Prediction of metabolomic and transcriptomic patient profiles

Dissertation zu Erlangung des Doktorgrades
der Naturwissenschaften (Dr. rer. nat.)
der Fakultät Biologie und Vorklinische Medizin

der Universität Regensburg

vorgelegt von

Inka Appel

aus Dachau

im Jahr 2011

Promotionsgesuch eingereicht am: 24.11.2011
Die Arbeit wurde angeleitet von: Prof. Dr. Wolfram Gronwald

Inka Appel Regensburg, 24.11.2011

Preface

Acknowledgements This work was carried out in the department of *Statistical Bioinformatics* of the Institute of Functional Genomics at the University of Regensburg. I thank all past and present colleagues for the good working atmosphere and fruitful scientific and non-scientific discussions. Furthermore, the thesis research has been funded within the framework of the research project "Functional and translational genomics for improved clinical outcome of acute leukemias" supported by the National Genome Research Network NGFN and NGFNplus, and the Bavarian Genome Network BayGene.

I am grateful to my supervisors *Rainer Spang* and *Wolfram Gronwald* for their constant advice. Especially, I thank Claudio Lottaz for giving me the opportunity to carry out my thesis research in the leukemia research project which allowed me to collaborate with experts from many different disciplines.

While working on this thesis I met many people who made contributions either direct or indirect. In particular I greatfully thank *Claudio Lottaz*, *Stefan Bentink*, *Juby Jacob*, *Julia Engelmann*, *Martin Almstetter*, *Katja Dettmer*, *Katharina Meyer*, and *Peter Butzhammer*. Special thanks go to *Toni Moll* and *Daniela Herold* who read drafts of this thesis and greatly improved it by their comments.

I am especially grateful to *Johannes*, my boyfriend, and his and my whole family for encouraging me throughout the time it took to finish the presented work.

Publications Most contents of this thesis have been published in scientific journals. The alignment algorithm INCA for comprehensive two-dimensional gas chromatography - time-of-flight mass spectrometry (GC×GC-TOF-MS) based metabolomics data in Chapter 3 has been published in *Analytical Chemistry* [4] (joint trust authorship with Martin F. Almstetter, Institute of Functional Genomics, Prof. Dr. Peter Oefner, University of Regensburg). A subsequent application of the algorithm to a serum and plasma sample collection with regard to their suitability for metabolomics

studies was published in *Electrophoresis* [19]. The comparison of INCA to the Statistical Compare alignment tool, which recently became commercially available as an add-on function for ChromaTOF version 4, is published in the *Journal of Chromatography A* [3] (joint trust authorship with Martin F. Almstetter). Furthermore, the estimation methods of local classification probabilities of diagnosis and their comparison to known probability estimators are published in *Bioinformatics* [7].
The detailed list of publications can be found in Chapter A.3.2.

Contents

List of Figures

List of Tables

Summary

Genetic and environmental conditions lead to global changes in the chemical composition of biological systems. The extended analysis of cellular metabolic pathway products gives insights into the functionality of enzymes in normal and pathological conditions. In the last years, metabolomics acquired an important role in detecting prognostic factors for various diseases including polycystic kidney disease.
Effective methods employed for metabolomic studies are nuclear magnetic resonance (NMR) spectroscopy and coupling two-dimensional gas chromatography with time-of-flight mass spectrometry (GC×GC-TOF-MS). For both technologies, software exists that detects signals within the raw data as candidates for metabolites. Some of these candidates are verified as metabolites by comparison to a metabolite library, many remain unknown. The metabolite candidates and the number of metabolite candidates vary across different measurements. While NMR covers up to 150 metabolites, GC×GC-TOF-MS detects thousands of metabolites in one measurement. Hence, combining different meseaurements by GC×GC-TOF-MS is ambitious and required to apply statistical methods and machine learning techniques.
Thus, we developed the integrative normalization and comparative analysis software tool INCA. INCA automatically identifies equal metabolite candidates among different measurements and combines them into one data matrix. The alignment algorithm is validated by an spike-in experiment and successfully applied to various metabolomic datasets. A commercial software tool was provided one year after completion of INCA. It is based on similar parameters and performs comparable.

Not only the detection of new prognostic factors is of interest but also the prediction of treatment response or disease status. Autosomal polycystic kidney disease (ADPKD) is a frequent cause of kidney failure. It is usually diagnosed at a progressed stage of renal cystic transformation due to a lack of reliable laboratory tests early in the disease. Hence, the prognosis of patients to develop ADPKD is challenging.
In terms of classification, prognosis is associated with the probability for developing

the disease. Healthy patients and patients where the disease is reliably diagnosed are used to learn a classifier. This classifier consists of a set of features that defines a decision rule separating the healthy from the diseased, and assigning a new patient to the correct class with high probability. In microarray based classification, the performance of classification algorithms has been analyzed in great detail. However, little attention has been given to the usefulness of probability estimates and this is even more true for metabolomic analyses. Thus, I developed probability estimation methods based on local errors and compared them to existing methods from gene expression profiling, text categorization and digit recognition.

I show that the local error based methods perform superior to more widely used methods, the PAM program, binary regression, and Compound Bayes classifiers. Especially the PAM approach performs poorly because its probability estimates depend on the number of selected features. I recommend not to make use of these estimates in the context of clinical diagnosis of patients. Although the estimators are evaluated on metabolomics data, I believe that similar results are obtained for different forms of clinical diagnosis based on high dimensional readouts, e.g. proteomic or transcriptomic profiling data. From the perspective of probability estimation the effective dimensionality is that of the feature signature and not that of the original data set. The dimensionality of gene expression based signatures described in the literature is well comparable to the metabolomics dataset.

Thesis organization

Most of the probability estimation methods are used in combination with different classifiers for gene expression data. Thus, I developed my local error based estimation methods on microarray datasets.

Chapter 1 starts with an introduction to transcriptomic and metabolomic data. The metabolomics techniques nuclear magnetic resonance spectroscopy and two-dimensional gas chromatography mass spectrometry are described in more detail.

In **Chapter 2**, classification algorithms and existing probability estimation methods from different fields are reviewed.

The alignment algorithm INCA for two-dimensional gas chromatography mass spectrometry data is presented and evaluated in **Chapter 3** together with its applications and a comparison to a commercial software tool.

Finally, my local error based probability estimation methods are defined and compared to existing methods based on several evaluation criteria (**Chapter 4**).

Zusammenfassung

Genetische Faktoren und Umwelteinflüsse verändern die chemischen Zusammensetzung biologischer Systeme global. Eine detaillierte Analyse zellulärer Stoffwechselprodukte bei normalen und pathologischen Bedingungen gibt Einblicke in die Funktionalität von Enzymen und anderen Metaboliten. In jüngster Zeit erfährt die Untersuchung von Metaboliten und deren Zusammenspiel bei der Detektierung prognostischer Faktoren verschiedenster Krankheiten, wie zum Beispiel Zystennieren, erhöhte Aufmerksamkeit.

Nukleare Magnetresonanzspektroskopie (NMR) und das Koppeln von zwei-dimensionaler Gaschromatographie an Flugzeitmassenspektrometrie (GC×GC-TOF-MS) werden erfolgreich bei metabolomischen Studien angewandt. Für beide Technologien gibt es Software, die in den Rohdaten potentielle Metabolite identifiziert. Einige dieser Kandidaten werden durch einen Abgleich mit einer Datenbank als Metabolite erkannt, viele bleiben unbekannt. Die Metabolitkandidaten und die Anzahl der Metabolitkandidaten ist in verschiedenen Messungen unterschiedlich. NMR kann bis zu 150 Metabolite in einer Messung identifizieren, GC×GC-TOF-MS mehrere Tausend. Das Verknüpfen verschiedener Messungen ist somit eine Herausforderung, aber unumgänglich vor dem Auswerten der Daten mittels statistischer Methoden und Techniken des maschinellen Lernens.
Zu diesem Zweck habe ich die Software INCA, kurz für "Integrative Normalization and Comparative Analysis", entwickelt. INCA erkennt automatisch gleiche Metabolitkandidaten über verschiedene Messungen hinweg und fasst diese in einer Datenmatrix zusammen. Der Alignmentalgorithmus wird durch ein Spike-In Experiment validiert und erfolgreich auf verschiedene metabolomische Datensätze angewendet. Ein Jahr nach der Veröffentlichung von INCA wurde von Dritten eine kommerzielle Software angeboten. Diese basiert auf ähnlichen Parametern und erzielt vergleichbare Resultate.

Nicht nur die Identifizierung neuer prognostischer Faktoren ist wichtig, sondern auch das Vorhersagen einer Therapieantwort oder des Krankheitsstadiums. Autosomale polyzystische Nieren (ADPKD) sind ein häufiger Grund für Nierenversagen. In der Regel wird die Krankheit erst in einem fortgeschrittenen Stadium der Zystennierentransformation diagnostiziert. ADPKD bei nierenkranken Patienten früh zu diagnostizieren ist eine Herausforderung.
Im Bereich Klassifikation wird die Prognose mit der Wahrscheinlichkeit assoziiert zu erkranken. Messungen von gesunden und kranken Patienten werden benutzt um einen Klassifikator zu lernen. Dieser Klassifikator besteht aus einer Menge von Eigenschaften, die die gesunden von den kranken Patienten trennt, und einen neuen Patienten mit hoher Wahrscheinlichkeit in die richtige Klasse einordnen kann. Bei der Klassifikation von Microarraydaten ist die Performance der Klassifikationsalgorithmen bereits detailliert untersucht. Jedoch wurde die Verlässlichkeit von geschätzten Wahrscheinlichkeiten besonders bei metabolomischen Analysen wenig untersucht. Ich habe Schätzmethoden für Diagnosewahrscheinlichkeiten entwickelt, die auf lokalen Fehlern basieren, und mit bekannten Methoden verglichen, die aus den Bereichen der Mustersuche in Genexpressionsdaten, Textkategorisierung und Ziffernerkennung stammen.
Ich zeige, dass Methoden, die auf lokalen Fehlern basieren, besser sind als weit verbreitete Methoden, wie das PAM Programm, binäre Regression und Compound-Klassifikatoren. Besonders PAM schneidet schlecht ab, da seine Wahrscheinlichkeiten von der Anzahl der ausgewählten Eigenschaften abhängt. Ich empfehle diese Schätzungen nicht im Kontext klinischer Diagnose von Patienten zu verwenden. Im Rahmen dieser Arbeit werden die Schätzer auf metabolomischen Daten ausgewertet und verglichen. Trotzdem lassen sich ähnliche Resultate für verschiedene Formen von klinischer Diagnose hochdimensionaler Daten, zum Beispiel proteomische oder transkriptomische Daten zur Mustersuche, erzielen. Von der Perspektive des Wahrscheinlichkeitschätzens ist die effektive Dimensionalität der Genexpressionssignaturen, wie sie in der Literatur beschrieben wird, gut vergleichbar mit der metabolomischer Datensätze.

Chapter 1

Comparing metabolomics and microarray data

This thesis addresses the prediction of metabolomic and transcriptomic patient profiles. The analysis of metabolomics data is an emerging field whereas the analysis of transcriptomic data using microarrays is well-elaborated. We developed a software tool that is able to combine metabolomics two-dimensional gas chromatography mass spectrometry measurements containing different features and numbers of features. Subsequently, the combined data are used to identify potential novel biomarkers for disease states (see Chapter 3). At time of development almost no software solutions were available. This chapter gives an overview on transcriptomic and metabolomic data (Sections 1.1 and 1.2). The focus will be on metabolomic data since analyses throughout my thesis are applied to this type of data (see Section 1.3).

