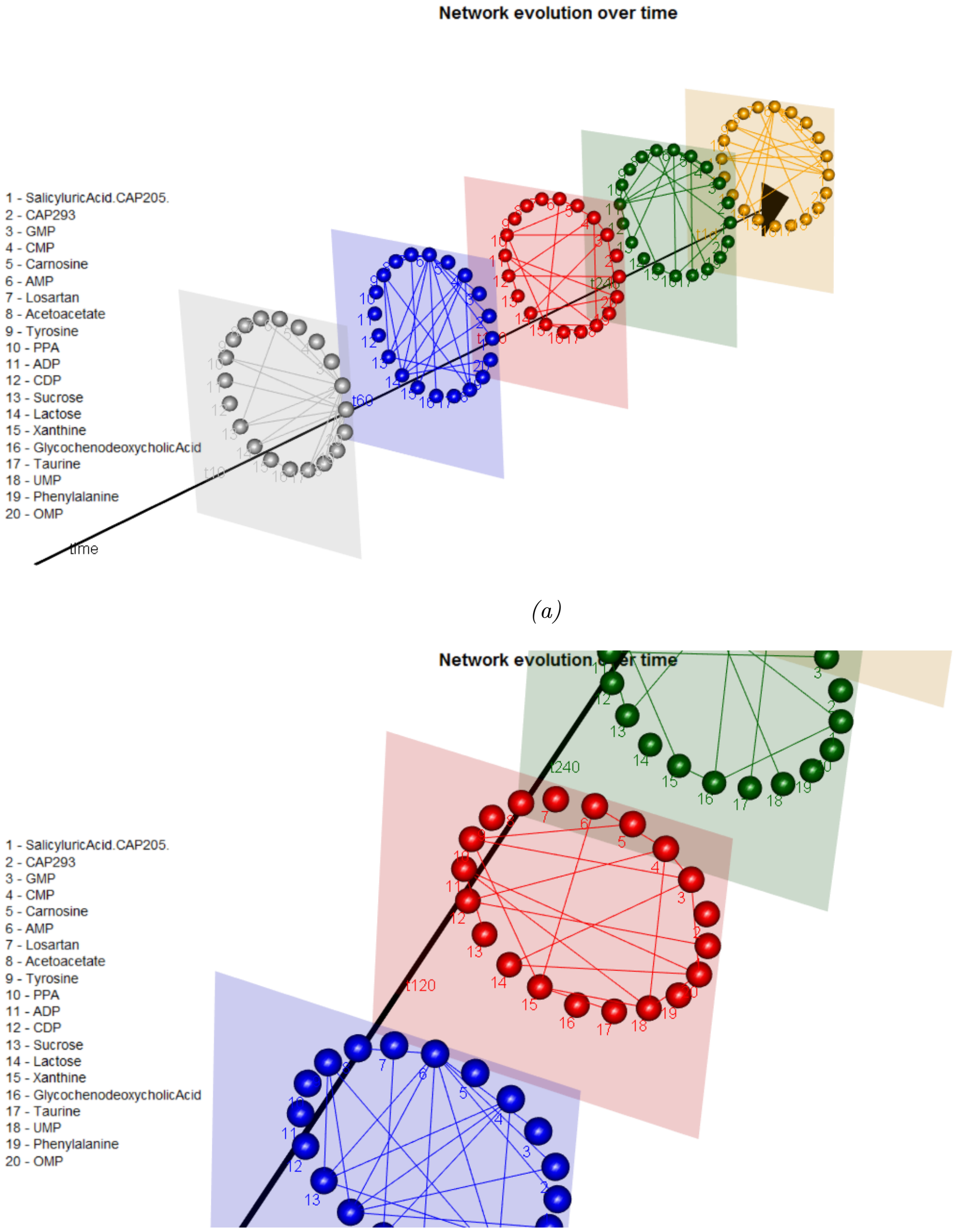
(e) threshold q90, discrete weighted

(f) threshold q90, continuous weighted

Figure 5.12: Combined network plots of networks at three different dynamic thresholds q50: (a), (b), q75: (c), (d) and q90: (e), (f) with discrete weighted edges (left side) and continuous weighted edges (right side).

5.3.5 3D Representation

Figure 5.13 depicts two snapshot images of the interactive 3D representation of the network graphs constructed with the dynamic threshold value q90. The interactive 3D visualization can be rotated around either of the three axes; also, it can be zoomed in and out. Thus, a thorough analysis of the visualized networks is ensured. For obvious reasons, this thesis will naturally only provide snapshot images in the printed version.



(a)

(b)

Figure 5.13: Snapshot of 3D representation of networks constructed with the dynamic threshold q90 for all of the measurement times. Networks at the individual points in time are depicted as planes along the time axis.

5.4 Biological Verification and Interpretation

The combined network plots depicted in figure 5.12 are investigated in detail in order to identify putative biomarker candidates based on the represented network graphs. Therefore, biological background information on the metabolites present in the networks is retrieved from the KEGG database. The relevant findings, which may be correlated to a MI and cardiovascular diseases, are summarized in table 5.2.

Table 5.2: Results of an exemplary KEGG database search. The first column represents metabolic pathways, the second shows metabolites playing a role in the respective pathway and the third column depicts disease entries linked to the respective metabolic pathway.

Metabolic pathway		**Related metabolites**		**Related diseases**	
KEGG ID	**KEGG Pathway**	**KEGG ID**	**KEGG Compound**	**KEGG ID**	**KEGG Disease**
map00230	Purine metabolism	C00144	GMP (Guanosine Monophosphate)	H00674	Anemia due to disorders of nucleotide metabolism
		C00385	Xanthine	H00824	Calcification of joints and arteries
map00240	Pyrimidine metabolism	C00112	CDP (Cytidine Diphosphate)	H00674	Anemia due to disorders of nucleotide metabolism
		C00105	UMP (Uracil Monophosphate)	H00824	Calcification of joints and arteries
map04152	AMPK signalling pathway	C00020	AMP (Adenosine Monophosphate)		
		C00008	ADP (Adenosine Diphosphate)		
map00340	Histidine metabolism	C00386	Carnosine		
map00500	Starch and sucrose metabolism	C00089	Sucrose		

The `KEGG Compound` database, which contains information about chemical compounds, is searched for some of the metabolites represented in the network. The resulting database entries are linked to other databases, for example `KEGG Pathway`, which contain information about the metabolic pathways in which the respective compound is involved. These pathway entries are filtered out considering only those which can be associated with cardiovascular diseases and which are displayed in table 5.2 together with the respective `KEGG ID`. The second column of the table contains the metabolites and their `KEGG ID` that take part in the respective metabolic pathway. In order to obtain an idea of the role of the identified pathways in cardiovascular diseases, disease entries associated with the respective pathways are provided in the table's third column. Some of the table's cells are empty due to missing database links in the `KEGG Pathway` entry or due to their unapparent association to cardiovascular diseases.

One of the metabolisms with no associated diseases listed in table 5.2 is the AMP-activated protein kinase (AMPK) signalling pathway. AMPK indicates the energy status of a cell and regulates the cellular and organism's metabolism. It is activated during metabolic stress if the generation of ATP is decelerated or ATP consumption increases and leads to inhibition of anabolic pathways. Therefore, it can be considered to be associated with cellular stress that occur after a MI [39]. Looking at the `KEGG Pathways` map of the AMPK signalling pathway, an interaction with the starch and sucrose metabolism can be detected.

Figure 5.14 gives an example of a pathway map retrieved from the `KEGG Pathways`. It depicts an overview of the histidine metabolism, metabolites that are involved herein and other pathways interacting with it. As can be seen in the pathway visualization highlighted in red, the histidine metabolism is interacting with the purine metabolism.

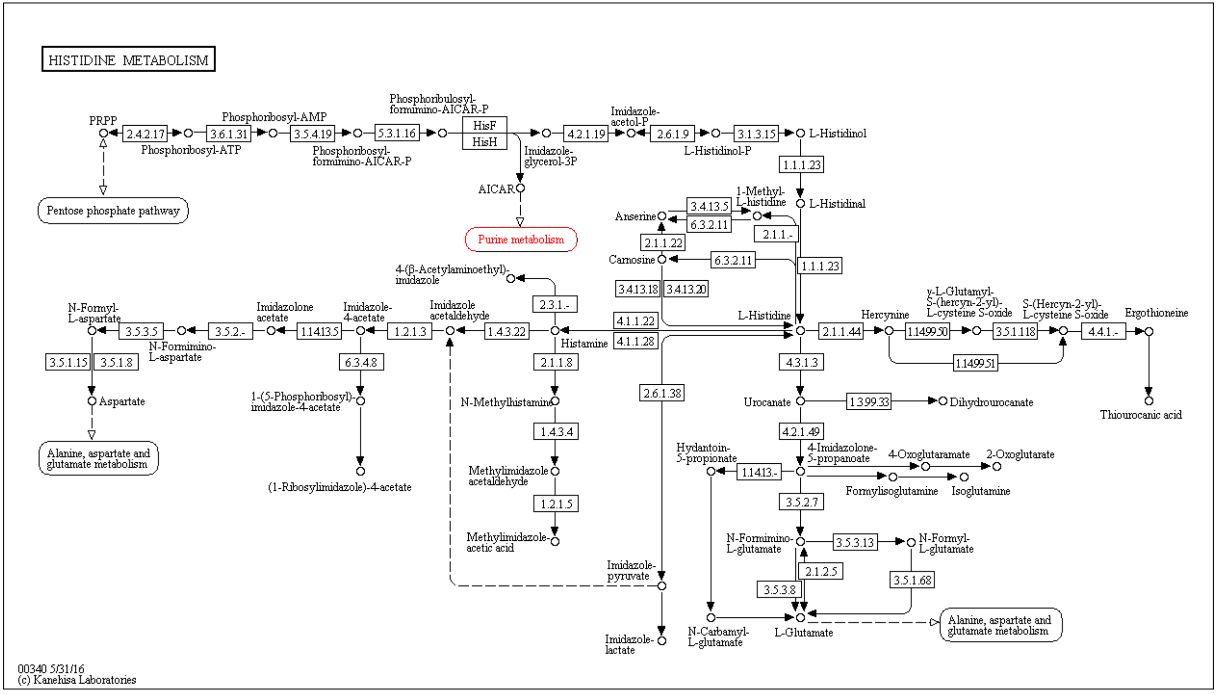


Figure 5.14: map00340 Histidine metabolism retrieved from [40]

6 Discussion and Conclusions

6.1 Data Preprocessing and Feature Selection

In chapter 3.2, several feature selection tools applicable to biological data were introduced. In fact, there is a great variety of feature selection methods used in a clinical context, including filter-based approaches combined with hypothesis testing, or wrapper-based methods such as SVM. For more information, see in section 3.2.1.

However, most of these proposed methods cannot be applied on paired sample data. The data used in this thesis stems from the same patients throughout several measurements and, consequently, has to be considered as paired data. Hence, a feature selection technique for the given data has to be chosen. Due to the fact that the pBI score calculation proposed in [31] was already used as a feature selection tool in the study (which indeed generated the given data), this method is also applied in the present thesis. This selection is further justified by the fact that there is only a small set of methods available for paired samples [3].