1.1 From the genome to the metabolome

The *genome* is the complete genetic hereditary information of an organism, and serves as blueprint for all cellular processes. Almost each cell of an organism stores a full copy of the genome. It is encoded in *DNA*, deoxyribonucleid acid, which consists of four building blocks, the *nucleotides*. Connected by sugar and phosphate, they form a single DNA chain. Two chains run in opposite directions to each other building a *double helix*.
The DNA is copied in small stretches, called *transcripts* (or messanger RNA), and

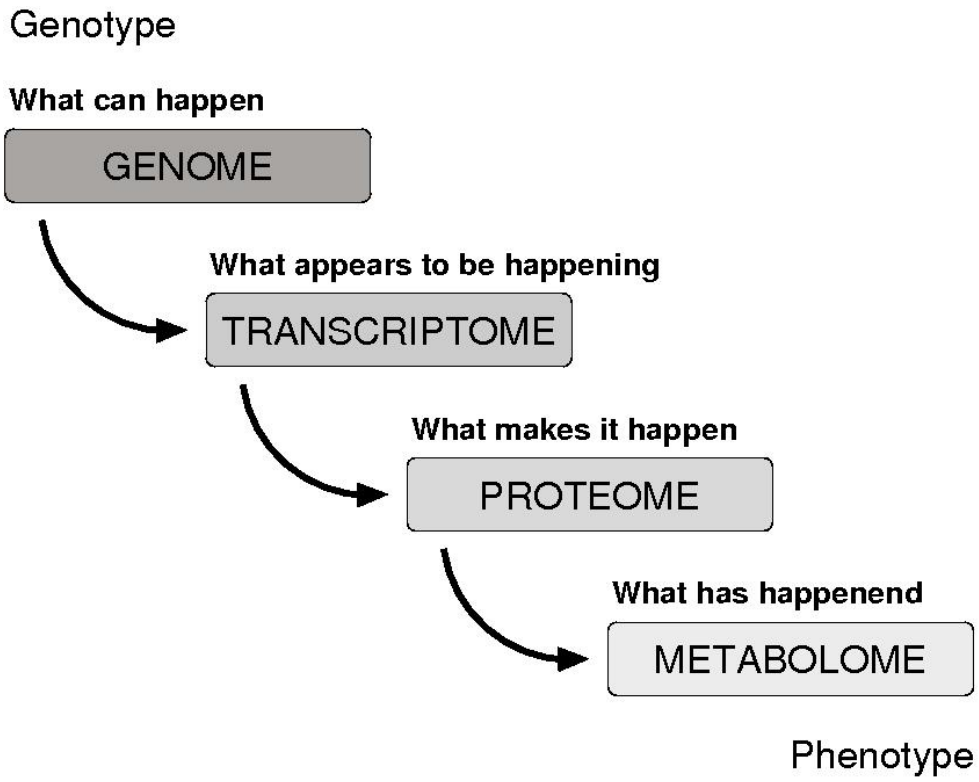


Figure 1.1: The "omics"-cascade: From genotype to phenotype.

further translated into cellular protein molecules. The translation process is determined by the *genetic code*: nucleotide triplets are transformed into 20 different amino acids, the building blocks of proteins. A stretch of DNA coding for a protein is called *gene*. A gene is *expressed* if it is copied or transcribed. The whole pool of transcripts in a cell constitutes the *transcriptome* and reflects the genes that are being actively expressed at any time.
Depending on the environment, *proteins* determine the metabolic capabilities of the cell. In contrast to the genome, the transcriptome and proteome react more easily to external environmental conditions.
Finally, the *metabolome* captures the network of metabolic reactions of a cell, and refers to the complete set of small molecules. These include sugars, amino acids, lipids, other small-molecule metabolites, and signalling molecules.

Dettmer *et al.* [20] summarize the relationships of the genome, transcriptome, proteome, and metabolome in the *"omics"-cascade* (see Figure 1.1).
The cascade also shows the relationship between genotype and phenotype. The genotype is implied by the set of all genes within a cell. Depending on the environmental

influences, it determines the morphological and physiological phenotype. Hence, genomics addresses the genotype whereas metabolomics researches into the phenotype of an organism.

1.2 What can be measured?

A variety of well established as well as still developing technologies is available for each "omics"-science.

Genomics Important techniques associated with genomics involve various sequencing methods. On the level of DNA, single genes or whole genomes are read and compared. Variations of single sequence positions and deletions, insertions or inversions of longer stretches of DNA within a gene can lead to expression changes of many other genes. This in turn can cause cancer. A prominent example is the breast cancer gene BRCA1, which helps to repair damaged DNA or to destroy cells if DNA cannot be repaired (reviewed by Venkitaraman [64]).
Besides sequencing DNA, the amino acid sequence of a protein can be determined to discover its structure and function.

Transcriptomics Measuring gene expression is based on the complementary binding properties of DNA to DNA or DNA to RNA. Genome-wide gene expression changes are measured using the microarray technology. Thousands of spots of different DNA sequences are located on a microarray, each part of a particular gene. Thus, thousands of genes can be analyzed simultaneously and searched for significantly different expression patterns among various conditions or tissues. Differentially expressed genes may be targets for drug development [13, 41] or new biomarkers which can be used to describe a certain tissue or disease state [37, 46, 60].

Proteomics and Metabolomics Proteins and other biomolecules are encoded in genes and therefore may be just as well biomarkers or drug targets. The structures of proteins can be analyzed using X-ray crystallography, the abundances using mass spectrometry (MS) or nuclear magnetic resonance (NMR) spectroscopy methods. Whereas X-ray crystallography is capable of analyzing large proteins and is often used for drug design, the latter methods focus on the analysis of biological fluids and the comparison of disease states or tissues to get insights into signalling pathways, or to detect novel biomarkers. An example within the area of drug design is the

structure of FK506-binding protein (FKBP) which forms a complex with ascomycin, an immunosuppressant [10]. A study of urine NMR data identified drug metabolites in common use in a sample of the U.S. human population, in particular, from acetaminophen and ibuprofen metabolites [34].

1.3 Measuring the metabolome

Metabolomics aims at the comprehensive quantitative analysis of all metabolites in a biological system [2]. It investigates metabolic changes caused by disease, environmental or genetic factors in an organism. This is challenging since the metabolome comprises thousands of small molecules present over a wide range of different concentrations [20, 21]. The number of metabolites varies among organisms and sample types. The prokaryote *E. coli* contains about 750 metabolites [48], whereas eukaryotic systems range from several thousands in humans [68] up to hundreds of thousands in various plant species [31, 50]. The main analytical techniques employed for metabolomic studies are based on nuclear magnetic resonance (NMR) spectroscopy and mass spectrometry (MS) [39].

In NMR spectroscopy a sample is exposed to a electromagnetic field. Atoms with specific nuclear properties (like ^{1}H, ^{13}C and ^{31}P) absorb electromagnetic radiation and emit a resonance frequency depending on the strength of the magnetic field and their position in the molecule. This frequency is recognized by a detector and translated to a single data signal using Fourier transformation. An example spectrum of a urine sample is shown in Figure 1.2. The Fourier-transformed signals (x-axis) are plotted against their intensities (y-axis). A compound consists of one or more intensity signals: citrate is represented by 4 signals, whereas formate has only one signal. Signals overlay and are disturbed by the water signal. 20 to 100 metabolites are identified in one measured NMR sample. Wishart estimates the number of metabolites for a sample to a maximum of 150 [69]. The need for high concentration levels limits the detection of metabolites. Concentrations must be in the range of $\mu g/ml$. Thus, NMR can detect hydrophilic compounds like sugars, amines and volatile ketones.

In mass spectrometry, the sample is vaporized (transferred to the gas phase), and the components of the sample are ionized and depending on the ionization used broken down into fragment ions by a ion source. The ions are separated according to their mass-to-charge ratio and recorded as different mass traces by a detector. To get rid

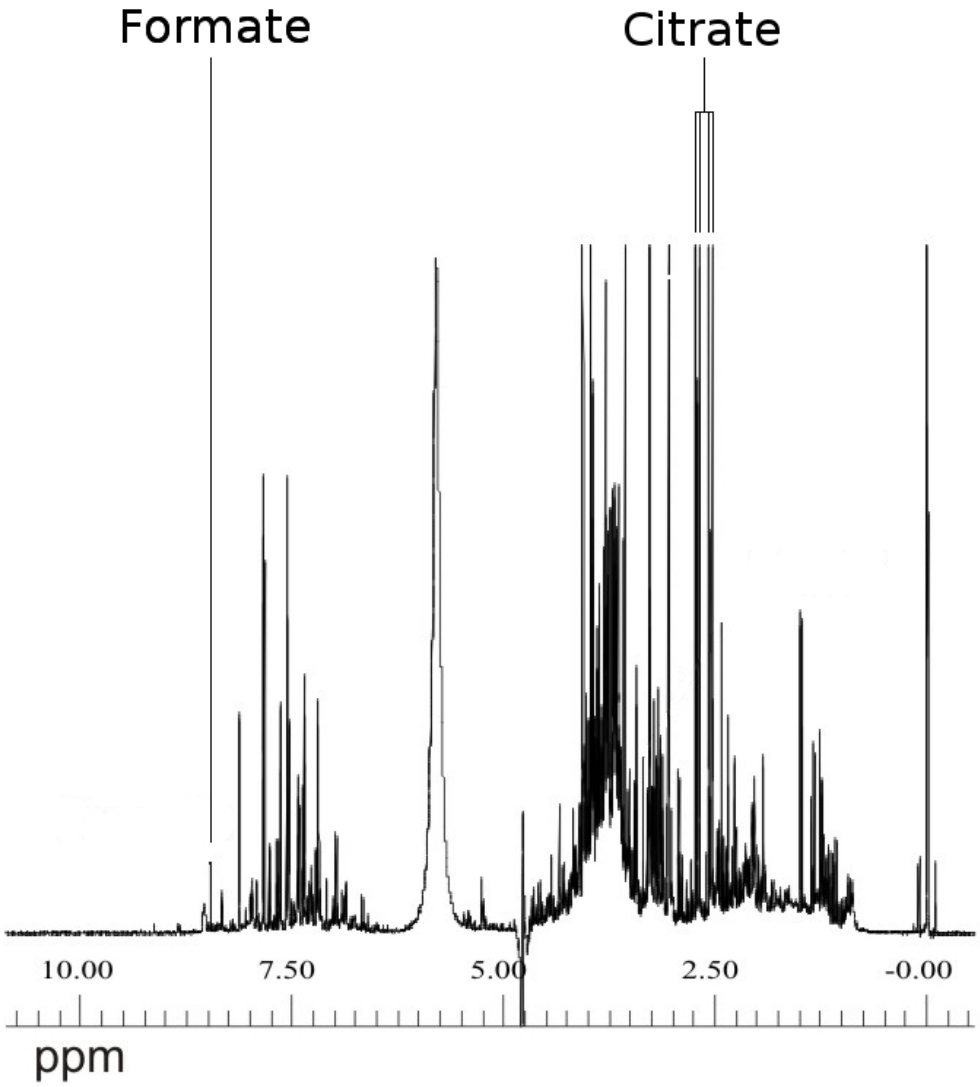


Figure 1.2: In a one-dimensional nuclear magnetic resonance (1D-NMR) spectrum of a urine sample Fourier-transformed signals are plotted against their signal intensities. The unit of measurement is parts per million (ppm).

of overlaying signals, metabolic components are pre-separated by either gas chromatography (GC), or liquid chromatography (LC). For GC, the components must be transferred to and maintained in the gas phase. This requires a sufficient vapor pressure. To extend the range of metabolites amenable to GC analysis derivatization can be performed to reduce the polarity and boiling point of the analytes. Frequently, silylation (here adding a trimethylsilylester group) in combination with methoximation is used for metabolic fingerprinting [30]. The sample is vaporized in an injector and sent through a silica capillary, the column, which is coated with the stationary phase, in most cases a highly viscous liquid. The metabolites leave the column at different time points depending on their vapor pressure and in case of polar stationary phases depending on their vapor pressure and polarity. Subsequently, they are detected by the mass spectrometer. A GC-MS chromatogram (see Figure 1.3) shows signal intensities of metabolites (y-axis) arriving at different time points at the detector (x-axis). Each data point represents a mass spectrum, which is the sum of all fragment ions and their intensities of a metabolite. The time is called retention time. In analogy to NMR, one metabolite consists of several signals and signals overlay. The detection limits of MS range from pg/ml to $\mu g/ml$. GC-MS methods concentrate on hydrophobic compounds and provide up to 2000 metabolites per measurement which corresponds to the number of human metabolites currently known [20].

MS is 10-fold more sensitive and covers much more metabolites per measurement than NMR. Both methods produce a spectrum. The ionisation destroys the metabolites, in NMR the sample can be measured several times. Choosing a metabolomics method depends on the substances to be measured, and their concentrations. Both methods are used for the comparative analysis of biological fluids like urine, serum or other tissue extracts.

The comparative analysis of disease states requires the identification of informative signals. The chemometric approach directly compares raw spectra to identify relevant spectral features. Each spectrum is divided into bins and a spectral feature is given by the area under the spectrum for each bin. These features are then used to identify the corresponding metabolites. In metabolomics the metabolites of each measurement are identified by comparing the raw spectra to a spectral reference library first. Subsequently, these metabolites are searched for biomarkers or informative pathways [69, 25]. The discovery of novel compounds as biomarker candidates is also called metabolic profiling, the detection of diagnostic pathways metabolic fingerprinting.

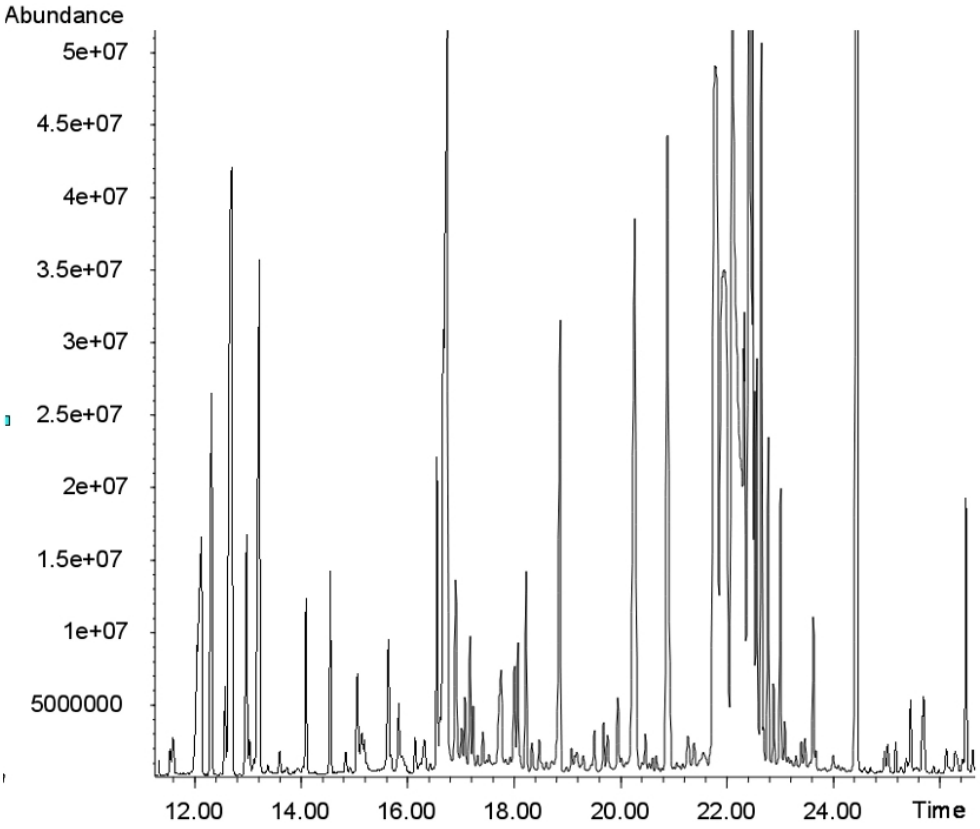


Figure 1.3: A one-dimensional gas chromatography mass spectrometry (1D-GC-MS) chromatogram shows signal abundances for different time points depending on the arrival of the metabolite at the detector.

1.4 Metabolomics using two-dimensional gas chromatography time-of-flight mass spectrometry data

Comprehensive two-dimensional gas chromatography (GC×GC) is suited for the analysis of complex samples [15, 1]. It offers a multiplicative increase in peak capacity by combining two columns with a thermal or flow based modulator, with thermal modulators being more prevalent. The thermal modulator focuses the sample from the first column periodically by freezing and heating in small segments that are then transferred to the second column [16, 9] (see Figure 1.4). This yields a second dimension retention time, a wider separation space, very narrow peaks, and thus, lower limits of detection. A chromatogram is commonly visualized by a two-dimensional plot where the axes denote the retention times and a color code the peak intensities (Figure 1.5). Blue indicates no signal, red high intensity. In the figure, the signal intensity is the total ion count of all fragment ions for a metabolite. A chromatogram can be plotted for each fragment ion separately as well. For broader signals is not clear whether more than one metabolite is present, e.g. the signal left of succinate. Many signals are less intensive and may or may not be a true metabolite. The set of all identified signals are defined as metabolite candidates. The term peak also refers to a metabolite candidate. These have to be evaluated using a database of identified metabolites or by an expert manually. Coupled to an electron ionization time-of-flight mass spectrometer (TOF-MS) for metabolite identification, GC×GC has been applied successfully to metabolic fingerprinting [36, 44, 35, 28]: the detection of changes among different conditions, like disease states or tissues.

Before comparing measurements, metabolite candidates within each chromatogram have to be identified and combined across chromatograms. The metabolite candidates and the number of metabolite candidates vary across measurements. Some may be present in the disease state, but not in the control sample. The presence of thousands of metabolite candidates per measurement calls for automatic solutions.

Different solutions have been suggested in recent years for the alignment and processing of GC×GC-TOF-MS spectra. Shellie *et al.* [57] directly compare chromatogram plots of mouse tissue extracts. First, all peaks in the chromatograms are represented by bubbles. Bubble plots of mutant as well as control mice are averaged. Further, the bubbles of the control mice plot are subtracted from the bubbles of the

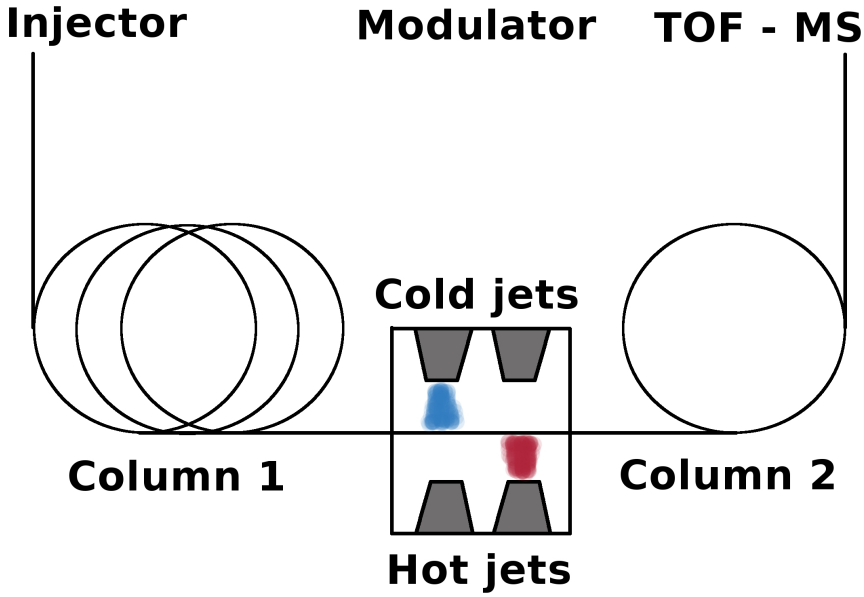


Figure 1.4: Schematic view of a two-dimensional gas chromatography time-of-flight mass spectrometer: The sample is injected, runs through the first column, is freezed and heated in small segments in the modulator before entering the second column (GC×GC), and finally gets ionized and is measured by the detector (TOF-MS).

mutant mice. Finally, the remaining peaks, which are identified as metabolites, are sorted in decreasing order of their bubble size. The parallel factor analysis (PARAFAC) algorithms [43, 6] use raw chromatographic data as input. Factor models search for consistent signals across samples for metabolite identification, alignment, and quantification of spectra. These complex models are computing-intensive.

Peaks across different measurements may be shifted because of a varying injection time or temperature. Therefore, Fraga *et al.* [24] and van Mispelaar *et al.* [62] propose algorithms to correct retention time variations in comprehensive two-dimensional separations. Both methods can only be applied to small regions of interest in two-dimensional data sets. A retention time correction of the entire chromatogram in both separation dimensions is described in Pierce *et al.* [52] and by Zhang *et al.* [73]. Pierce *et al.* adjust peak shifts using reference peaks in windows of first and second dimension retention times across samples. The correlation optimized warping (COW) algorithm of Zhang *et al.* corrects time shifts and combines measurements in one step by dividing the chromatogram plots into smaller squares. These are stretched and compressed such that standard peaks overlay across measurements. The overlap is

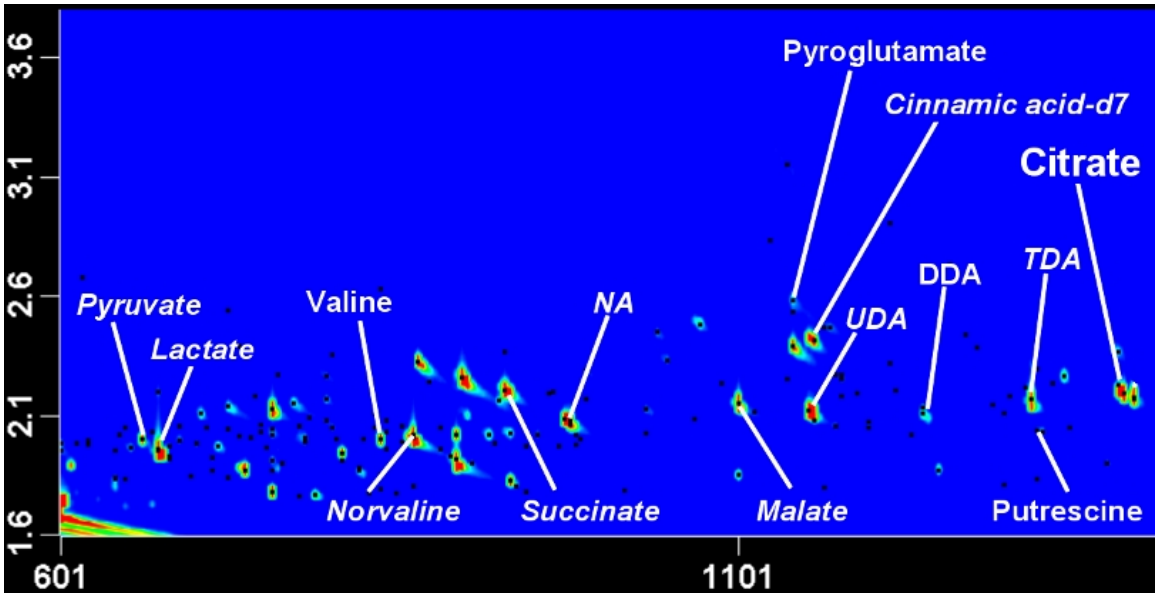


Figure 1.5: A two-dimensional chromatogram plot where the axes describe the retention times generated by the two columns. The color encodes the peak intensities with blue indicating no signal against background, and red high intensity. In this figure, the signal intensity is the total ion count of all fragment ions for a metabolite. Plotting a specific fragment ion is possible as well.

measured by correlation. Alternatively, Schmarr *et al.* [56] apply an image processing approach that is routinely used in the comparison of two-dimensional protein gels for the compensation of run-to-run variations and peak merging.