The pBI score computation in this thesis is applied in two different contexts. In fact, it is used after data preprocessing to extract potential biomarker candidates from the entire metabolite pool investigated and thus provides a basis for metabolite preselection. Hence, pBI scores are calculated for metabolite concentration data at different measurement points. As can be seen in figure 5.1, the range of these score values varies over the measurement time, thereby indicating different extents and directions of concentration changes within each metabolite.

In addition to the boxplot diagram which provides an overview of the calculated scores, the bar plot diagrams (examples are given in figures 5.2 and 5.3) allow for a more detailed visualization of pBI scores of the individual metabolites at different measurement points. The scores of the metabolites are ranked according to their respective absolute values for each measurement, and the 20 top-ranked metabolites are included in the plot for each case. Due to the fact that a high pBI score correlates with a high change in concentration over time [31], metabolites with vast changes in concentration are more likely to be good biomarker candidates and are therefore considered important for the further analysis. The bar plot diagrams provide a comparison of the scores at different times and provide information about which metabolites are considered as important for which measurement. In addition, the bar plot diagrams differentiate between positive score values, that indicate an increase in concentration, and negative scores, that indicate a decrease in concentration values.

In order to ensure good recognizability of the various representation and visualization methods applied with regard to the data provided in this thesis, a group of 20 metabolites yielding the highest pBI score values is included in the visualization. The number of metabolites is set empirically after comparing the figures for different numbers of metabolites with regard to recognizability and interpretability. However, any other number of metabolites could be chosen with respect to the data that is to be analyzed.

6.2 Network Visualization

The second application of the pBI score presented in this thesis is its usage during the process of constructing networks. After selecting the metabolites considered as putative biomarker candidates from the pool of investigated metabolites, a network graph is created. The metabolites that form the graph's nodes are joined by edges. In order to find significant edges that highlight the relations between two metabolites out of all possible connections, a tool for selecting such edges has to be applied. In this thesis, the graph inference is performed according to Netzer et al. [2], i.e. by calculating the pBI* score for all binary logarithmic metabolite-to-metabolite relations. A threshold is set to include only those ratios as edges in the network graph, which yield higher pBI* scores than the specified threshold value. As described in [2], each edge should represent a chemical interaction between the two respective metabolites. Thus, the task of choosing a threshold value for the edge creation is crucial with respect to the biological interpretation of the inferred networks. In this thesis, two different approaches are used to set thresholds. Networks that were constructed using these two approaches were compared in section 5.3.2.

Both of the methods used for threshold selection are based on quantile calculation. The approach for calculating the quantiles and applying them as cutoff values for the given data provides a simple and consistent way of selecting significant edges. Therefore, it is also used in [2]. It can be applied on any data since it always calculates thresholds with regard to the range of the dataset given.

The first method uses quantiles to dynamically set the threshold values. The advantage of this approach is that networks at different times exhibit the same number of edges for a given single threshold. Thus, the edges in each individual network are normalized by the maximum score for each point in time, thus allowing to independently model the inter-metabolite relations for each of the points in time. Consequently, there is a different cutoff value for each network and each time. Furthermore, the impact of outliers is reduced as compared to static threshold values.

The second method uses the dynamically calculated cutoff values of one threshold quantile, calculates the mean and then applies the resulting value as static threshold for network inference at all measurement points. Consequently, this method allows a better comparability of changes in the score values over different times. The number of edges present in the networks at different measurement points can vary to a great extent, even though all of these edges show score values higher than the same cutoff value. Thus, networks exhibiting more metabolite-to-metabolite ratios with high scores yield a higher number of edges in the network.

In general, it can be concluded that with increasing threshold values, less edges are displayed in a network. Consequently, single edges can be highlighted and more easily examined. However, the task of setting an appropriate threshold value is not a trivial one since it poses a trade-off between presenting as much detail as possible, i.e. preserving potentially important information, and presenting too much detail at the expense of recognizability. Consequently, a threshold value has to be chosen depending on the data to be analyzed and visualized – and also with regard to the underlying research question.

6.2.1 Heatmap Representation

The network visualization as heatmap (depicted in figures 5.7 and 5.8) allows a detailed overview of the interactions of a metabolite with all others in a network. Due to the colors according to the defined threshold values, connections can be easily classified as strong, moderate or weak interactions. In addition, relations between the considered and the other metabolites depicted in the heatmap can be assessed in terms of their respective concentrations due to the color scheme applied, e.g. finding out which metabolites have higher concentration values than others. Thus, a lot of information that cannot be visualized in a network graph representation can be included in this compact method.

All heatmap visualizations presented are created using static threshold values for a classification into weak, moderate and strong connections. In this context, a static threshold implies the same cutoff value for each of the points in time depicted in one heatmap, whereas the heatmap construction on a dynamic base results in a different cutoff value for each measurement time depicted in one heatmap. However, for either of the approaches, threshold values remain the same for all individual heatmaps and therefore enable a comparison between heatmaps of different metabolites.

In this thesis, only the static threshold approach is realized for the heatmap representation of networks. This is done for the sake of a fair comparison at different measurements. Using the same cutoff values for classification leads to the same color key for all times taken into consideration (they are represented as columns) of the heatmaps created. Therefore, a straightforward interpretation of individual heatmaps and their values at different times is provided.

One big advantage of the heatmap representation in comparison to its network graph counterpart is its transparency in terms of being comprehensive. Due to the color representation, a heatmap remains readable even for large amounts of data. Furthermore, its interpretation is simplified due to highlighting different classes and thereby emphasizing strong metabolite-to-metabolite connections.

However, the usability of heatmaps is limited as they only consider one metabolite and its connections to others in one single heatmap. If multiple metabolites and their connections are to be considered and interpreted, a heatmap has to be created and examined for each of the desired metabolites. Consequently, heatmap representations provide a powerful tool for a detailed analysis but are not as suitable for gaining overall insights into a network graph as network representations.

6.2.2 Comparison of Different Threshold Values

Choosing an appropriate threshold value for network inference is a challenging task because it has to meet both the requirement of presenting as much information as possible and of enabling a thorough interpretation. Different threshold values for both dynamic and static approaches are compared in this thesis (see section 5.3.2).

Depending on the threshold value chosen for network inference, it is possible to sort out less important or weaker metabolite-to-metabolite connections. When the threshold level is low, it can provide a good general overview of a network although it may not allow to differentiate between single metabolites regarding their node degree. Furthermore, a low threshold enables the identification of trends. If multiple edges accumulate around one metabolite or a group of metabolites and if there is a high density of lines associated with them (as can be seen in figure 5.9c for the metabolites salycyluric acid or CAP293), these metabolites yield a high number of interactions and might thus play an important role. In contrast, if the threshold level is set high, stronger connections can be viewed in detail and a network gains transparency. As a consequence, the interpretation of single metabolites and their connections is simplified but it is more likely to exclude edges which might provide essential information.

As can be seen in figure 5.9, the number of edges is reduced with an increasing value for dynamically calculated thresholds. As the threshold calculation is done by computing the respective quantile value, the number of edges present in a network directly depends on the chosen quantile for dynamic threshold calculations.

With this in mind, it becomes apparent that choosing a threshold value is always a compromise between the amount of data to be depicted and the interpretability of the visualization. To answer the question of which aspect is more important, both the research issue and specific questions of interest have to be taken into consideration. Therefore, the choice of an appropriate threshold has to be made with regard to the data to be analyzed as well as depending on the respective research goals.

6.2.3 Comparison of Networks at Different Points in Time

One aspect of this thesis is to investigate networks inferred at different consecutive measurement times in order to visualize the dynamics of longitudinal data. Figures 5.10 and 5.11 provide an illustration of networks inferred with the same threshold value for data of different measurements. It can be seen that the edge constellations change for each of the times: while some edges are present at several times, for example the edge connection of the metabolites AMP and sucrose, other edges only appear once over several points in time, for example the edge connecting the metabolites salycyluric acid and CAP293 for a dynamic threshold of q75. This information allows to get a first idea about the role of individual metabolites in the human body with regard to their occurrence and the number of metabolic pathways in which they are involved. However, if conclusions are drawn based on these findings, they have to be confirmed in a further verification step.

In addition, a network visualization over time allows to draw conclusions about the importance of a metabolite judged by its degree. This is a measure for the number of metabolites it is connected to. Several metabolites show a high degree in some of the networks over time, which may lead to the conclusion that they play an important role in biochemical processes triggered by the MI.

In conclusion, it can be said that comparing individual networks over time allows insights into the underlying metabolic processes during the monitoring of measurements. However, in order to confirm these assumptions it is mandatory to consider the respective data and perform a thorough biological verification and validation of the findings.

6.2.4 Comparison of Combined Network Plots

One main issue of this thesis is to implement a method for visualizing the dynamic changes in metabolic activity after a MI over a defined time frame and based on network graphs. The combined network graph representation depicted in figure 5.12 is one approach for realizing this aim. Networks of multiple measurement points are combined and all edges are weighted and displayed in the graph.

In general, two different approaches are applied in order to determine the edge weights of the combined networks. The first, the discrete calculation of the weights, is a simple approach where edge weights are determined by summarizing all adjacency matrices. It is straightforward to realize and calculation efforts are limited. Here, the weights take discrete values according to their occurrence frequency in the networks to combine. The second approach takes into account the respective score values of all edges of the network graphs to combine. This results in continuous weight values which are related to the strengths of the connections over time.

The calculated weight values are rescaled to meet a range of zero to one for both approaches. This allows for re-usability of the methods and provides a graphical representation which is easy to recognize. It also ensures that both approaches can be applied to a broad range of different data sets by covering both small and high numbers of networks to combine.

In addition, the rescaled edge weights are weighted again for visualization in terms of their line width, by using a fourth degree polynomial function to emphasize strong interactions. In other words, the values that are in the range from zero to one are expanded in order to attenuate low scores and highlight only those edges which yield higher scores. Thus, for illustrating the networks in figure 5.12, a non-linear weighting of the edges is applied. However, other weighting functions such as linear or higher-order polynomial weighting can also be applied – always depending on the desired graph representation.

Comparing the different edge weighting approaches, it can be seen that especially for smaller threshold values, more edges are highlighted with thick lines for the discrete calculation of edges than for those of a continuous type. In general, if the threshold value for the network inference is smaller, more potential edges are available to be summed up for different points in time. As a result, the probability that one edge occurs more than once increases in comparison to networks with a higher threshold level. Therefore, it is more likely to find multiple highlighted edges in combined network plots inferred with lower threshold values.

The reason why there are less strongly highlighted edges in combined graphs with continuously calculated edge weights with the same threshold level is that the scores of all edges are taken into consideration. The described phenomenon can be observed comparing subplots (c) and (d) of figure 5.12. Subplot (c) shows more strongly highlighted edges than subplot (d). In contrast, some of the edges highlighted in (c) do not yield high score values – for example the edge connecting the metabolites glycochenodeoxycholic acid and AMP, as can be confirmed when looking at (d) where this edge is not highlighted that strongly. However, other edges, e.g. the connection between UMP and CMP, are emphasized as being more important when considering the score values. The difference between the two edge weighting approaches becomes clearly visible when comparing subplots (e) and (f). Subfigure (e) only yields one strongly highlighted edge, whereas subfigure (f) classifies three edges as important by strong highlighting.

In conclusion, it can be said that for high threshold values, the discrete weighting approach could lead to misinterpretations due to neglecting the score values. In contrast, the continuous weighting approach represents score value information and therefore allows a sophisticated evaluation of the network graph. However, when choosing the more suitable of the proposed weighting approaches, both the data as well as the purpose of its visualization have to be taken into account.