The methods mentioned so far align raw GC×GC-TOF-MS data based on two-dimensional retention times, ignoring the specific fragment ion mass spectrum of a metabolite. They ignore viable data preprocessing functions, provided by the instrument software - for instance by ChromaTOF, such as peak finding, peak integration, library search and signal-to-noise filtering. Oh *et al.* [49] developed the MSort software for GC×GC-TOF-MS data that uses data processing provided by the Leco ChromaTOF software to generate peak tables. The algorithm creates a reference table of metabolite candidates using first and second dimension retention times and the linear correlation of fragment mass spectra. Peaks not present in a large number of samples are excluded from alignment. Then, the reference peaks are searched and combined across all samples. Peaks not present in the reference table are lost. The calculation of all pairwise correlations requires a lot of computer memory. Wang *et al.* [65] propose a distance and spectrum correlation optimization (DISCO) al-

gorithm that uses the Euclidean distances of standardized two-dimensional retention times and the correlation of mass spectra for peak alignment. It works similar to my alignment algorithm [4] (INCA) developed one year before.
INCA sorts metabolite candidates into an alignment table according to tolerance parameters for retention times and mass spectra. The tolerance parameters may be set by the user or can be optimized by spiked-in metabolites. No metabolite candidates are lost during alignment. A detailed description of INCA alignment and applications is given in Chapter 3. In 2010, Leco provided their own commercial alignment tool Statistical Compare (SC). It is implemented in the Leco ChromaTOF software version 4. We compared SC to the in-house developed algorithm INCA (see Chapter 3).
Recently, Castillo *et al.* provided a open source data analysis platform called Guineu [11]. Guineu uses peak lists and performs data alignment based on retention times and mass spectral information.

The Statistical Compare alignment tool of Leco company The description of the Statistical Compare alignment tool is based on information received from Leco. Many details are missing and cannot be recovered.

In the Statistical Compare (SC) feature of ChromaTOF, peaks are aligned based on first (1st) and second (2nd) dimension retention times and mass spectra. Pairwise sample comparisons of all samples are made peak by peak. Peaks across samples will be grouped together if they are within a specified retention time window and share the best spectral match. The spectral match uses the match algorithm from the National Institute of Standards and Technology (NIST), the same algorithm that is used in library searches.

Following the pairwise comparisons, common peaks are linked from sample to sample to create groups of common analyte peaks. During the grouping conflicts may arise due to the parameters specified in the data processing method (retention time window and mass spectral match threshold) or variability in peak interferences and retention time shifts. The software takes various steps encoded in non-accessible source code to resolve these conflicts. Failure to resolve conflicts results in the exclusion of peaks from the final table.

To assign a name to an entry in the aligned peak table, the peak of each sample gets a peak weight. This weight is based on the peak shape of an extracted mass ion, which

is unique for that peak, and the quality of the found peak. The best quality peak within a group of aligned peaks is selected to conduct a library search for that group of peaks. The name of the matching library compound appears in the compound table.

Peak finding performed during data preprocessing may yield different unique masses for identical compounds in different samples. However, SC has to choose a mass for quantification from the unique masses. The quant mass is a mass that uniquely identifies a metabolite in a specified retention time window. This quant mass is the most common unique mass for the peak across all samples. If a different unique mass has been selected for an analyte in any of the samples, the peak profile of the specified quant mass is checked. If the peak profile is good, the quant mass is used. If the peak profile is poor, the profile of the unique mass is used and the ratio of the quant mass to the unique mass in the peak true spectrum is multiplied by the unique mass peak profile to produce the quant mass peak profile for calculating the peak height and peak area. In a summary, if the unique masses of a identical compound across samples differ, the peak area of the differing compound is interpolated.

Chapter 2

Review on classification probabilities

In this chapter I explain the concepts of classification algorithms (Section 2.1) and motivate classification probabilities (Section 2.2). Existing estimation methods for pairs of disease states in the context of computer-based learning are reviewed in Section 2.3. These probability estimation methods involve parameters, which need to be trained. Together with parameters of the classification procedure I discuss the whole training and validation framework in Section 2.4.

2.1 Classification of "omics" data

In supervised learning, feature selection, classification, and model selection are closely connected. A decision rule, also called the *classifier*, is inferred from training data where class membership is known. This rule is applied to new data to predict class membership. The classifier is based on a set of features. In metabolomics, these features are metabolite candidates, in transcriptomics gene expression profiles which are able to separate healthy patients from patients of known disease (see Figure 2.1). The metabolite candidates or the gene signature are choosen according to a quality criterion like the maximal classification accuracy. The classes are disease entities, such as leukemia subtypes [29], risk groups [63], treatment response [12] or disease outcome [67].

Both, metabolomics and transcriptomics datasets, comprise less measurements than measured features. Ion abundances of a metabolite or gene expression values are continuous variables. Hence, the learning concepts developed for transcriptomics data can be applied to metabolomics data. In microarray based classification, a variety

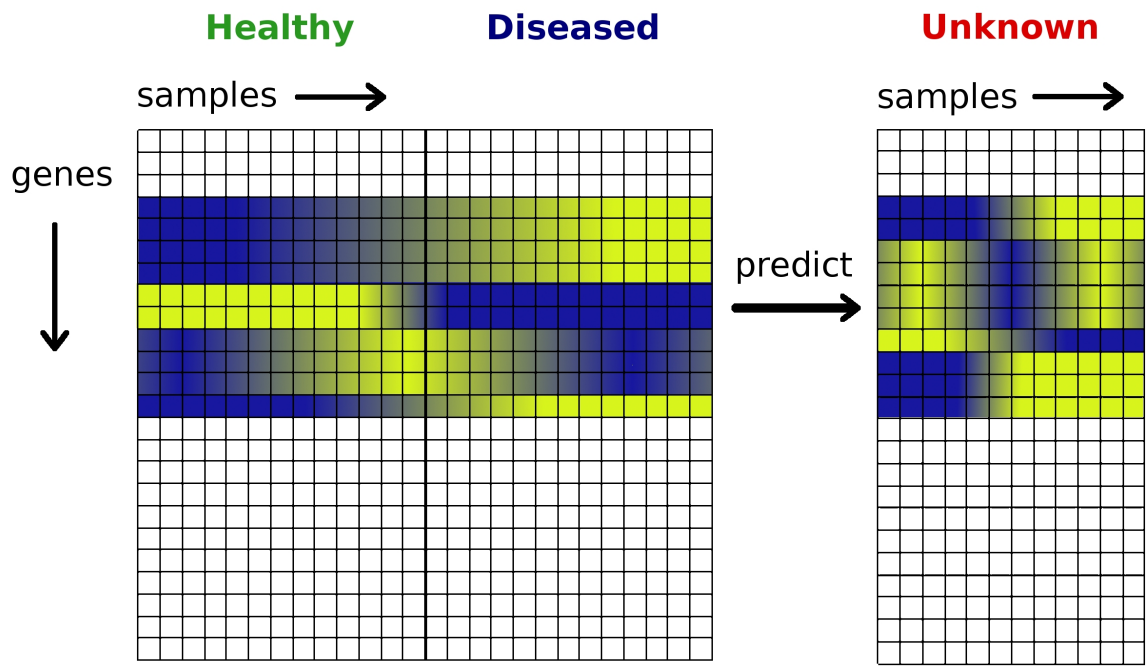


Figure 2.1: Healthy and diseased patients are used to learn a gene signature which is able to separate the two groups with small misclassification error. Transcriptomic data are arranged in a matrix where the rows correspond to genes and columns to patients. The gene signature is colored in blue and yellow indicating different levels of gene expression. The samples of unknown group label are predicted as healthy if their expression profile of the signature genes is similar to the expression profiles within the healthy group.

of classification algorithms have been proposed and critically compared [14, 23, 72]. Examples are neural networks, support vector machines, ridge regression, variants of linear discriminant analysis (LDA) such as Fisher's LDA or diagonal LDA, decision trees, and nearest neighbour approaches.

Typically, the discriminating genes are selected before learning a classifier (*filter* methods) or in combination with the classification algorithm (*wrapper* and *embedded* methods) [55, 72]. Filter methods rank genes using statistical measures or tests such as Fisher's ratio, t-statistics, χ^2-statistic, information gain, or Pearson's correlation. The top ranked genes with highest classification accuracy are selected as gene signature. Afterwards, a classifier predicts new patients based on the fixed set of signature genes. Wrapper methods directly employ a classifier for assessing the diagnostic information of genes. The genes are weighted: the better the ability to separate the healthy from the diseased patients, the higher the weight of the gene. Those with lowest weights are iteratively excluded (recursive feature elimination),

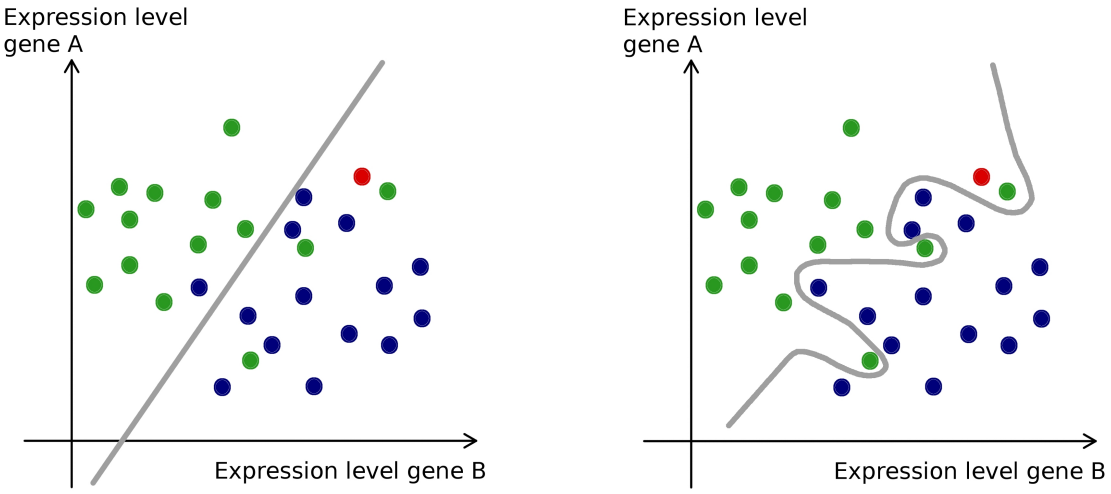


Figure 2.2: The gene expression levels of two genes span a two-dimensional space. Healthy (green dots) and diseased (blue dots) patients are separated by a gray line. On the left panel, patients are separated by a linear boundary with some misclassifications. The right panel shows a more complex boundary which does not make errors on the training data. The new patient (red dot) is predicted as healthy but is located next to the diseased patients. Intuitively, he should be predicted as diseased.

or those with highest weights are iteratively added to the gene signature (forward feature selection), as long as the classification accuracy increases. While weights are fixed for filter methods, they are iteratively adjusted for wrapper methods. While filter methods consider each gene separately, wrapper methods take interactions or coexpression of genes into account. Wrapper methods search heuristically for signature genes. Embedded methods are coupled to a classifier which selects the genes to infer the optimal decision boundary.

In general, classification methods can be distinguished in linear and non-linear models [40] (see Figure 2.2). Linear models base their prediction of a new patient on weighted genes. These weights define a boundary hyperplane between healthy and diseased patients of the training data. In Figure 2.2, the left panel shows the linear separation (gray line) of healthy (green dots) and diseased (blue dots) patients. The two-dimensional space is spanned by the gene expression levels of two genes. The two classes can not be separated without misclassifications. One diseased patient is located within the healthy group and some healthy patients within the diseased. The more complex boundary of a non-linar model on the right panel does not make any

errors, but the new patient (red dot) is predicted as healthy, although it is nearer to the diseased patients and intuitively, should be predicted as diseased.

The training data is a sample from the whole population of healthy and diseased patients. The classifier applies to the whole population. The complex model fits the training data but the misclassification rate on new patients nevertheless might be much higher than that of simpler models (see also Wessels *et. al* [66]). A misclassification rate for new patients can be estimated by putting a part of the training data aside as test set. The classifier is learned on the training set and applied to the test set offering a classification accuracy which can be used as an estimation whether a new patient is predicted correctly. If the training set is partitioned into k parts and each part is taken apart once as test set, this loop is called cross-validation. For more details on training and testing, the reader is referred to Section 2.4.

2.2 Why probabilities?

Diagnosis, prognosis and prediction of treatment response based on transcriptomic, proteomic or metabolomic profiles is a well developed field [58, 42]. It very much depends on the classification problem at hand, whether an almost error free classifier can be developed or whether some classification errors are unavoidable regardless of what algorithm is chosen.

In the latter case it is natural that a clinician asks for the reliability of an individual diagnosis before moving on to treatment decisions. Classification algorithms are typically evaluated by the frequency of misclassifications in cross-validation or on an independent test set. However, these performances are averages over many predicted cases. They tell little about the reliability of an individuals diagnosis. The case might be easier or more difficult to diagnose than the average in the test set. Reliability of individual classifications can be expressed in terms of classification probabilities. For each case, every class is assigned a value $p_j \in [0, 1]$. This value is an estimated probability that the case belongs to that class, given the profiling data available for the given case.

In microarray based classification, the performance of classification algorithms has been analyzed and compared in great detail [22, 66]. However, little attention has been given to the usefulness of probability estimates and this is even more true

for metabolomic analyses. In fact, only relatively few classification algorithms for genomic and metabolomic profiles estimate class probabilities and in the majority of clinical papers on the performance of classifiers such case specific probabilities are not shown.

One reason for the limited use of probability estimates in genomics might be that they are hard to validate. While classifications are either right or wrong, it is less clear whether a classification probability is correct or incorrect. The probability estimates typically rely on model assumptions that vary across classification models, which makes them difficult to compare. Nevertheless, one can ask the question: When is a classification probability useful? I argue that it is most useful, if it flags incorrect classifications as low confidence classifications. In other words: if a classifier produces confident class probabilities close to one, these should be correct classifications.

2.3 Probability estimators

In the next paragraphs, a selection of class probability estimators is reviewed including the estimator from the popular prediction analysis for microarrays (PAM) program [60] and methods more widely used in different application areas like text categorization and digit recognition.

Notations: Let x_{ij} be a data matrix with $i = 1, 2, \ldots, m$ denoting features (metabolites) and $j = 1, 2, \ldots, n$ denoting cases. Further let C_k be a vector storing the true class membership (disease types) of cases with $k \in 1, 2$.

The end product of all linear classification algorithms is a classification rule that assigns a case to class 1 if $s(x) < 0$ and otherwise to class 2, where $x = (x_1, x_2, \ldots, x_p)$ is the vector of intensity values of p signature features, $w = (w_1, w_2, \ldots, w_p)$ a vector of corresponding feature weights, and $s(x) =< w, x > -b$ with distance to the origin b. Note that the vector w spans a line orthogonal to the separating hyperplane, $< w, x >$ is the orthogonal projection of profile x onto this line, and $s(x)$ the distance of x to the hyperplane. Intuitively, cases that are closer to the separating hyperplane are less reliably classified than those that are further away.

Naive Bayes Estimates (NB) Tibshirani *et al.* [60] proposed a method for class prediction in DNA microarray studies based on nearest shrunken centroids. Each class k is represented by a shrunken centroid $\bar{x}_k$ and a case x is assigned to the class with the nearest centroid. By shrinking standard class centroids to the overall centroid less-informative genes are weighted down or filtered out, thus yielding sparse classifiers. Formally, the discriminant score

$$\delta_k(x) = \sum_{i=1}^{p} \frac{(x_i - \bar{x}_{ik})^2}{\sigma_i^2} - 2 \cdot log\pi_k$$

is calculated independently for each class, where the sum runs over all genes with non-zero weights after shrinkage Δ, σ_i is the pooled within-class standard deviation of gene i, and π_k the estimated proportion of cases from class k in the entire population. A case is then assigned to the group k with minimal $\delta_k(x)$. Classification probabilities are also derived from the $\delta_k(x)$ by assuming independence and normality with equal within-class variance of all p classifier genes. Under these assumptions Bayes' theorem yields for two class classification probabilities

$$p_k(x) = \frac{e^{-\frac{1}{2}\delta_k(x)}}{e^{-\frac{1}{2}\cdot\delta_1(x)} + e^{-\frac{1}{2}\cdot\delta_2(x)}}. \tag{2.1}$$

Note that the nearest shrunken centroid classification rule defines a separating hyperplane with normal vector $w_i = \frac{2\cdot(\bar{x}_{i2}-\bar{x}_{i1})}{\sigma_i^2}$, and equation 2.1 can be translated to

$$p_k(x) = \frac{1}{1 + e^{-\frac{1}{2}\cdot s(x)}}.$$

The proof of the transformation of Equation 2.1 can be found in Appendix A.1. In this approach every gene in the classifier is assumed to contribute independent evidence as to whether the case x belongs to class k or not. This assumption is mostly not justified biologically and produces artifacts that will be discussed in Chapter 4.

Compound Bayes Estimates (CB) Another type of class probability estimators that is used for microarray classification problems is the Compound Bayes estimator. It was introduced in Wright *et al.* [70] and was used for classifying diffuse large B-cell lymphoma (DLBCL) into two biologically and clinically distinct subgroups called ABC and GCB lymphomas, and a class of "unclassifiable" cases, which comprises all cases with borderline classification probabilities. It needs to be noted that the classes

ABC and GCB were defined from the gene expression data in an unsupervised analysis which puts the analysis outside of our supervised classification context. Nevertheless, the estimator can be applied in a supervised setup without modification. In line with the PAM approach, the CB estimator models both classes individually using normal distributions. However, unlike the PAM approach, the CB estimator models the one-dimensional distributions of the projected data $s(x_j)$ instead of the multi-dimensional distributions of the original data x_j. Given a separating hyperplane with normal vector w and associated classification scores $s(x_j) = \langle w, x_j \rangle - b$, one assumes that the $s(x_j)$ are distributed normally in both classes with possibly different means μ_k and standard deviations σ_k. Bayes rule now yields

$$p_k(x) = \frac{\phi_1(s(x); \hat{\mu}_1, \hat{\sigma}_1^2)}{\phi_1(s(x); \hat{\mu}_1, \hat{\sigma}_1^2) + \phi_2(s(x); \hat{\mu}_2, \hat{\sigma}_2^2)}$$

where $\phi(x; \mu, \sigma^2)$ represents the normal density function with mean μ and variance σ^2. The four parameters $\hat{\mu}_1, \hat{\mu}_2, \hat{\sigma}_1, \hat{\sigma}_2$ are estimated from the projected data $s(x_j)$.

Both methods above model distributions for both classes separately and invoke Bayes theorem to obtain class probabilities. A well established complementary approach is the direct estimation of class probabilities through binary regression. The approach has many ramifications some of which have been applied to genomic data [67, 18].

Binary Regression (BReg) In the evaluation of probability estimation functions, the class of binary regression models is represented by the approach described in Platt [53], which fits the logistic model

$$p_k(s(x)) = \frac{1}{1 + e^{A \cdot s(x) + B}}$$

by minimizing a cross-entropy error function to adjust the parameters A and B. Although Platt [53] uses this estimator in combination with linear and non-linear support vector machines, it can also be used together with our linear classifiers without changes, since the regression simply operates on a set of precalculated classification scores $s(x_j)$ without exploiting any properties implied by the method that generated these scores.

The three estimators described so far are parametric in that they assume that the data is generated according to certain parametrized families of distributions (normal

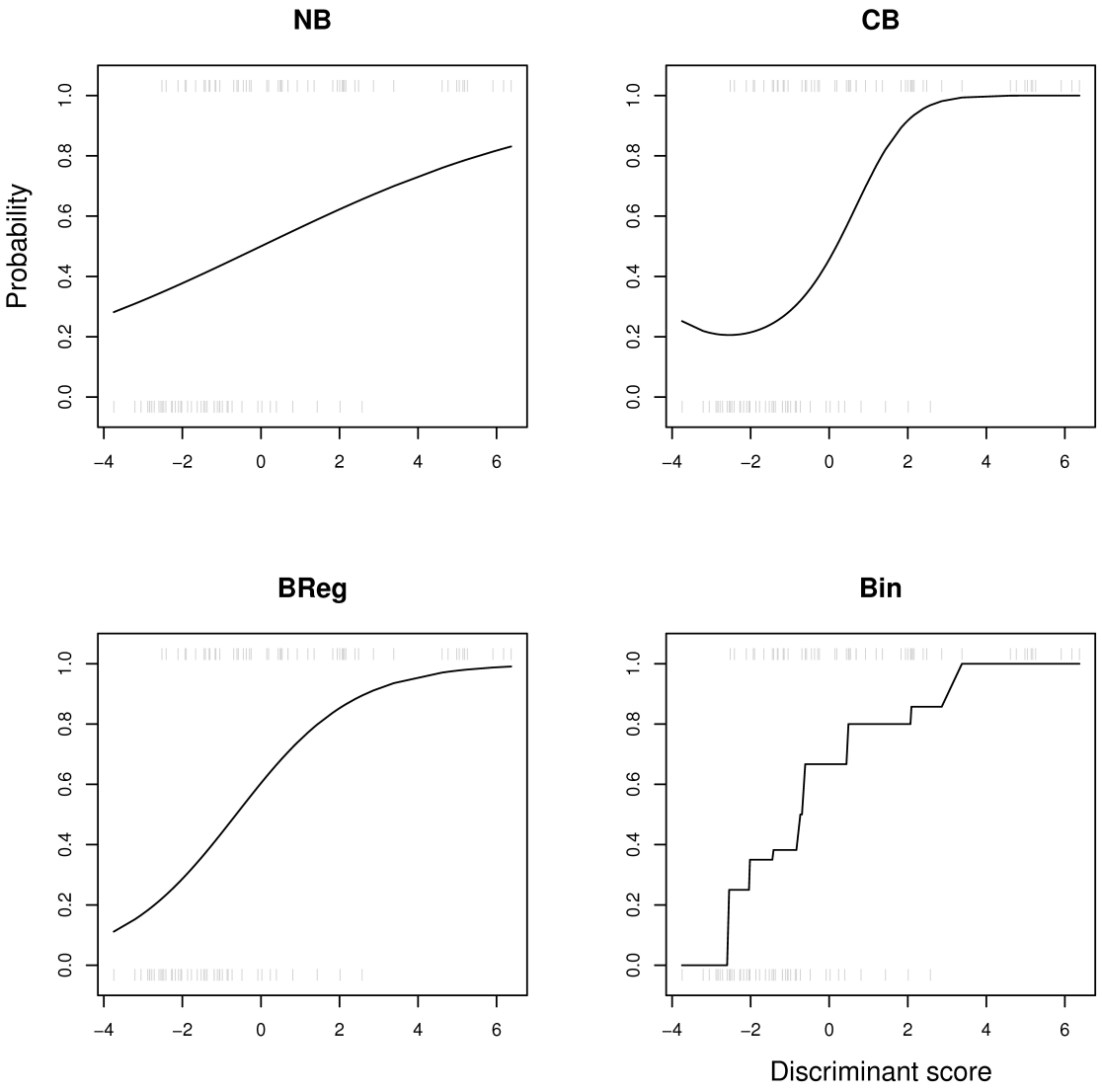


Figure 2.3: The Figure shows class probability functions (CPFs) for probability estimation methods Naive Bayes, Compound Bayes, Binary Regression and Local Error Frequencies using binning.

distributions) or in that the regression function is known (logistic) up to some adjustable parameters. In the next paragraph, a non-parametric estimator based on monotone regression will be described.