6.3 Biological Verification and Interpretation

As described in section 3.2 and discussed in the previous sections, the verification of the identified biomarker candidates is an essential task in the biomarker discovery process. In order to gain information about putative biomarkers, databases such as KEGG can be used. In the following, the findings of Lewis et al. [1] are compared to those provided by the proposed visualization methods implemented in this thesis. As these methods are realized using the same data as [1], this comparison provides information about the performance of the discussed network representations.

The metabolites alanine, hypoxanthine, isoleucine/leucine, malonic acid and threonine are identified to have significant changes in concentration early after PMI, i.e. for instance at 10 minutes, as reported in [1]. Comparing these findings to the top ranked metabolites depicted in barplot 5.2, it can be seen that these metabolites are mostly present in the figure as well. However, for the network visualization in the present thesis, pBI scores of the individual metabolites at all measurement times are considered. Therefore, only 20 metabolites yielding the highest scores with regard to all times are included in the network representation. Hence, the metabolites depicted in figure 5.10a, which is a graph illustration at time t10 (with threshold q75), are not the same as the ones with the highest scores for t10 that are shown in 5.2. Due to the preselection of metabolites with regard to all measurement points, not all metabolites ranked as important at a specific time point are included. Nevertheless, a comparison of the inferred network graphs and, thus, a dynamic visualization over the measurement time frame are provided.

Netzer et al. [2] propose a method for metabolic network construction where each edge represents the interaction between two metabolites. They confirm their approach for a network-based visualization and biochemical interpretation of metabolites associated with physical exercise using `KEGG Pathways` and literature reviews. In this thesis, the method proposed by [2] is used and adapted in order to identify biomarkers associated with a MI and cardiovascular diseases. The combined network graphs illustrated in 5.12 show weighted metabolite-to-metabolite interactions based on pBI* score values and thus on the connection's strength. As can be seen in the network graphs inferred with high thresholds (see figures 5.12e and 5.12f), the metabolites sucrose and AMP show a strong connection, i.e. strongly weighted edges. This means that they show connections at several measurements and can therefore be considered as strongly interacting. A KEGG database search, as described in 5.4, shows that AMP is a component of the AMPK signalling pathway and sucrose is part of the starch and sucrose metabolism. Further investigation performed with KEGG indicates an interaction of the two pathways and thereby confirms the findings based on 5.12.

In addition, the heatmap representation of AMP (figure 5.7) shows that the metabolite strongly interacts with others at all times. This could indicate the AMPK signalling pathway to be highly involved in different metabolic processes. On the opposite, strong interactions of the metabolite CAP293 mainly occur early at t10 and late at t1d. There could be various reasons for this but in order to gain a deeper understanding, an analysis with the aid of databases and basic biochemical research is necessary.

It should be stated that the associations given in this thesis are only exemplary. Additional findings are shown in table 5.2. However, in order to perform an overall validation of the proposed methods and findings, further investigation is needed.

6.4 Conclusion and Future Outlook

Summarizing, it can be said that this thesis provides a variety of different dynamic network-based approaches for the identification of new metabolic biomarkers in cardiovascular diseases. Several different visualization methods are proposed and discussed for this purpose. One main issue to consider with regard to all of the presented methods is the data preselection. The task of identifying putative biomarker candidates prior to visualization is essential in order to ensure a sophisticated network representation. All parameters used for preselection as well as for graph construction have to be chosen carefully with regard to the respective data.

It was shown that the approach for the graph construction proposed in [2] provides a useful tool for network inference and can indeed be used as a basis for different graph representations. However, limitations in graph visualization are due to the number of nodes to display. A compromise between the number of nodes and the chosen threshold values needs to be found in order to achieve meaningful visualizations.

Different graph representations are realized so as to highlight various aspects of the provided data. The dynamic changes in metabolite concentrations can be visualized using the combined network graphs, whereby details in metabolite-to-metabolite interactions can be examined using heatmap representation. Consequently, the graph representation of one's choice depends on the aspect of interest in the given data. In addition, the implemented graph representations can only be used as significant biomarker prediction tools when combined with biochemical validation of the findings.

The present thesis proposes a selection of tools applicable for the visualization of dynamic networks of putative biomarker candidates in cardiovascular diseases. It also provides an insight into representation methods of dynamic graphs which can be applied in the field of biomarker research. Besides the methods discussed in this work, a wide range of other visualization tools such as arc diagrams can be adapted to provide assistance in the discovery of biomarkers.

A

Appendix

A.1 Barplots of Paired Biomarker Identifier Scores

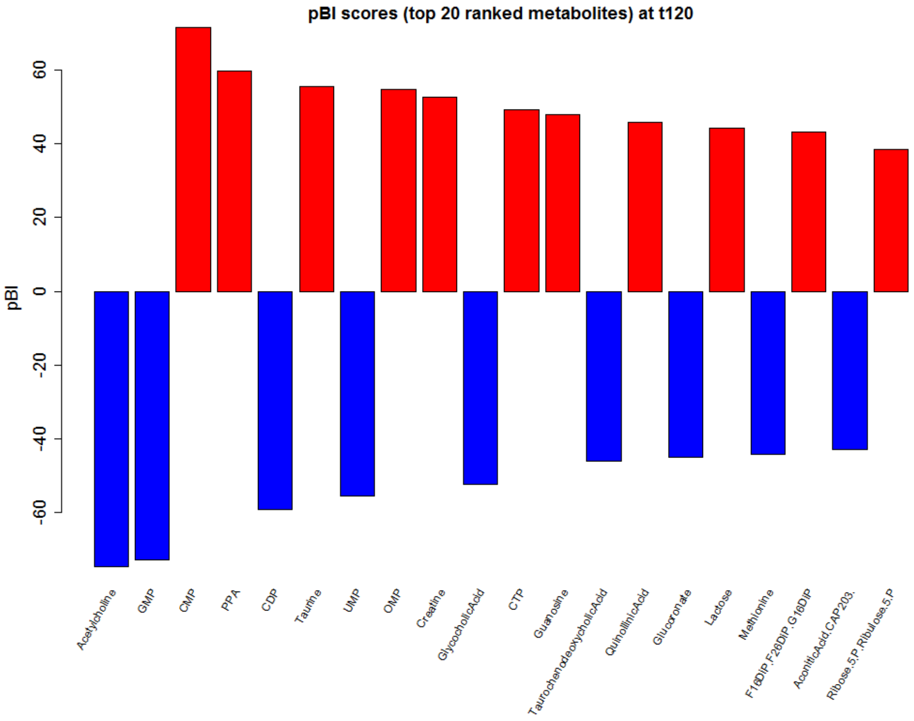


Figure A.1: Barplot of the top 20 metabolites, ranked according to their respective pBI score values at measurement time point t120. The red bars indicate a positive pBI score value and the blue bars indicate a negative pBI score value.

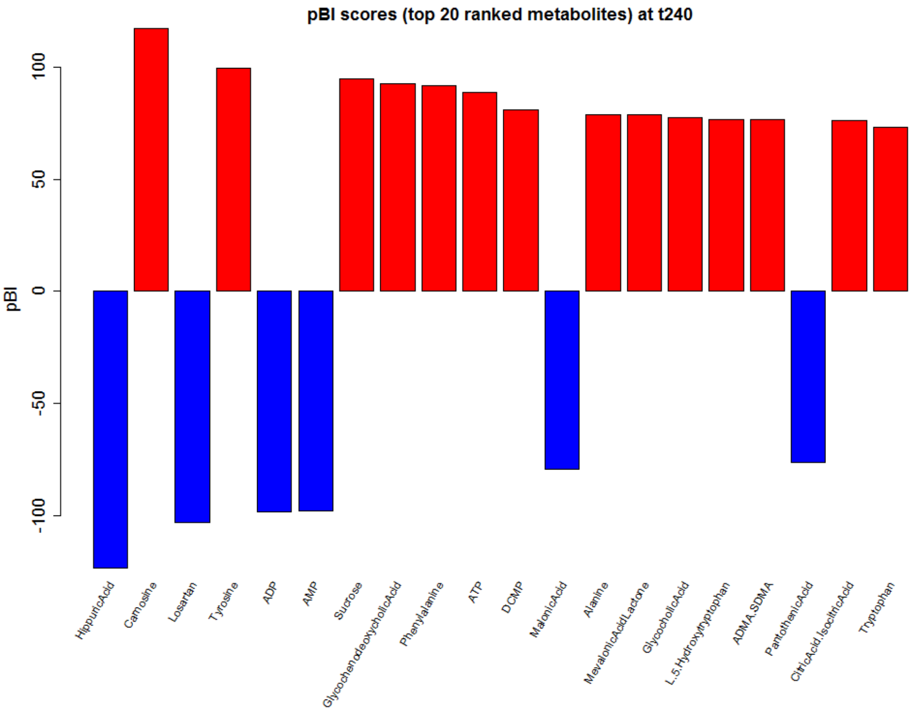


Figure A.2: Barplot of the top 20 metabolites, ranked according to their respective pBI score values at measurement time point t240. The red bars indicate a positive pBI score value and the blue bars indicate a negative pBI score value.

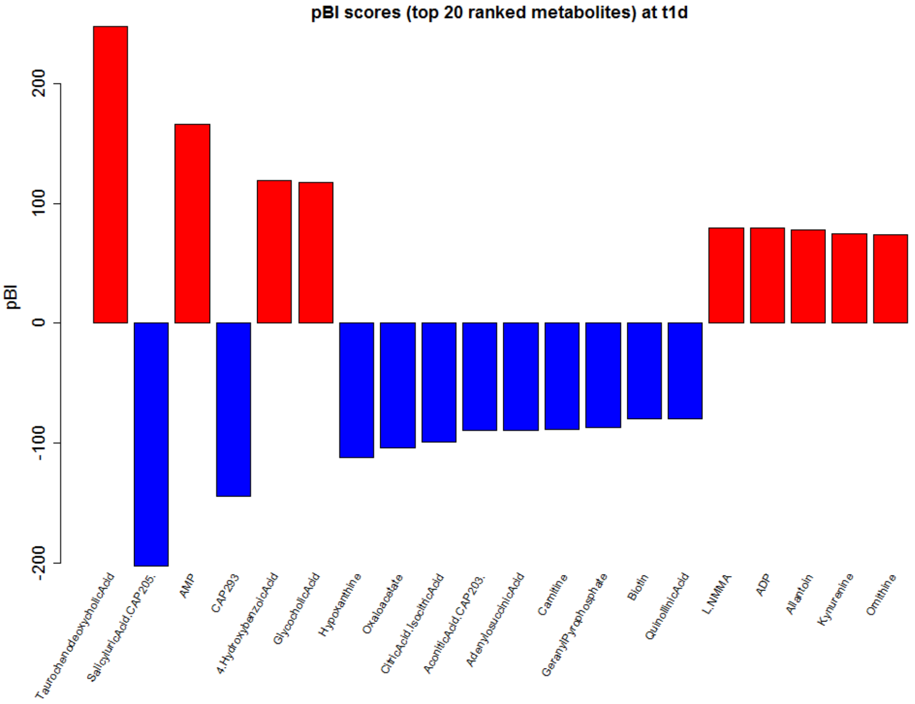
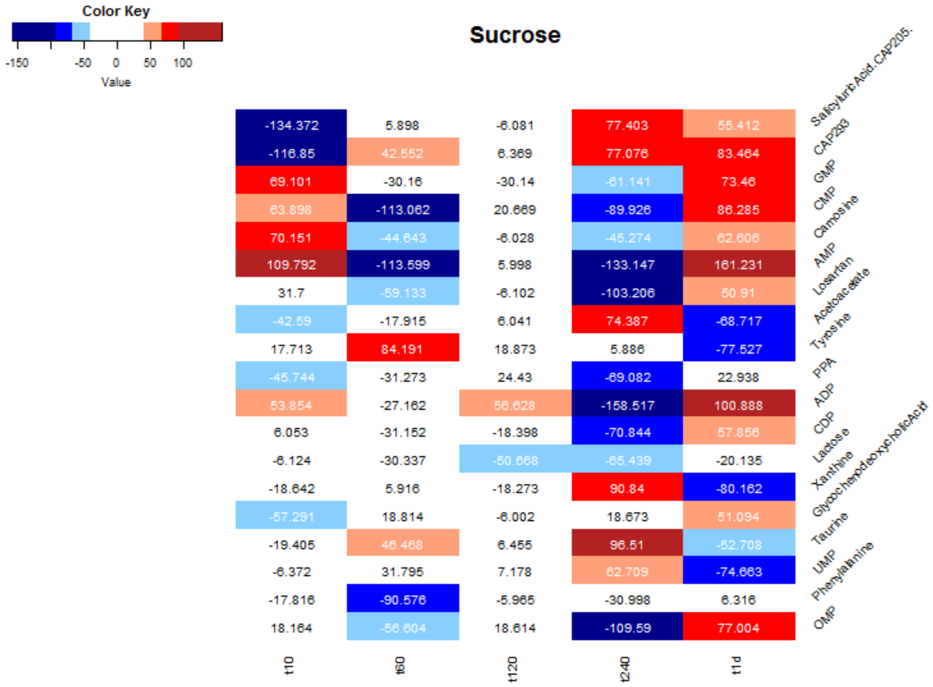