Local Error Frequencies (LEF(Bin)) If the shape of the regression function that relates classification scores to class probabilities is unknown, it can be estimated from local misclassification frequencies. Zadrozny *et al.* [71] sort cases by the scores $s(x_j)$

and split them into equally sized disjoint bins. The local class k frequency $F_k(x)$ of case x is then calculated as the relative frequency of class k cases that fall into the same bin as x. The local error frequencies can be interpreted as the classification performance of the algorithm given the distance to the separating hyperplane. Naturally it becomes smaller in bins closer to the decision boundary. However, the estimates $F_k(x_j)$ do not need to be strictly monotonous in $s(x_j)$. Hence, and perhaps counter-intuitively, in a few cases that are closer to the separating hyperplane might be judged more reliably classified than some of those that are further away from it.

To assure monotonicity of estimated probabilities with increasing score Zadrozny *et al.* [71] use monotone regression as implemented in the pair-adjacent violators algorithm (PAVA) [8]. PAVA yields the maximum likelihood estimates of the desired probabilities by replacing $F_k(x_j)$ and $F_k(x_{j+1})$ with their average when the monotonicity constraint is violated. This averaging process is continued until an ordered set of probabilities is obtained. As with previous methods it needs to be noted that the original work by Zadrozny *et al.* [71] used the estimator in combination with a different classification algorithm (decision trees). However, since this method is also based only on post processing classification scores, it can be used in the linear classifier context as well. Examples for all class probability functions (CPFs) are shown in Figure 2.3.

2.4 Training and validating class probability functions

In the previous section, procedures that estimate classification probabilities $p_k(x_j)$ from classification scores $s(x_j)$ were reviewed. The end product is always an estimated class probability function (CPF) $p_k(s(x))$ that maps the score $s(x)$ to a number between 0 and 1 that is interpreted as the probability that case x is in class k. The CPF needs to be trained on sets of classification scores.

Training includes the estimation of distribution parameters like the class means and variances in the Compound Bayes Estimator, the regression parameters A and B of the binary regression approach, or the local error frequencies $F_k(x_j)$ and the non-parametric regression curves resulting from monotone regression. In addition, training might depend on tuning parameters like the number of bins or the bandwidth of a gaussian kernel, or the number of nearest neighbors in the adaptive LEF approach (see Chapter 4).

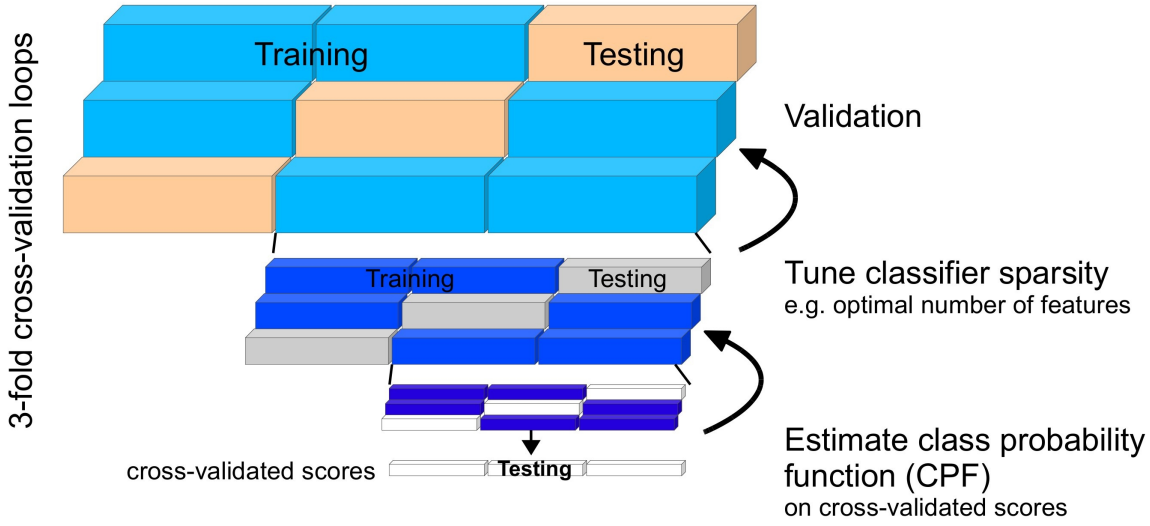


Figure 2.4: The Figure schematically shows a 3-fold nested cross-validation. In the innermost loop the class probability function CPF is optimized, in the next loop classifier sparsity is tuned and in the outermost loop validation is performed. The different loops are marked by the size of the corresponding bars. Note that in the Figure each loop is further separated into different rows to indicate that data present in that loop are split several times in training and test data to ensure that all data of a specific loop are used once for testing. The outermost loop contains all data, of this the training data of the outermost loop are passed to the middle loop where the passed data are split again in training and test data and the training data of the middle loop are transfered to the innermost loop where again a splitting in training and test data is performed.

Once a CPF is estimated, it can be used to predict classification probabilities of test cases x'_j by first calculating the scores $s(x'_j)$ using a linear classifier and then plugging them into a CPF to gain the class probability $p_k(s(x'_j))$. This requires a CPF and a classifier and both need to be previously learned from data. It is important that the test cases x'_j were not included in any of these learning processes.

With respect to CPF estimation it is important to distinguish between training scores $s(x_j)$ and test scores $s(x'_j)$, since it is known that their distributions can be greatly different [5]. Compared to test scores, training scores display a better but unrealistic separation of classes. This overfitting phenomenon can greatly affect the estimated CPFs as will be shown in Section 4.3. As an example, here a 3-fold nested cross-validation is used that covers the processes of classifier estimation, CPF estimation, parameter tuning, and evaluation. Parameters that need to be calibrated include the tuning parameters of the local error frequency approach, which is called Θ and the shrinkage parameter Δ of the nearest shrunken centroid classification algorithm that

controls the sparsity of the classifier.

1. **CPF Estimation**
   In the inner-most loop the shrinkage parameter Δ is fixed. N_1 cases are left out and the remaining cases are used to learn a classifier, which is applied to the left out cases yielding scores $s_\Delta(x_j)$. By leaving out all cases in turn, cross-validated scores are achieved for all cases that entered the innermost cross-validation loop. Note that these scores result from different classification rules (learned hyperplanes) for different bins of left out cases. A CPF is estimated from these scores for a variety of values of the parameter Θ. For each value the $p_k^{\Delta,\Theta}(x_j)$ are computed using one of the methods described above. They are evaluated with respect to their classification performance by calculating the negative log-likelihood of true classes

$$-\log(L(\Theta)) = -\sum_k \sum_{j \in C_k} \log p_k^{\Delta,\Theta}(x_j) \tag{2.2}$$

   The Θ with minimal $-\log(L)$ is chosen and the corresponding CPF $p_k^\Delta(\cdot)$ is returned to the middle cross-validation loop.

2. **Tuning Classifier Sparsity**
   In the middle loop, N_2 cases are left out. The remaining cases are forwarded to the inner loop varying Δ. For every Δ, the inner loop returns a CPF $p_k^\Delta(\cdot)$ which is applied to the left out cases of the middle loop. These are evaluated by their misclassification rate and the optimal value of Δ and with it the optimal number of features is determined. The optimized CPF $p_k(\cdot)$ is returned to the outer loop.

3. **Validation**
   In the outer loop N_3 cases are left out. The remaining cases are forwarded to the middle loop, which returns a CPF $p_k(\cdot)$ to be applied to the left out cases. Finally, this leaves a set of cross-validated probabilities $p_k(x_j)$ which will be evaluated with respect to different criteria in Section 4.3.

Note that both the inner loop and the middle loop include internal loops that vary the parameters Θ and Δ. However, these are not cross-validation loops, since they do not involve leaving out additional cases. Clearly only the non-parametric estimators based on local error frequencies include tuning parameters in CPF estimation. For

all other methods there is no internal loop in the inner cross-validation. Moreover, our design allows different CPF estimation procedures to use different numbers of features for classification. This is enabled by the middle cross-validation loop. This is important since some CPF estimators, for instance the PAM estimator, are very sensitive to the number of features. The entire cross-validation design is summarized in Figure 2.4.

Chapter 3

Processing 2-dimensional gas chromatography time-of-flight mass spectrometry data

In this chapter the preprocessing steps of two-dimensional gas chromatography time-of-flight mass spectrometry (GC×GC-TOF-MS) data are described (Section 3.1). This technology lacks in software to compare different measurements. Hence, we developed an alignment algorithm (INCA) which combines measurements into one data matrix (Section 3.2). The tool is validated by an spike-in experiment and applied to an E. coli dataset and a serum versus plasma sample collection. Recently, an alignment algorithm has been integrated into the firm software package for the chromatograph (manufacturer Leco Corp.). We compare INCA to the new software tool (see Section 3.3).

INCA and the comparison to the new software tool was performed in cooperation with Martin F. Almstetter (Institute of Functional Genomics, Head Prof. Peter Oefner, University of Regensburg). In his thesis, he focused on the experimental work related to the validation and biomedical application of INCA whereas I focused on the bioinformatics work and, in particular, the development and validation of the algorithm. The equal contributions to this body of research are documented in two joint first authorships [4, 3].

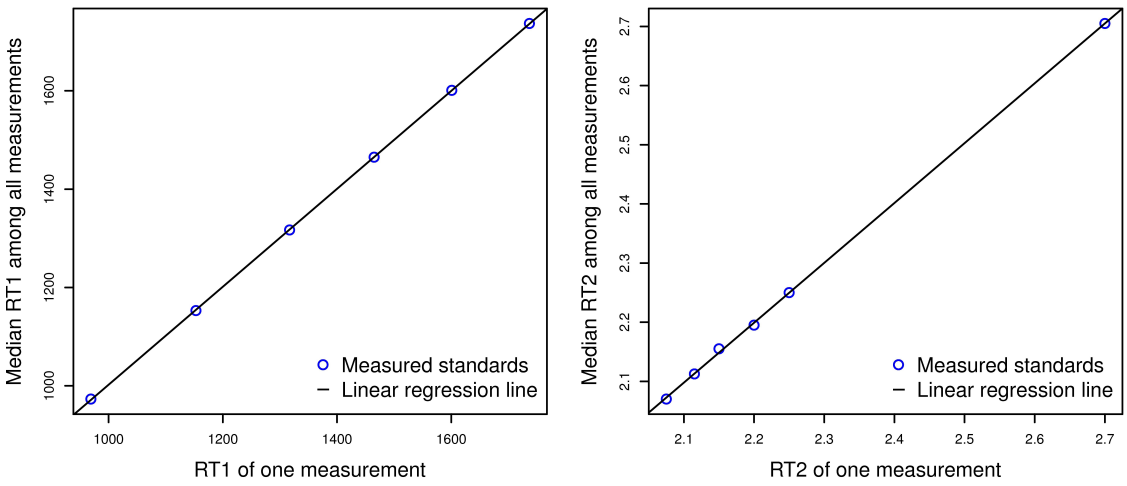


Figure 3.1: Linear models of first and second dimension retention times for a GC×GC-TOF-MS measurement based on six odd-numbered fatty acids. A linear model consists of a scaling factor and an offset which best correlate the observed retention times for the standards in the corresponding measurement to the median across all measurements.

3.1 Data preprocessing

Raw data output from the mass spectrometer is processed with the Leco ChromaTOF software. The validation and applications of INCA alignment is based on software version 3.34 [4, 19], the comparison to the commercial software tool on version 4.32 [3]. Detailed parameter adjustments before processing can be found in Almstetter and Appel *et al.* [4, 3] and are examined and optimized by Martin Almstetter. The result of preprocessing is a peak list in tab-delimited format for each chromatogram containing peak names or Unknown, first-dimension retention times, second-dimension retention times, peak areas for the mass trace m/z 73 (m/z: mass to charge ratio), and metabolite candidate mass spectra containing only fragment ions above a certain intensity to reduce noise. The peak area of the mass trace m/z 73 that corresponds to the trimethylsilyl cation is choosen because it is formed for each trimethylsilyl derivative upon electron ionization. The signal to noise threshold is set to 500.

There are two types of technical variance in a measurement: variation in peak intensity and retention time.

The second type of variance influences the alignment process. Hence, it is adjusted beforehand by two independent linear models for the first and second dimension retention time. The models are fitted on odd-numbered fatty acids included as standards in each measurement. The fatty acids are well separated peaks that occur uniformly over the entire separation range except for the early part of the chromatogram. There, smaller odd-numbered fatty acids are not included as standards because of their natural occurrence in the biological matrix, here *E. coli*. A linear model consists of a scaling factor and an offset for each measurement which best correlate the observed retention times for the standards to the median across all measurements (see Figure 3.1). Multiplying by the scaling factor and adding the offset to the retention times of all peaks compensate shifts.

The first type of technical variance does not influence the alignment algorithm and is corrected afterwards. Small variations in peak intensity, for example caused by slight changes in the amount of sample injected, are adjusted by an internal standard, tridecanoic acid, present in all samples. The areas of all peaks are divided by the area integral of the m/z 271 trace of tridecanoic acid for each measurement. Zero values, which are introduced during alignment, are set to the minimum peak area prior to log-transformation. Zero values result from (i) signals falling below the predefined threshold of S/N 500, (ii) features not matching the tolerance parameters, (iii) improper deconvolution of mass spectral signals due to signal saturation, and (iv) failure of the ChromaTOF software to place peak markers.

3.2 Combining different measurements

After correcting for the main types of technical variance the retention time of the same metabolite still varies across measurements. The preprocessing by the software also includes sources of variance. The finding of metabolite peaks is not trivial because of overlaying metabolite mass spectra. Thus, deconvoluted mass spectra slightly vary in the number of fragment ions detected and relative signal intensities.

The biological and technical variance is taken into account by four tolerance parameters: one for each retention time, an overlap of fragment ions between mass spectra and a tolerance for their relative intensities. The alignment algorithm starts with an empty alignment matrix. The aligned metabolite candidates are arranged in rows, their characteristics in the columns. Characteristics are intervals of retention times, a

```
Initialize threshold parameter t_1st, t_2nd, t_ol, t_rate
Aligned <- matrix() # Initiate empty alignment matrix with
                    # columns 'Name', 'rt1st_lower_bound',
                    # 'rt1st_upper_bound', 'rt2nd_lower_bound',
                    # 'rt2nd_upper_bound', 'spectrum S',
                    # areas for files, 'mean_rt1st', 'mean_rt2nd'
Peaks <- matrix(measurements) # Read raw data with columns 'Name',
                    # 'rt1st', 'rt2nd', 'spectrum S', 'area'
for each peak in Peaks:
  if Aligned is EMPTY: generate new entry using peak; next peak
  rt1st <- which(abs(Aligned$mean_rt1st - peak$rt1st) <= t_1st)
  if any rt1st:
    rt2nd <-
      which(abs(Aligned$mean_rt2nd[rt1st] - peak$rt2nd) <= t_2nd)
    if any rt2nd:
      for each S in Aligned$S[rt1st[rt2nd]]:
        is <- duplicated fragment ions of S$ions and peak$ions
        smallerS <- min(len(S), len(peak$S))
        if (len(is) >= smallerS * t_ol:
          h <- abs(peak$S$h[is] - S$h[is])
          if max(h) <= t_rate: # align peak to Aligned entry
            update retention time boundaries and means
            update S as union set of fragment ions
              and renormalize heights
            remember area at specified file position
            next peak
  else: generate new entry using peak
```

Figure 3.2: Pseudocode of alignment algorithm. Parameters and associated tolerances are t_1st and t_2nd which correspond to first and second dimension retention times, while t_ol and t_rate correspond to relative overlaps of fragment ions and their relative signal intensities for a given feature.

mass spectrum consisting of fragment ions, and an area detected for m/z 73 for each measurement. The fragment ions of the mass spectrum are scaled so that the highest abundance is 999. The default value for the areas detected for m/z 73 is zero.

To fill in the alignment matrix, the peak lists of all measurements are read into a raw peak list. This list is sorted according to first followed by second dimension retention times. Peaks are added one by one to the alignment matrix starting at the top of the raw list. A raw peak is aligned to a metabolite candidate in the alignment matrix if all tolerance parameters are satisfied. Otherwise a new candidate containing the peaks retention times and spectrum is generated. A retention time is close enough

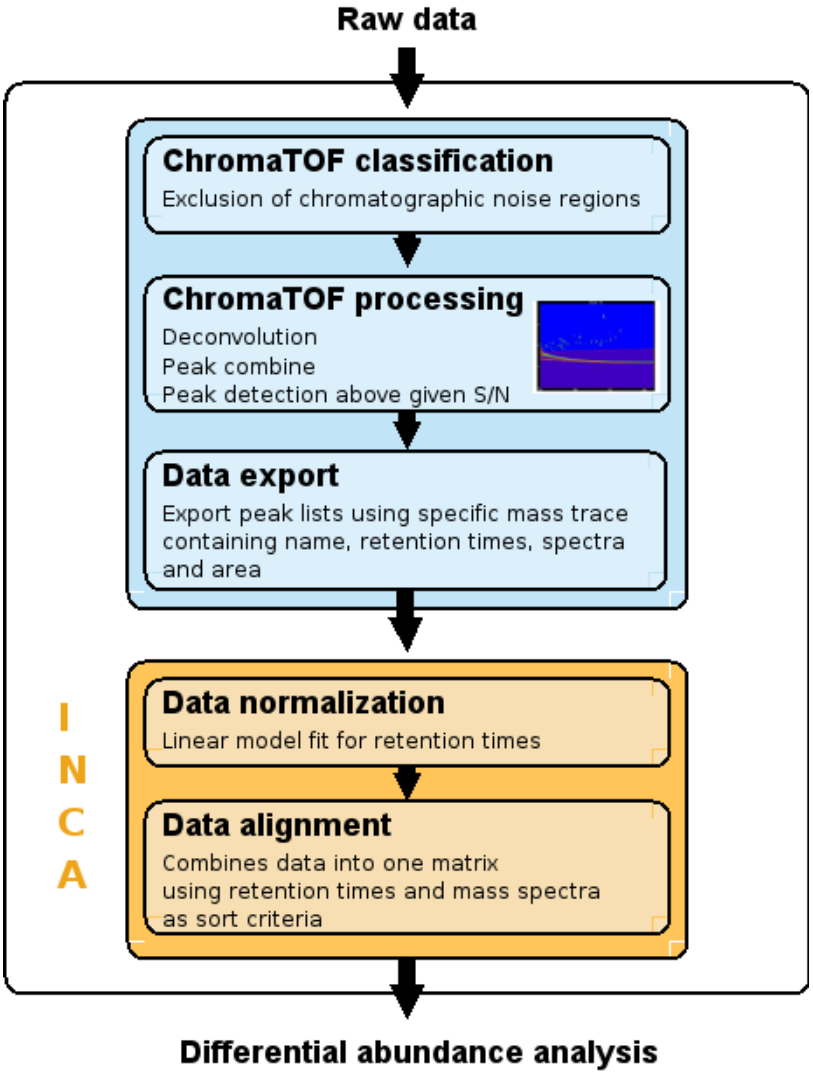


Figure 3.3: Flowchart for automated preprocessing of GC×GC-TOF-MS data.

if its distance from the entrys corresponding mean of interval boundaries is smaller than the corresponding tolerance parameter.

Two spectra are defined as similar if their fragment ions have a sufficient overlap and the difference in ion abundances of the overlapping ions is small. Then, a new union set of fragment ions and respective relative abundances is generated. Finally, the appropriate area detected for m/z 73 is stored in the aligned data matrix. The tolerance parameters are choosen by the user or automatically adjusted using a standard mixture added to each sample before measurement (see Section 3.3). The peak alignment module is implemented in R and accessed via the shell. The pseudo code can be

found in Figure 3.2. The combined procedure of retention time correction and data alignment is named INCA, integrative normalization and comparative analysis. A flowchart including the automated preprocessing of GC×GC-TOF-MS data is shown in Figure 3.3.

3.3 Validating the algorithm by a spike-in experiment

Spike-in experiment A mixture of 20 metabolites is added to the extract of an *E. coli* wildtype strain where the spiked-in metabolites do not naturally occur. These are 2-hydroxybutyrate, 3-hydroxybutyrate, 2-hydroxy-3-methylbutyrate, 3-methyl-2-oxovalerate, malonate, nicotinate, phenylacetate, dimethylsuccinate, decanoate, mandelate, adipate, erythritol, phenyllactate, triethanolamine, dodecanoate, suberate, xylitol, vannilate, mannitol, and eicosanoate. Seven spike levels are generated with 0.25, 0.275, 0.3125, 0.375, 0.5, 0.625, ad 1.0 nmol absolute of each analyte representing fold changes 1.1, 1.25, 1.5, 2.0, 2.5 and 4.0 by comparing all spike levels to the base level 0.25. Each sample is prepared in six replicates and samples are measured in random order. For correcting technical variance across measurements an internal standard solution containing odd-numbered, saturated straight chain fatty acids (C_9 - C_{19}) is added to each sample.

Measuring failed for two samples of the highest spike level. These two are excluded and the analysis is done based on 40 spike-in samples. The spike-in dataset is preprocessed as described in Section 3.1. The peak list of a sample contain 500 to 1000 peaks.

Optimizing the tolerance parameters The tolerance parameters are set such that the 20 spike-in metabolites are optimally aligned: the spike-ins are found and combined across all measurements, and no false metabolite peaks are added to spike-in entries in the alignment matrix. Due to a Leco software update in 2010, data preprocessing and thus the optimization process is run for the validation of INCA [4] (software version 3.34) and the comparison to the commercial software tool [3] (software version 4.32).

After preprocessing, the spike-in dataset is aligned using INCA for 288 different parameter combinations (Table 3.1). In Almstetter and Appel *et al.* [4], sixteen parameter settings are able to align the 20 standard compounds in a single row each, yielding 800 out of 800 true positives. The optimal parameter setting is 8 seconds and 0.1 seconds for the first and second dimension retention time, and 90% and 40% overlap of m/z values and relative ion intensities.

Parameter	Tolerance
t_1st [s]	4, 8, 12
t_2nd [s]	0.025, 0.05, 0.075, 0.1, 0.125, 0.15
t_ol	1.0, 0.9, 0.8, 0.7
t_rate	0.1, 0.2, 0.3, 0.4

Table 3.1: Parameters and associated tolerances tested for feature alignment. t_1st and t_2nd are tolerances for the first and second dimension retention time measured in seconds, t_ol for the overlap of fragment ions and t_rate for the ion intensities.

For the optimal parameter setting the alignment matrix has 4726 metabolite candidate entries. Many entries have only one or few non-zero values for the m/z 73 area. These are assumed to be false peaks or noise. Thus, candidates detected in at least 50% of all samples are selected for further analysis. This peak reduction results in 517 peaks with 17% zero values. These peaks are assumed to be true metabolites. This is highly compatible with multivariate analysis according to Gika *et al.* [26], who suggested that up to 50% zero values are tolerated. Then, the peak areas are normalized using tridecanoic acid and log transformed.

Furthermore, the spike-in dataset is used to check whether GC×GC-TOF-MS is able to detect fold changes. When comparing the samples of two spike-in levels, only the spike-in metabolites should vary. Differentially abundant metabolites are identified using t-statistics assuming equal variance in both spike-in levels. T-statistics are generated using the R Bioconductor package MULTTEST [54]. Sorting the list of metabolites in decreasing order according to the absolute value of the t-statistic put differential metabolites on top of the list. The spike-in metabolites should be found on top and are counted as true positives. All other metabolites should be true negatives. For each rank r of the sorted t-statistic, the sensitivity (true positive rate (TPR)) and 1 - specificity (false positive rate (FPR)) are calculated by comparing the metabolites

down to rank r to the list of spike-ins. The efficiency across all ranks is illustrated in receiver operating characteristic (ROC) curves. The false positive rate is plotted against the true positive rate (Figure 3.4). The area under curve (AUC) is at most 1. If it is one the spike-in metabolites are found before all other metabolites.