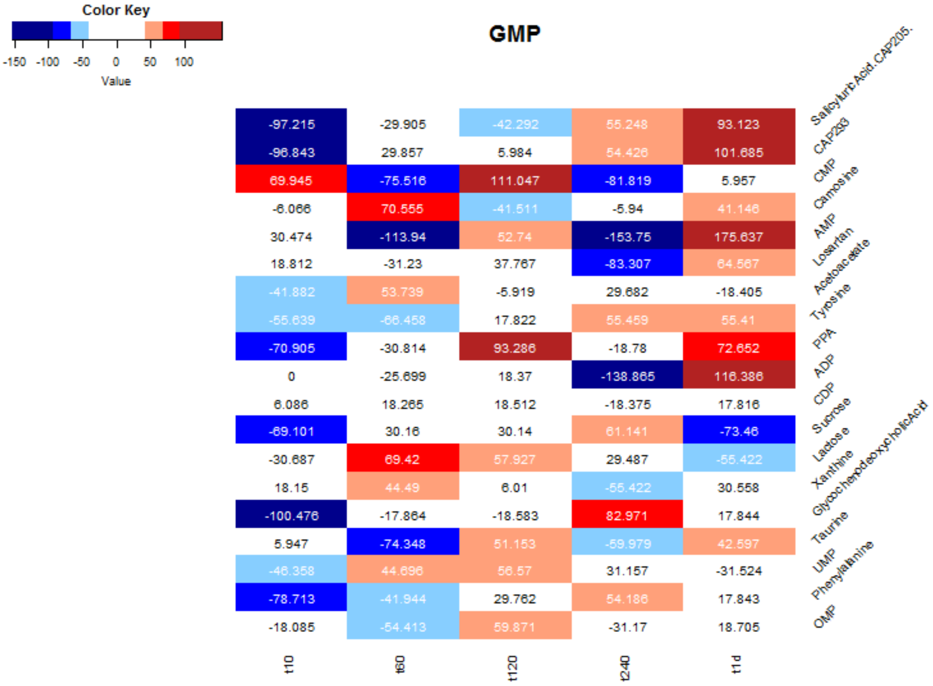
Figure A.3: Barplot of the top 20 metabolites, ranked according to their respective pBI score values at measurement time point t1d. The red bars indicate a positive pBI score value and the blue bars indicate a negative pBI score value.

A.2 Heatmaps



t10	t60	t120	t240	t1d	
-134.372	5.898	-6.081	77.403	55.412	SalicyluricAcid.CAP205.
-116.85	42.552	6.369	77.076	83.464	CAP233
69.101	-30.16	-30.14	-61.141	73.46	GMP
63.898	-113.062	20.669	-89.926	86.285	CMP
70.151	-44.643	-6.028	-45.274	62.606	Carnosine
109.792	-113.599	5.998	-133.147	161.231	AMP
31.7	-59.133	-6.102	-103.206	50.91	Losartan
-42.59	-17.915	6.041	74.387	-68.717	Acetoacetate
17.713	84.191	18.873	5.886	-77.527	Tyrosine
-45.744	-31.273	24.43	-69.082	22.938	PPA
53.854	-27.162	56.628	-158.517	100.888	ADP
6.053	-31.152	-18.398	-70.844	57.856	CDP
-6.124	-30.337	-50.668	-65.439	-20.135	Lactose
-18.642	5.916	-18.273	90.84	-80.162	Xanthine
-57.291	18.814	-6.002	18.673	51.094	GlycochenodeoxycholicAcid
-19.405	46.468	6.455	96.51	-52.708	Taurine
-6.372	31.795	7.178	62.709	-74.663	UMP
-17.816	-90.576	-5.965	-30.998	6.316	Phenylalanine
18.164	-56.604	18.614	-109.59	77.004	OMP

Figure A.4: Heatmap representing the metabolite connections originating from metabolite Sucrose. Numbers in the map field give the pBI scores of the binary logarithmic ratios between Sucrose and the respective metabolites. Colors indicate the strength of connection associated with the absolute score, whereas a high value indicates a strong connection and is represented by a dark color.*

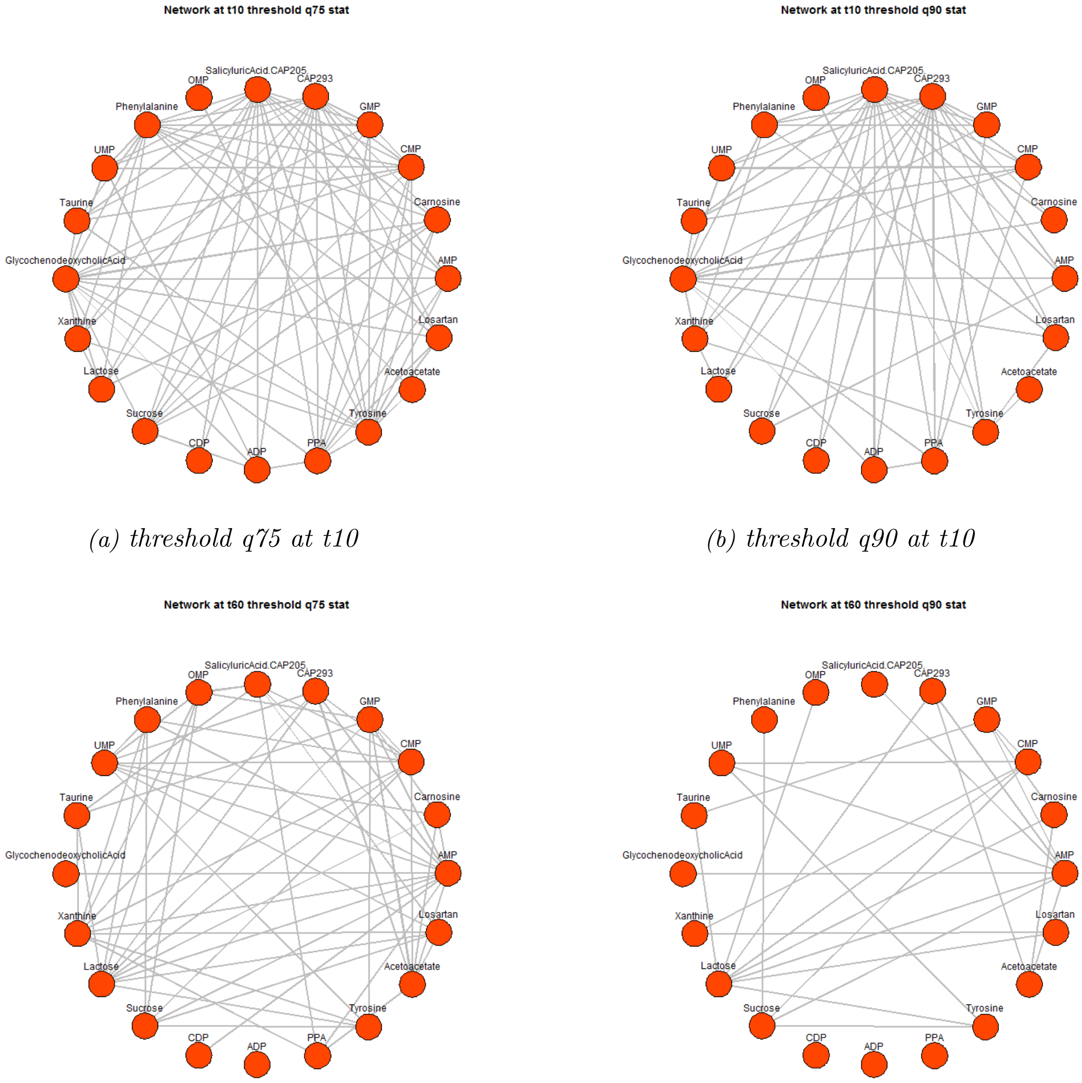


t10	t60	t120	t240	t1d	
-97.215	-29.905	-42.292	55.248	93.123	SalicyluricAcid.CAP205.
-96.843	29.857	5.984	54.426	101.685	CAP233
69.945	-75.516	111.047	-81.819	5.957	CMP
-6.066	70.555	-41.511	-5.94	41.146	Carnosine
30.474	-113.94	52.74	-153.75	175.637	AMP
18.812	-31.23	37.767	-83.307	64.567	Losartan
-41.882	53.739	-5.919	29.682	-18.405	Acetoacetate
-55.639	-66.458	17.822	55.459	55.41	Tyrosine
-70.905	-30.814	93.286	-18.78	72.652	PPA
0	-25.699	18.37	-138.865	116.386	ADP
6.086	18.265	18.512	-18.375	17.816	CDP
-69.101	30.16	30.14	61.141	-73.46	Sucrose
-30.687	69.42	57.927	29.487	-55.422	Lactose
18.15	44.49	6.01	-55.422	30.558	Xanthine
-100.476	-17.864	-18.583	82.971	17.844	GlycochenodeoxycholicAcid
5.947	-74.348	51.153	-59.979	42.597	Taurine
-46.358	44.696	56.57	31.157	-31.524	UMP
-78.713	-41.944	29.762	54.186	17.843	Phenylalanine
-18.085	-54.413	59.871	-31.17	18.705	OMP

Figure A.5: Heatmap representing the metabolite connections originating from metabolite GMP. Numbers in the map field give the pBI scores of the binary logarithmic ratios between GMP and the respective metabolites. Colors indicate the strength of connection associated with the absolute score, whereas a high value indicates a strong connection and is represented by a dark color.*

A.3 Network Graphs

A.3.1 Comparison of Networks at Different Measurement Times with Static Thresholds q75 and q90


Network at t10 threshold q75 stat

Network at t10 threshold q90 stat

Network at t60 threshold q75 stat

Network at t60 threshold q90 stat

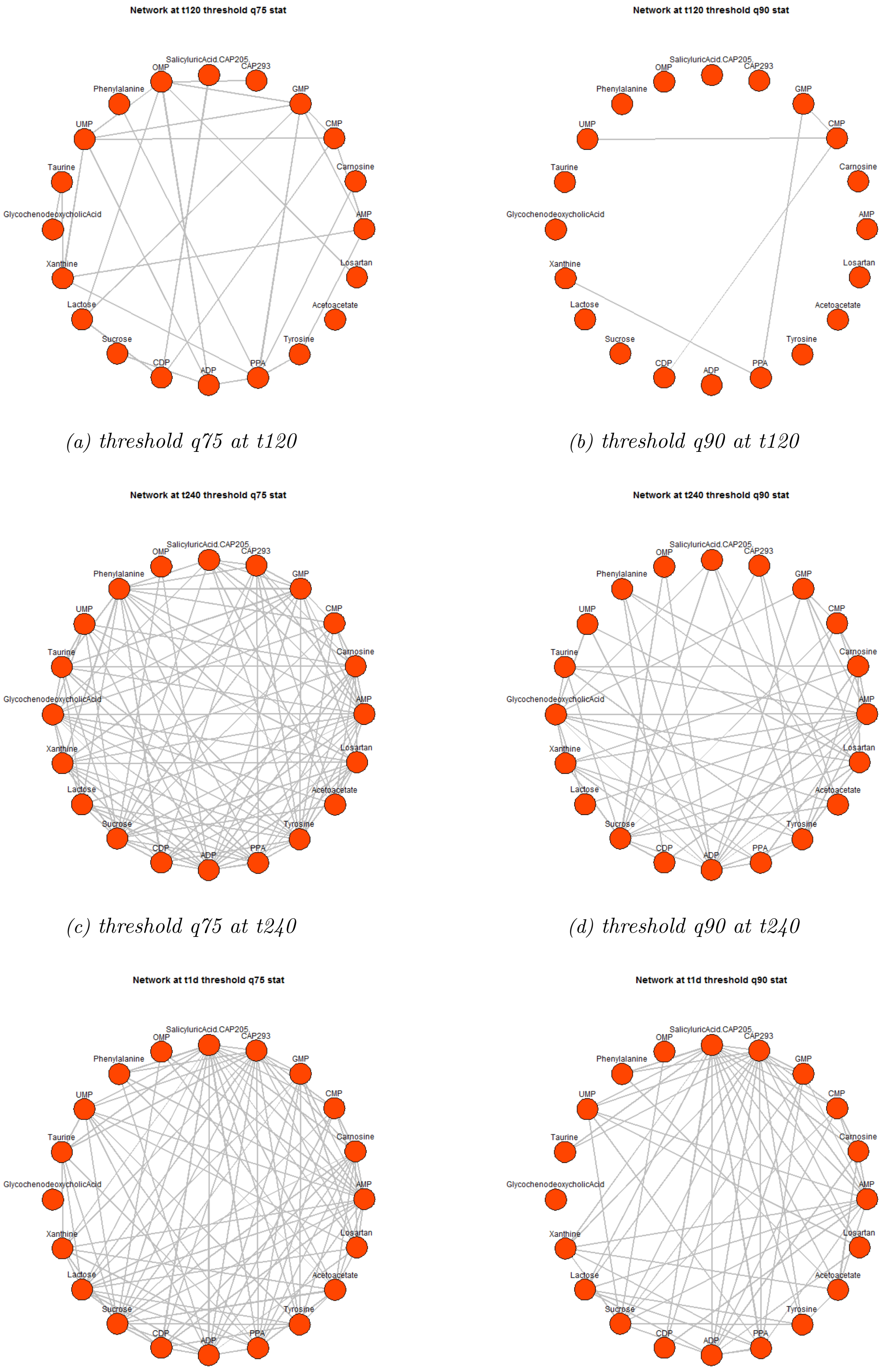
(a) threshold q75 at t10

(b) threshold q90 at t10

(c) threshold q75 at t60

(d) threshold q90 at t60

Figure A.6: Network plots in circle layout at statically calculated thresholds q75 and q90 at two different measurement time points t10: (a), (b) and t60: (c), (d).



Network at t120 threshold q75 stat

Network at t120 threshold q90 stat

(a) threshold q75 at t120

(b) threshold q90 at t120

Network at t240 threshold q75 stat

Network at t240 threshold q90 stat

(c) threshold q75 at t240

(d) threshold q90 at t240

Network at t1d threshold q75 stat

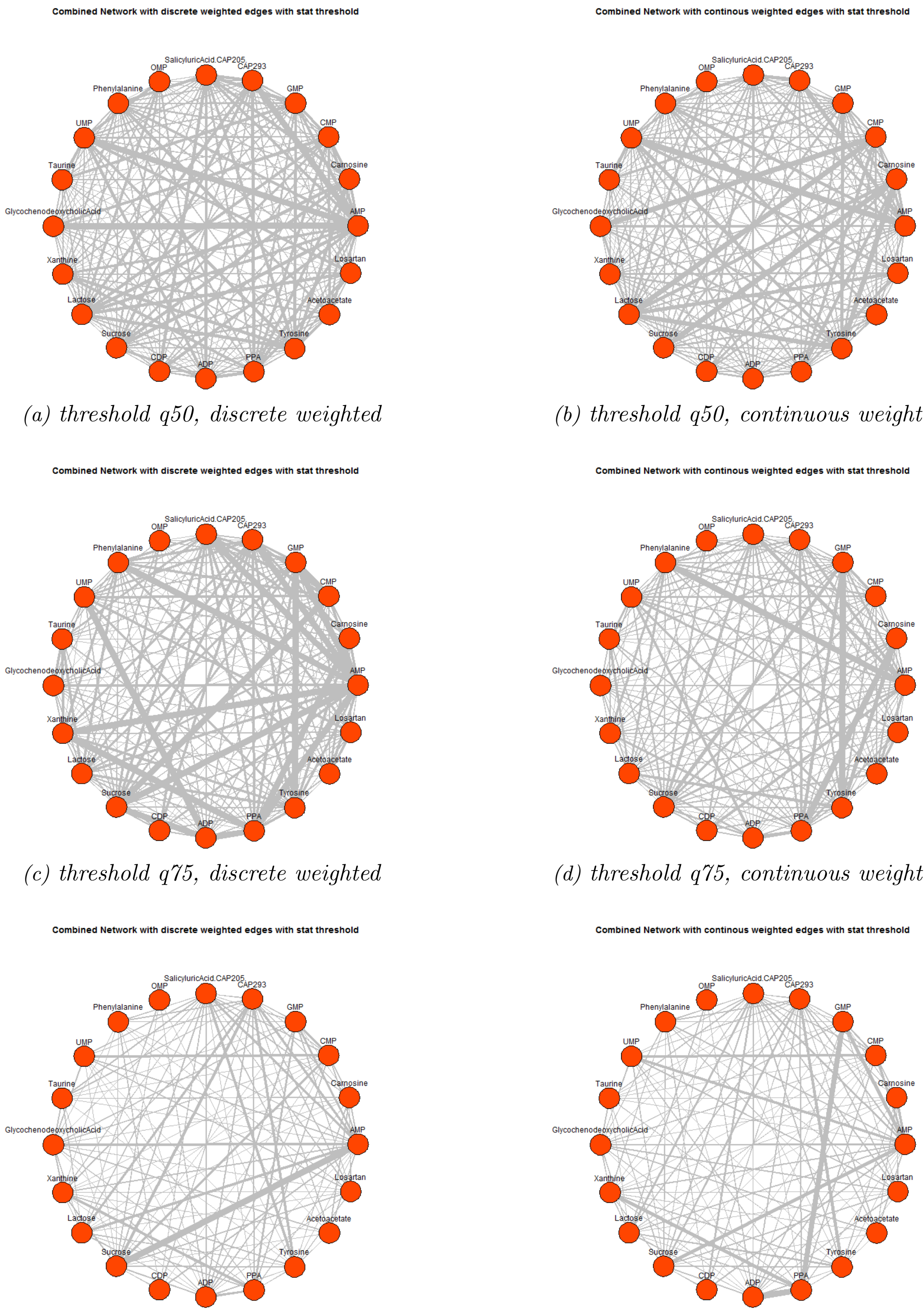
Network at t1d threshold q90 stat

(e) threshold q75 at t1d

(f) threshold q90 at t1d

Figure A.7: Network plots in circle layout at statically calculated thresholds q75 and q90 at three different measurement time points t120: (a), (b), t240: (c), (d) and t1d: (e), (f).

A.3.2 Comparison of Combined Network Plots for Static Thresholds


Combined Network with discrete weighted edges with stat threshold
Combined Network with continous weighted edges with stat threshold
SalicyluricAcid.CAP205
OMP
CAP293
Phenylalanine
GMP
UMP
CMP
Taurine
Carnosine
GlycochenodeoxycholicAcid
AMP
Xanthine
Losartan
Lactose
Acetoacetate
Sucrose
Tyrosine
CDP
PPA
ADP

(a) threshold q50, discrete weighted

(b) threshold q50, continuous weighted

(c) threshold q75, discrete weighted

(d) threshold q75, continuous weighted

(e) threshold q90, discrete weighted

(f) threshold q90, continuous weighted

Figure A.8: Combined network plots of networks at three different statical thresholds q50: (a), (b), q75: (c), (d) and q90: (e), (f) with discrete weighted edges (left) and continuous weighted edges (right).

A.4 R Code

Listing A.1: R-Code: Functions implemented to create and visualize different network representations.

```
#-----------------------------------------------------------------------------
preprocessingAllData <- function(datalist, excludeMetabolThresh=60,
                                 numMetToInclude=15, medianToSub=TRUE,
                                 pbiQuantToInclude=0.75)

  for(j in 1:length(datalist)){
    cleaned <- datalist[[j]]

    tperc <- nrow(datalist[[j]])/100*excludeMetabolThresh
    temp <- numeric(0)
    for(i in 4:ncol(datalist[[j]])){
      sumNA <- length(which(is.na(datalist[[j]][,i])))
      if(sumNA >= tperc){
        temp <- c(temp, i)
      }
    }
    if(length(temp) != 0){cleaned <- datalist[[j]][,-temp]}

    # replace na’s with median of each column
    reflabel <- cleaned$Classlabel[1]
    partOne <- cleaned[cleaned$Classlabel==reflabel,]
    partTwo <- cleaned[cleaned$Classlabel==(reflabel+1),]

    for(i in 4:ncol(cleaned)){
      if(medianToSub==TRUE){
        colsubsPartOne <- median(partOne[!is.na(partOne[,i]),i])
        colsubsPartTwo <- median(partTwo[!is.na(partTwo[,i]),i])
      }else{
        colsubsPartOne <- mean(partOne[!is.na(partOne[,i]),i])
        colsubsPartTwo <- mean(partTwo[!is.na(partTwo[,i]),i])
      }
      partOne[is.na(partOne[,i]),i] <- colsubsPartOne
      partTwo[is.na(partTwo[,i]),i] <- colsubsPartTwo

      cleaned <- rbind(partOne, partTwo)
    }
    datalist[[j]] <- cleaned
  }

  # reduce metabolite label length
  for(j in 1:length(datalist)){
    tmp <- strsplit(names(datalist[[j]][,4:ncol(datalist[[j]])]), ’_’)
    for (i in 1:length(tmp)){
      names(datalist[[j]])[3+i]<-tmp[[i]][3]
    }
  }

  # reduce matrices at all time points to same length, containing same
      metabolites
  data <- datalist
  colnamesAll <- NULL
  for(i in 1:length(data)){
    colnamesAll <- c(colnamesAll, colnames(data[[i]]))
  }
  uniqueNames <- unique(colnamesAll)

  boolMatrix <- matrix(FALSE, length(uniqueNames), length(data))
  boolVector <- NULL
  dataSameLength <- vector(’list’, length(data))

  for (i in 1:length(uniqueNames)){
    for (j in 1:length(data)){
      boolMatrix[i,j] <- uniqueNames[i] %in% colnames(data[[j]])
    }
    boolVector[i] <- all(boolMatrix[i,])
  }

  for(j in 1:length(data)){
    dataSameLength[[j]] <- data[[j]][uniqueNames[boolVector==TRUE]]
  }

  tempMet <- NULL
  data <- dataSameLength

  if(numMetToInclude > ncol(data[[1]])-3){numMetToInclude = ncol(data[[1]])-3}

  pbiscores <- vector(’list’, length(data))
  for(i in 1:length(data)){
    pbiscores[[i]] <- pBI(t(data[[i]][4:ncol(data[[i]])]), classlabels=data[[i]]$Classlabel,
                          referenceclasslabel=data[[i]]$Classlabel[1],
                          ids=data[[i]]$ID, useMedian = TRUE, lambda = 100,
                          plotScores = FALSE, numTopRankedToPlot = 10,
                          bars.cols = c("red","blue"))
    pbiTemp <- pbiscores[[i]]
    pbiQuantThresh <- quantile(abs(pbiTemp), pbiQuantToInclude)
    scoresTopQuant <- pbiTemp[abs(pbiTemp)>pbiQuantThresh]
    scoresTopNormed <- scoresTopQuant/max(abs(scoresTopQuant))
    # save metabolites to include in a new variable
    tempMet <- c(tempMet, scoresTopNormed)
  }
  scoresRanked <- tempMet[match(sort(abs(tempMet), decreasing = TRUE), abs(tempMet))]
  uniqueScoresRanked <- unique(names(scoresRanked))

  metToInclude <- uniqueScoresRanked[1:numMetToInclude]

  for(i in 1:length(data)){
    data[[i]]<- cbind(data[[i]][c(1,2,3)], data[[i]][,na.omit(metToInclude)])
  }
  return(list(data=data, pbiscores=pbiscores))
}


#-----------------------------------------------------------------------------
overviewBoxplotDiffThresholds <- function(pbiScores, cutoffs, useAbs=TRUE,
                                          filePath, time, plotOverview=TRUE,
                                          mainText=’Boxplots of absolute values of pBI data’){

  colVec <- c("darkgreen", "orange", "red")

  if(is.list(pbiScores)){
    pbi <- matrix(NA, nrow=length(pbiScores[[1]]), ncol=length(pbiScores))
    colnames(pbi) <- time
    rownames(pbi) <- names(pbiScores[[1]])
    for(i in 1:length(pbiScores)){
      pbi[,i] <- pbiScores[[i]]
    }
  } else if(is.matrix(pbiScores)){
    pbi <- pbiScores
  } else {stop("Input data has to be a list or a matrix!")}

  # calculate different threshold values for metabolite preselection
  cutVal <- matrix(’numeric’, nrow=length(cutoffs), ncol=ncol(pbi))
  row.names(cutVal) <- cutoffs
  colnames(cutVal) <- time

  cutValMean <- vector(’numeric’, length=length(cutoffs))

  legendText <- vector(’character’, length=length(cutoffs))

  for(i in 1:length(cutoffs)){
    tempCutVal <- as.numeric(strsplit(cutoffs[i], split=’q’)[[1]][2])
    tempCut <- vector(’numeric’, ncol(pbi))
    for(j in 1:ncol(pbi)){
      tempCut[j] <- quantile(abs(pbi[,j]), tempCutVal/100)
    }
    cutVal[i,] <- tempCut
    cutValMean[i] <- mean(tempCut)

    legendText[i] <- as.expression(bquote(tau[.(i)] ~ "=" ~ .(round(cutValMean[i],2)) ~ "( mean of" ~ .(cutoffs[i]) ~ ")"))
  }
  names(cutValMean) <- cutoffs

  if(plotOverview==TRUE){
    if(useAbs){
      png(filename=paste(filePath, 'abs_pBI_thresholds.png', sep=""), width=800, height=700)
      boxplot(abs(pbi), main=mainText, ylab='abs(pBI)', xlab='measurement time points')
      for(i in 1:length(cutValMean)){
        abline(h=cutValMean[i], col=colVec[i], lwd=2)
      }
      legend('top', legend=legendText,
             lty=1, col=c('darkgreen','orange', 'red'))
      dev.off()
    } else {
      png(filename=paste(filePath, 'pBI_scores.png'), width=800, height=700)
      boxplot(pbi, main='Boxplots of pBI data', ylab='pBI score', xlab='measurement time points')
      dev.off()
    }
  }
  return(cutValMean)
}

#-------------------------------------------------------------------------------
calcGraph <- function(datalist, time, threshold='q50'){

  # initialize empty matrix for storing the scores
  temp<-0
  for(i in (ncol(datalist[[1]][4:ncol(datalist[[1]])])-1):1) temp <- temp+i
  Scores <- matrix(nrow=temp, ncol=length(datalist))

  Graph <- vector("list", length(datalist))

  for (i in 1:length(datalist)){
    data <- datalist[[i]]
    pbi_data <- t(data[,4:ncol(data)])
    refl <- data$Classlabel[1]

    temp <- pBIGraph(pbi_data, classlabels=data$Classlabel,
                     referenceclasslabel=refl, ids=data$ID, useMedian = TRUE,
                     lambda = 100, threshold = threshold, plotGraph = FALSE,
                     edge.file = NULL)

    Scores[,i] <- temp$scores
    Graph[[i]] <- temp$graph
    colnames(Scores) <- time
    rownames(Scores) <- names(temp$scores)
  }
    return(list(g=Graph, ratios=Scores))
}

#-------------------------------------------------------------------------------
.transferRatioVec <- function(pbiall, Scores){

  # numer of metabolites
  n <- ncol(pbiall[[1]][4:ncol(pbiall[[1]])])

  MetaboliteRatios <- vector("list", n)

  # create vector for indicating the position of ratios of next metabolite
  pos <- numeric(n)
  for (i in 1:(n-1)){
    pos[i+1] <- pos[i]+n-i
  }

  for (j in 1:ncol(Scores)){
    RatioScores <- Scores[,j]
    #length of RatioScores vektor
    k <- length(RatioScores)
    #claculate vektor with ratios of one metabolite and store it into the list MetaboliteRatios

    for (i in 1:length(pos)){
      if(i==1){
        # get all ratios of first metabolite
        tmp <- RatioScores[(pos[i]+1):pos[i+1]]
        # store names of these ratios
        tmpname <- names(RatioScores)[(pos[i]+1):pos[i+1]]
        # split names of ratios and use only second metabolite as new name (first metabolite is the same for all)
        names(tmp) <- unlist(strsplit(tmpname,"/"))[2*(1:(length(MetaboliteRatios)-1))]

      } else if(i==length(pos)){
        tmp <- -RatioScores[pos[2:i]-(n-i)]
        tmpname <- rownames(as.matrix(RatioScores))[pos[2:i]-(n-i)]
        names(tmp) <- unlist(strsplit(tmpname,"/"))[2*(1:(length(MetaboliteRatios)-1))-1]
      } else{
        tmp <- c(-RatioScores[pos[2:i]-(n-i)], RatioScores[(pos[i]+1):pos[i+1]])
        #names(tmp) <- rownames(as.matrix(RatioScores))[c(pos[2:i]-(n-i),(pos[i]+1):pos[i+1])]
        tmpname1 <- rownames(as.matrix(RatioScores))[pos[2:i]-(n-i)]
        tmpname1 <- unlist(strsplit(tmpname1,"/"))[2*(1:(i-1))-1]
        tmpname2 <- rownames(as.matrix(RatioScores))[(pos[i]+1):pos[i+1]]
        tmpname2 <- unlist(strsplit(tmpname2,"/"))[2*(1:(length(MetaboliteRatios)-i))]

        names(tmp) <- c(tmpname1, tmpname2)
      }
      MetaboliteRatios[[i]]<- cbind(MetaboliteRatios[[i]],as.matrix(tmp))
    }
  }
  # labeling of list elements
  names(MetaboliteRatios) <- colnames(pbiall[[1]][4:ncol(pbiall[[1]])])
  return(MetaboliteRatios)
}

#-----------------------------------------------------------------------------
.plotHeatmap <- function(data, filepath, thresholdList){

  my_palette =c("darkblue", "blue", "skyblue1", "white", "lightsalmon", "red", "firebrick")

  if(is.vector(thresholdList)){
    col_breaks <- as.numeric(c(thresholdList['minVal'], thresholdList["strongLim"],
                               thresholdList['modLim'], thresholdList['weakLim'],
                               thresholdList['weakLim'], thresholdList['modLim'],
                               thresholdList['strongLim'], thresholdList['maxVal']))
    cutoff <- as.numeric(thresholdList["weakLim"])

  } else if(is.list(thresholdList)){
    col_breaks <- as.numeric(c(thresholdList[[i]]['minVal',], thresholdList[[i]]["strongLim",],
                               thresholdList[[i]]['modLim',], thresholdList[[i]]['weakLim',],
                               thresholdList[[i]]['weakLim',], thresholdList[[i]]['modLim',],
                               thresholdList[[i]]['strongLim',], thresholdList[[i]]['maxVal',]))
    cutoff <- as.numeric(thresholdList[[i]]["weakLim",])
  }

  col_breaks[2:4] <- col_breaks[2:4]*(-1)

  for(i in 1:length(data)){
    colBreaks <- col_breaks
    minVal <- min(data[[i]])
    maxVal <- max(data[[i]])

    colBreaks[1] <- minVal
    colBreaks[8] <- maxVal

    if(abs(minVal) < thresholdList[’strongLim’]){colBreaks[1:2] <- c(minVal, minVal+0.01)}
    if(abs(maxVal) < thresholdList[’strongLim’]){colBreaks[7:8] <- c(maxVal-0.01, maxVal)}

    cellnoteCol <- vector(’character’, numel(data[[i]]))
    controlVar <- numel(data[[i]])

    # generate cell note color vector for one heatmap
    for(j in 1:nrow(data[[i]])){
      for(k in ncol(data[[i]]):1){
        tempVal <- data[[i]][j,k]
        if(abs(tempVal)>(cutoff)){
          cellnoteCol[controlVar] <- "white"
        }else{cellnoteCol[controlVar] <- "black"}
        controlVar <- controlVar-1
      }
    }
    png(filename=paste(filepath,’Metabolite ’,names(data[i]),’.png’, sep=""), width=800, height=600)
    heatmap.2(data[[i]],
              cellnote = round(data[[i]],3),
              main = paste(names(data[i])),
              notecol=cellnoteCol,
              density.info="none",
              trace="none",
              margins =c(8,12),
              col=my_palette,
              breaks=colBreaks,
              dendrogram="none",
              Rowv = FALSE,
              Colv="NA",
              srtRow = 45,
              cexRow=1,cexCol=1,
              key=TRUE,
              lhei=c(0.75,4.25)
    )
    dev.off()
  }
}

#------------------------------------------------------------------------------
plotGraphs <- function(graphList, numGraphToCombine, mycoord, thresh, time,
                       filepathToStorePlots, metaboliteNames, pbiAllRatios, calcMode=’dyn’){

  combGraph <- vector(’list’, length=2)
  names(combGraph) <- c(’discrete’, ’continuous’)

  combGraph[[1]] <- binWeights(graphList)
  combGraph[[2]] <- dynWeights(graphList, pbiAllRatios)

  for(i in 1:length(graphList)){
    png(filename=paste(filepathToStorePlots, ’Network_’, time[i], ’_’, names(thresh), calcMode,’.png’, sep=""), width=800, height=750)
    gplot(intergraph::asNetwork(graphList[[i]]), mode = "circle", gmode=’graph’, coord = mycoord,
          usearrows = FALSE, main=paste(’Network at’, time[i], ’threshold’, names(thresh), calcMode),
          label=names(V(graphList[[i]])), label.pos = 3, label.cex=1,
          vertex.cex=1.5, vertex.col=’orangered’,
          edge.col = ’grey’)
    dev.off()
  }

  pathAdd <- c(’quad’, ’poly4’)
  weightFac <- c(2,4)

  # combined network
  for(j in 1:length(weightFac)){
    dir.create(paste(filepathToStorePlots, pathAdd[j], "/" ,sep=""))

    for(i in 1:length(combGraph)){
      png(filename=paste(filepathToStorePlots, pathAdd[j],'/', 'Network_', names(combGraph[i]),'_weighted_', numGraphToCombine,
                         '_graphs_', names(thresh), calcMode, '.png', sep=""), width=800, height=750)
      gplot(intergraph::asNetwork(combGraph[[i]]), mode = "circle", gmode='graph', coord = mycoord,
            usearrows = FALSE, main=paste("Combined Network with", names(combGraph[i]), 'weighted edges with',calcMode,'threshold'),
            label=names(V(combGraph[[i]])), label.pos = 3, label.cex=1,
            vertex.cex=1.5, vertex.col='orangered',
            edge.lwd = 7.75*(E(combGraph[[i]])$weights)^weightFac[j]+0.25, edge.col = 'grey')
      dev.off()
    }
  }
}

#-------------------------------------------------------------------------------
binWeights <- function(graphList){

  # create combined graph out of given number of graphs (at given time points)
  gall <- igraph::graph.empty(n=0, directed=FALSE)
  adjsum <- 0

  for (i in 1:length(graphList)){
    if(length(V(gall))==0){
      gall <- graphList[[i]]
    } else{
      gall <- igraph::union(gall, graphList[[i]])
    }
    # calculate edge weights for combined plot
    adjsum <- adjsum + get.adjacency(graphList[[i]], sparse=FALSE)
  }
  # rescale adjsum with maximum value to get rangen [0,1]
  adjsum <- adjsum/max(adjsum)

  # get edgelists from gall
  nodeEdgeFrom <- get.edgelist(gall)[,1]
  nodeEdgeTo <- get.edgelist(gall)[,2]

  # get positions where each vertex of nodeEdgeFrom/nodeEdgeTo occurs in adjacency martrix
  posx<-NA
  posy<-NA
  for(i in 1:length(nodeEdgeFrom)){
    posy[i] <- which(rownames(get.adjacency(gall))==nodeEdgeFrom[i])
    posx[i] <- which(colnames(get.adjacency(gall))==nodeEdgeTo[i])
  }
  # find all positions of nodeEgdeFrom/To in adj matrix and store increase weights attribute
  # when multiple occurence
  for(i in 1:length(posx)){
    E(gall)$weights[i] <- adjsum[posx[i],posy[i]]
  }
  return(gall)
}

#-------------------------------------------------------------------------------
dynWeights <- function(graphList, pbiAllRatios){

  weightedAdjMat <- vector('list', length(graphList))
  names(weightedAdjMat) <- colnames(pbiAllRatios)

  for(i in 1:ncol(pbiAllRatios)){
    tempGraph <- graphList[[i]]
    threshold <- quantile(abs(pbiAllRatios[,i]), 0.75)
    ratiosAboveThresh <- abs(pbiAllRatios[which(abs(pbiAllRatios[,i])>threshold),i])

    normRatios <- vector('numeric', length=length(ratiosAboveThresh))
    names(normRatios)<-names(ratiosAboveThresh)
    #calculate normalzed ratios of edges included in the graph
    for(j in 1:length(ratiosAboveThresh)){
      # normalize ratio scores to max at each time point
```

```r
      normRatios[j] <- ratiosAboveThresh[j]/max(ratiosAboveThresh)
    }
    adjMat <- get.adjacency(tempGraph, sparse=FALSE)

    for(k in 1:length(names(normRatios))){
      splitName <- strsplit(names(normRatios)[k], '/')
      rowName <- splitName[[1]][1]
      colName <-  splitName[[1]][2]

      adjMat[rowName, colName] <- normRatios[k]
      adjMat[colName, rowName] <- normRatios[k]
    }
    weightedAdjMat[[i]] <- adjMat
  }
  sumAdjMat <- matrix(0, nrow=nrow(weightedAdjMat[[1]]), ncol=ncol(
      weightedAdjMat[[1]]))
  for(i in 1:length(weightedAdjMat)){
    sumAdjMat <- sumAdjMat + weightedAdjMat[[i]]
  }
  # rescale sumAdjMat with maximum value to get rangen [0,1]
  sumAdjMat <- sumAdjMat/max(sumAdjMat)

  # create graph out of all graphs
  gall <- igraph::graph.empty(n=0, directed=FALSE)

  for (i in 1:length(graphList)){
    if(length(V(gall))==0){
      gall <- graphList[[i]]
    } else{
      gall <- igraph::union(gall, graphList[[i]])
    }
  }

  # get edgelists from gall
  nodeEdgeFrom <- get.edgelist(gall)[,1]
  nodeEdgeTo <- get.edgelist(gall)[,2]

  #get positions where each vertex of nodeEdgeFrom/nodeEdgeTo occurs in martrix
  posx<-NA
  posy<-NA
  for(i in 1:length(nodeEdgeFrom)){
    posy[i] <- which(rownames(get.adjacency(gall))==nodeEdgeFrom[i])
    posx[i] <- which(colnames(get.adjacency(gall))==nodeEdgeTo[i])
  }
  # find all positions of nodeEgdeFrom/To in adj matrix and store increase
  # weights attribute when multiple occurence
  for(i in 1:length(posx)){
    E(gall)$weights[i] <- sumAdjMat[posx[i],posy[i]]
  }
  return(gall)
}

#---------------------------------------------------------------
threeDimPlot <- function(graphList, metaboliteNames, mycoord, nodepos,
                         numGraphToCombine){

  ID <- seq(1,length(metaboliteNames),1)
  tempID <- cbind(ID, metaboliteNames, x=mycoord[,1], y=mycoord[,2])

  # initialize edgelist for graph 1 to 3
  edgeList <- vector('list',numGraphToCombine)
  for(i in 1:numGraphToCombine){
    edgeList[[i]]<- get.edgelist(graphList[[i]])
  }

  posxfrom <- tempID[metaboliteNames==edgeList[[i]][1,1]][3]
  posxto <- tempID[metaboliteNames==edgeList[[i]][1,2]][3]
  posyfrom <- tempID[metaboliteNames==edgeList[[i]][1,1]][4]
  posyto <- tempID[metaboliteNames==edgeList[[i]][1,2]][4]

  # create 3D plot
  open3d(params = getr3dDefaults(),useNULL = rgl.useNULL())

  posz <- seq(3, 3*numGraphToCombine, 3)
  zrep <- nrow(mycoord)
  mycol <- c(rep('grey',zrep), rep('blue',zrep), rep('red',zrep),
             rep('darkgreen',zrep), rep('orange', zrep))
  mycol2 <- c('grey', 'blue', 'red', 'darkgreen', 'orange')

  # set coordinates for all networks -> replicate them
  NodesToPlot <- cbind(x=rep(mycoord[,1],numGraphToCombine),
                       y=rep(mycoord[,2],numGraphToCombine),
                       z=sort(rep(posz, zrep)))

  spheres3d(NodesToPlot[,1], NodesToPlot[,2], NodesToPlot[,3],
            radius=0.1, color=mycol)
  for(i in 1:numGraphToCombine){
    text3d(mycoord[,1],mycoord[,2],posz[i]-0.2, text=seq(1,zrep,1),
           color=mycol2[i])
  }

  #z-axis
  arrow3d(p0=c(-1.5,-1.5,0),p1=c(-1.5,-1.5,max(3*numGraphToCombine+3)),
          type = "extrusion", col='black', s= 0.05, width = 0.01,
          thickness = 0.1)
  text3d(-1.7,-1.7,1, texts = 'time', color='black')


  for(j in 1:length(edgeList)){
    for (i in 1:nrow(edgeList[[j]])){
      posxfrom <- tempID[metaboliteNames==edgeList[[j]][i,1]][3]
      posxto <- tempID[metaboliteNames==edgeList[[j]][i,2]][3]
      posyfrom <- tempID[metaboliteNames==edgeList[[j]][i,1]][4]
      posyto <- tempID[metaboliteNames==edgeList[[j]][i,2]][4]

      lines3d(c(posxfrom,posxto),c(posyfrom,posyto),c(posz[j],posz[j]), color=
            mycol2[j])
    }
  }

  for(i in 1:numGraphToCombine){
    planes3d(0,0,1, -posz[i], alpha=0.2, col=mycol2[i])
    text3d(-1.3,-1.3,posz[i], texts = time[i], color=mycol2[i])
  }

  mylegend <-NULL
  for(i in 1:length(metaboliteNames)){
    mylegend <- paste(mylegend, i, '-', metaboliteNames[i], '\n')
  }

  bgplot3d({
    plot.new()
    title(main = 'Network evolution over time', line = 3, cex.main=1.5)
    mtext(side = 1, mylegend, line = 1, adj=0, cex=1.2)
  })

}

#------------------------------------------------------------------------------
.createGraph <- function(biscores, nodes, threshold, edge.file, plotGraph)

  significant <- names(.getSignificantValues(abs(biscores), threshold))
  if(length(significant)==0)
    stop("There are no ratios higher than the defined threshold!")
  g <- .getGraph(nodes, significant)
  V(g)$label <- V(g)$name

  if(!is.null(edge.file))
  {
    edgelist <- .getEdgelist(nodes, significant)
    write.table(edgelist,file=edge.file, row.names = FALSE, col.names = FALSE,
                quote=FALSE)
  }
  E(g)$scores <- biscores[significant]
  if(plotGraph)
    plot(g)
  return(g)
}
```

```

#-------------------------------------------------------------------------------
.plotScores <- function(scores, numTopRankedToPlot, method,
                        bars.cols = c("red","blue"), time)
{
   numTopRanked <- min(numTopRankedToPlot, length(scores))
  scoresRanked <- scores[match(sort(abs(scores), decreasing = TRUE), abs(scores))]
  scoresTopRanked <- scoresRanked[1:numTopRanked]
  cols <- ifelse(scoresTopRanked >= 0, bars.cols[1], bars.cols[2])
  zoom <- 1.5

  #original.params <- par()
  par(mar = c(11, 4, 2, 2) + 0.2) #add room for the rotated labels

  barPos <- barplot(scoresTopRanked, space=0.2,
                    main=paste(method, "scores (top", numTopRanked, "ranked metabolites) at",
                               time, sep=" "),
                    ylab=method, col=cols, cex.axis = zoom, cex.names=zoom/2,
                    cex.lab=zoom, cex.main=zoom, xaxt="n")

  lablist<-names(scoresTopRanked)
  text(barPos, par("usr")[3]-6, srt = 60, adj= 1, xpd = TRUE,
       labels =lablist, cex=1)
}
```

Bibliography

[1] G. D. Lewis, R. Wei, E. Liu, E. Yang, X. Shi, M. Martinovic, L. Farrell, A. Asnani, M. Cyrille, A. Ramanathan, O. Shaham, G. Berriz, P. A. Lowry, I. F. Palacios, M. Taşan, F. P. Roth, J. Min, C. Baumgartner, H. Keshishian, T. Addona, V. K. Mootha, A. Rosenzweig, S. A. Carr, M. A. Fifer, M. S. Sabatine, and R. E. Gerszten, "Metabolite profiling of blood from individuals undergoing planned myocardial infarction reveals early markers of myocardial injury," *The Journal of Clinical Investigation*, vol. 118, no. 10, pp. 3503–3512, 2008.

[2] M. Netzer, K. M. Weinberger, M. Handler, M. Seger, X. Fang, K. G. Kugler, A. Graber, and C. Baumgartner, "Profiling the human response to physical exercise: a computational strategy for the identification and kinetic analysis of metabolic biomarkers," *Journal of Clinical Bioinformatics*, vol. 1, no. 1, p. 34, 2011.

[3] C. Baumgartner, M. Osl, M. Netzer, and D. Baumgartner, "Bioinformatic-driven search for metabolic biomarkers in disease," *Journal of Clinical Bioinformatics*, vol. 1, no. 2, 2011.

[4] G. A. Pavlopoulos, M. Secrier, C. N. Moschopoulos, T. G. Soldatos, S. Kossida, J. Aerts, R. Schneider, and P. G. Bagos, "Using graph theory to analyze biological networks," *BioData Mining*, vol. 4, no. 10, 2011.

[5] National Center for Biotechnology Information. (1986) MeSH: Biomarkers. Accessed: 2017-01-23. [Online]. Available: https://www.ncbi.nlm.nih.gov/mesh?term=biomarker

[6] Biomarkers Definitions Working Group, "Biomarkers and surrogate endpoints: Preferred definitions and conceptual framework," *Clinical Pharmacology & Therapeutics*, vol. 69, no. 3, pp. 89–95, 2001.

[7] R. S. Vasan, "Biomarkers of Cardiovascular Disease," *Circulation*, vol. 113, no. 19, pp. 2335–2362, 2006.

[8] WHO Internatioal Programme on Chemical Safety. (2001) Biomarkers in Risk Assessment: Validity and Validation. Accessed: 2017-01-04. [Online]. Available: http://www.inchem.org/documents/ehc/ehc/ehc155.htm

[9] K. Strimbu and J. A. Tavel, "What are Biomarkers?" *Current opinion in HIV and AIDS*, vol. 5, no. 6, pp. 463–466, 2010.

[10] A. B. Hill, "The Environment and Disease: Association or Causation?" *Proceedings of the Royal Society of Medicine*, vol. 58, no. 5, pp. 295–300, 1965.

[11] J. K. Aronson, "Biomarkers and surrogate endpoints," *British Journal of Clinical Pharmacology*, vol. 59, no. 5, pp. 491–494, 2005.

[12] X. Wang, C. Baumgartner, D. Shields, H. Deng, and J. Beckmann, *Application of Clinical Bioinformatics*, ser. Translational Bioinformatics. Springer Netherlands, 2016.

[13] R. Bosch, X. Zhang, and N. Sandker, "Study Design Issues in Evaluating Immune Biomarkers," *Current opinion in HIV and AIDS*, vol. 8, no. 2, pp. 147–154, 2013.

[14] B. Dawson and R. Trapp, *Basic & Clinical Biostatistics: Fourth Edition*, ser. LANGE Basic Science. McGraw-Hill Education, 2004.

[15] J. McDermott, J. Wang, H. Mitchell, B.-J. Webb-Robertson, R. Hafen, J. Ramey, and K. D. Rodland, "Challenges in Biomarker Discovery: Combining Expert Insights with Statistical

Analysis of Complex Omics Data," *Expert Opinion on Medical Diagnostics*, vol. 7, no. 1, pp. 37–51, 2013.

[16] H. J. Issaq and T. D. Veenstra, "Chapter 1 - Biomarker Discovery: Study Design and Execution," in *Proteomic and Metabolomic Approaches to Biomarker Discovery*, H. J. Issaq and T. D. Veenstra, Eds. Academic Press, 2013, pp. 1–16.

[17] B. Heejung, M. Davidiana, X. K. Zhou, H. L. van Epps, and M. Mazumdar, *Statistical Methods in Molecular Biology*, 1st ed., ser. Methods in Molecular Biology 620. Humana Press, 2010.

[18] S. H. Huang, "Supervised feature selection: A tutorial," *Artif. Intell. Research*, vol. 4, no. 2, pp. 22–37, 2015.

[19] Y. Saeys, I. Iñaki, and P. Larrañaga, "A Review of Feature Selection Techniques in Bioinformatics," *Bioinformatics*, vol. 23, no. 19, pp. 2507–2517, 2007.

[20] M. Kanehisa and S. Goto, "KEGG: Kyoto Encyclopedia of Genes and Genomes," *Nucleic Acidy Res.*, vol. 28, pp. 27–30, 2000.

[21] R. Kohavi, "A Study of Cross-validation and Bootstrap for Accuracy Estimation and Model Selection," in *Proceedings of the 14th International Joint Conference on Artificial Intelligence - Volume 2*. Morgan Kaufmann Publishers Inc., 1995, pp. 1137–1143.

[22] E. G. Cerami, G. D. Bader, B. E. Gross, and C. Sander, "cPath: open source software for collecting, storing, and querying biological pathways," *BMC Bioinformatics*, vol. 7, no. 1, p. 497, 2006.

[23] M. A. Harris, J. Clark, A. Ireland, J. Lomax, M. Ashburner, R. Foulger, K. Eilbeck, S. Lewis, B. Marshall, C. Mungall, J. Richter, G. M. Rubin, J. A. Blake, C. Bult, M. Dolan, H. Drabkin, J. T. Eppig, D. P. Hill, L. Ni, M. Ringwald, R. Balakrishnan, J. M. Cherry, K. R. Christie, M. C. Costanzo, S. S. Dwight, S. Engel, D. G. Fisk, J. E. Hirschman, E. L. Hong, R. S. Nash, A. Sethuraman, C. L. Theesfeld, D. Botstein, K. Dolinski, B. Feierbach, T. Berardini, S. Mundodi, S. Y. Rhee, R. Apweiler, D. Barrell, E. Camon, E. Dimmer, V. Lee, R. Chisholm, P. Gaudet, W. Kibbe, R. Kishore, E. M. Schwarz, P. Sternberg, M. Gwinn, L. Hannick, J. Wortman, M. Berriman, V. Wood, N. de la Cruz, P. Tonellato, P. Jaiswal, T. Seigfried, and R. White, "The Gene Ontology (GO) database and informatics resource." *Nucleic Acids Res*, vol. 32, pp. D258–61, 2004.

[24] "Overview of KEGG rescource," accessed: 2016-03-02. [Online]. Available: http://www.kegg.jp/kegg/kegg1a.html

[25] World Health Organization, "World health statistics," Tech. Rep., 2009.

[26] D. Chan and L. Ng, "Biomarkers in acute myocardial infarction," *BMC Medicine*, vol. 8, no. 34, 2010.

[27] E. Antman, J.-P. Bassand, W. Klein, M. Ohman, J. L. L. Sendon, L. Rydén, M. Simoons, and M. Tendera, "Myocardial infarction redefined — a consensus document of The Joint European Society of Cardiology/American College of Cardiology committee for the redefinition of myocardial infarction: The Joint European Society of Cardiology/ American College of Cardiology Committee," *Journal of the American College of Cardiology*, vol. 36, no. 3, pp. 959–969, 2000.

[28] S. Mythili and M. Narasimhan, "Diagnostic Markers of Acute Myocardial Infarction," *Biomedical Reports*, vol. 3, no. 6, p. 743–748, 2015.

[29] F. S. Apple, "Tissue specificity of cardiac troponin I, cardiac troponin T and creatine kinase-MB," *Clinica Chimica Acta*, vol. 284, pp. 151–159, 1999.

[30] J. F. Tucker, R. A. Collins, A. J. Anderson, J. Hauser, J. Kalas, and F. S. Apple, "Early diagnostic efficiency of cardiac troponin I and Troponin T for acute myocardial infarction," *Academic emergency medicine*, vol. 4, no. 1, pp. 3–5, 1997.

[31] C. Baumgartner, G. D. Lewis, M. Netzer, B. Pfeifer, and R. E. Gerszten, "A new data mining approach for profiling and categorizing kinetic patterns of metabolic biomarkers after myocardial injury," *Bioinformatics*, vol. 26, no. 14, pp. 1745–1751, 2010.

[32] B. Carré, *Graphs and networks*, ser. Oxford applied mathematics and computing science series. Clarendon Press, 1979.

[33] J. Aldous, S. Best, and R. Wilson, *Graphs and Applications: An Introductory Approach.* Springer London, 2003, no. 1.

[34] J. McHugh, *Algorithmic graph theory.* Prentice Hall PTR, 1990.

[35] R Development Core Team, *R: A Language and Environment for Statistical Computing*, R Foundation for Statistical Computing, Vienna, Austria, 2008. [Online]. Available: http://www.R-project.org

[36] G. Csardi and T. Nepusz, "The igraph software package for complex network research," *InterJournal*, vol. Complex Systems, p. 1695, 2006.

[37] D. Adler, D. Murdoch, and others, *rgl: 3D Visualization Using OpenGL*, 2016, r package version 0.96.0. [Online]. Available: http://CRAN.R-project.org/package=rgl

[38] T. M. Fruchterman and E. M. Reingold, "Graph drawing by force-directed placement," *Software: Practice and experience*, vol. 21, no. 11, pp. 1129–1164, 1991.

[39] M. M. Mihaylova and R. J. Shaw, "The AMP-Activated Protein Kinase (AMPK) Signaling Pathway Coordinates Cell Growth, Autophagy, & Metabolism," *Nature cell biology*, vol. 13, no. 9, pp. 1016–1023, 2011.

[40] Histidine Metabolism. Accessed: 2017-03-07. [Online]. Available: http://www.genome.jp/kegg-bin/show_pathway?scale=1.0&query=Purine&map=map00340&scale=&auto_image=&show_description=hide