GC×GC-TOF-MS is not able to find a 1.1-fold change. An AUC of 0.5 indicate that the spike-in metabolites are distributed randomly among all metabolites. Fold-changes 2.0 and 4.0 yield high true positive rates of 92% and 96% for a small number of false positives. This matches sensitivities reported for gene expression data [38].

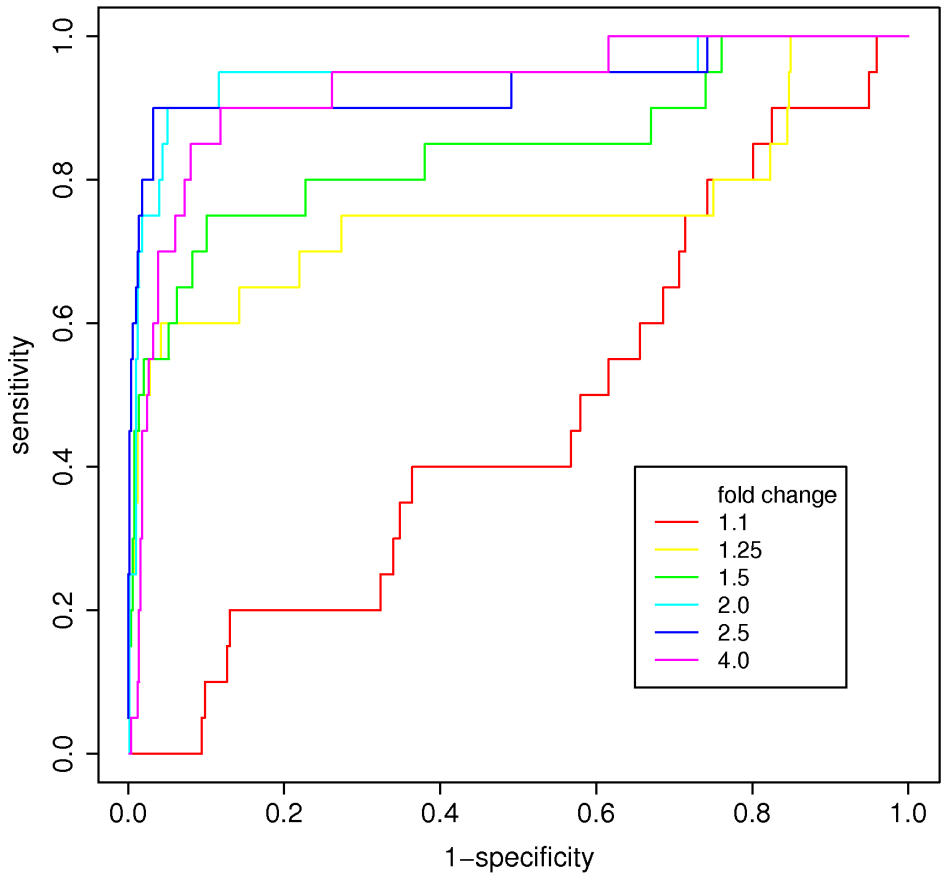


Figure 3.4: ROC curves for different fold changes (FC). True positive rate (TPR) and false positive rate (FPR) improve with increasing fold changes.

In Almstetter and Appel *et al.* [3], the optimal parameter setting is 4 and 0.05 seconds

for the first and second dimension retention time, and 80% and 20% overlap of m/z values and relative ion intensities. The alignment matrix has 4404 entries. After data reduction (candidates detected in at least 50% of all samples) 447 metabolite candidates remain with 17% zero values.

Compare the peak alignment algorithm to a recently developed software tool
The alignment algorithm *INCA* and the commercial software tool *Statistical Compare* (SC) are validated by the spike-in experiment as described in Section 3.3. The raw data are preprocessed with software version 4.32. The same preprocessing steps are chosen for SC and INCA (see Section 3.1). Peak alignment is carried out using first and second dimension retention times and mass spectra as sort criteria.

The output of SC is transformed to sustain a suitable data matrix for statistical analysis. The matrix stores the metabolite candidates in the rows and the characteristics in the columns. An entry is characterized by a peak name, an average and interval of first and second dimension retention time, an area count, and the m/z 73 peak area integrals for each measurement. SC alignment generates 887 metabolite candidiates. Candidates detected in at least 50% of all samples are selected for further analysis. This peak reduction results in 458 metabolites with a maximum of 14% zero values. Then, the peak areas are normalized with tridecanoic acid and log transformed.

For the INCA alignment, optimal tolerance parameters are determined using the 20 standard compounds that had been spiked into an *E. coli* wild type extract. The optimal parameter setting (see Section 3.3) results in 4404 metabolite candidates before and 447 metabolites after data reduction. The original number of metabolite candidates of INCA is much higher (4404) compared to SC (887), but after filtering, INCA yield a lower number of metabolites (447) than SC (458). INCA aligns all peaks from the raw peak lists, and thus guarantees that no data are lost. SC excludes peaks if conflicts in the peak grouping occur. For example, if a metabolite occurs only in one of two possible pathological conditions this metabolite also will be excluded. The higher number of metabolites of SC after data reduction is explained by the ability of SC to access the raw data after alignment to verify quant masses and automatically assign peak names. The areas for the m/z 73 mass trace of INCA and SC are manually compared to reconstruct the correct alignment of each spike-in metabolite across all samples. Only few areas of SC do not match the areas of INCA.

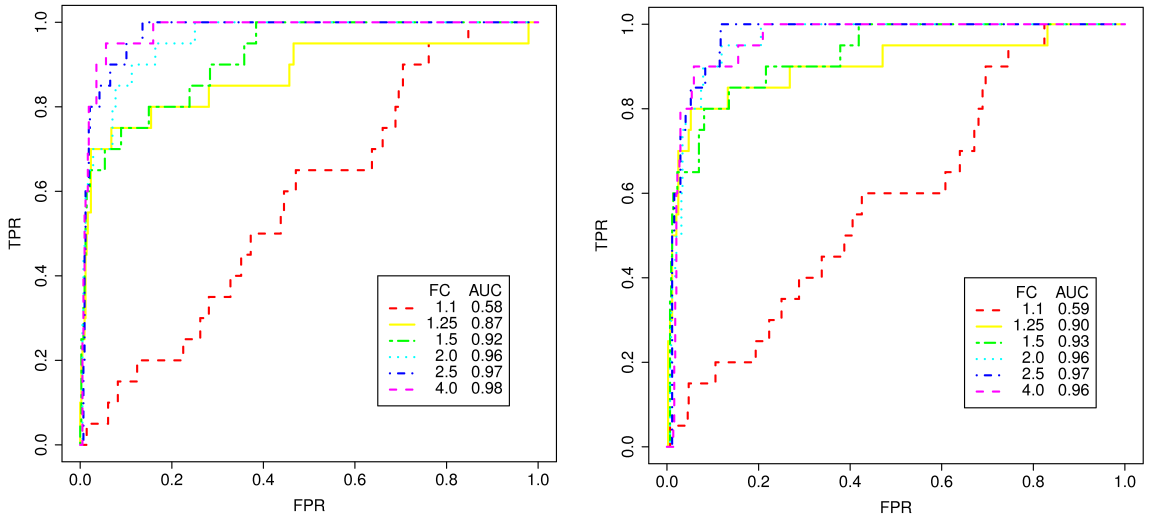


Figure 3.5: ROC curves after INCA (left) and SC (right) alignment for different fold changes (FC) and corresponding AUC values. TPR and FPR improve with increasing fold changes.

The efficiency for detecting fold changes is illustrated in receiver operator characteristic (ROC) curves. ROC curves for INCA and SC are shown in Figures 3.4 and 3.5. Both alignment algorithms can not detect a 1.1-fold change. The areas under the curve (AUC) for INCA and SC are 0.58 and 0.59. A 1.25-fold change is found by Statistical Compare (AUC of 0.90) more effectively than by INCA (AUC of 0.87). The AUCs for the 2.5- and 4-fold change are similar. But even the 4-fold change still yield false positives prior to reaching 100% true positives. In Figure 3.6, the AUCs for all pairwise fold changes of SC are plotted against the AUCs of INCA. Overall, SC outperforms INCA in the spike-in experiment.

True changes in metabolite concentration have to be detected against a background of thousands of spectral signals. Hence, we investigated whether the spike-in fold changes can be reproduced quantitatively. SC capitalizes on the great advantage of constantly assigning the same abundant unique mass trace to an identical compound in different samples. It was not possible to exploit a compound-specific unique mass for the metabolic fingerprinting with INCA, because we have no access to the raw data after alignment. Thus, INCA has to rely on the characteristic fragmentation behavior of trimethylsilyl derivatives upon EI ionization by using only the area integrals for m/z 73 as a quantitative measure for all peaks.

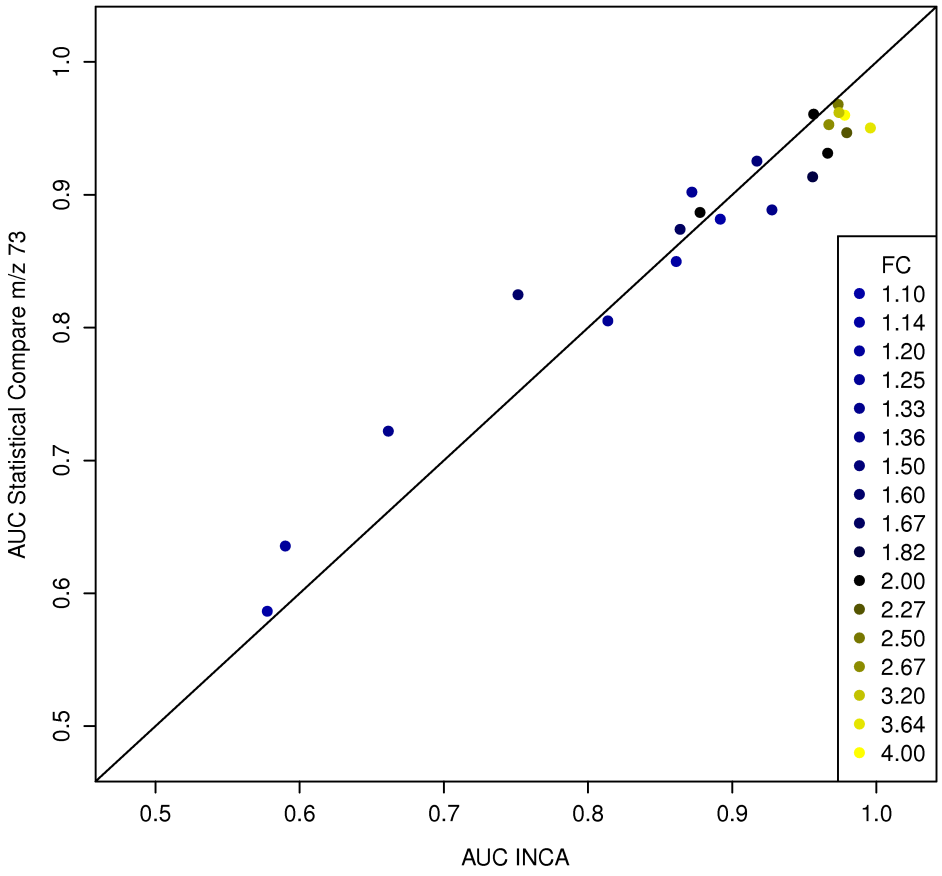


Figure 3.6: Area under the curve (AUC) comparison plot between Statistical Compare and INCA for all pairwise fold changes.

Between expected and observed fold changes of the m/z 73 linear dependencies are observed. An example, triethanolamine, is shown in Figure 3.7. Corresponding figures for the other spike-ins can be found in the Appendix A.2. Triethanolamine quantified by the integral of the m/z 262 (SC) instead of the m/z 73 trace (INCA) shows a nearly ideal regression line with a slope of 1 and an offset of 0. For all 20 spiked-in metabolites the correlation between observed and expected fold changes increases when integrating the unique mass trace extracted by SC rather than the m/z 73 fragment ion trace used by INCA. The ranges of regression coefficients improve from 0.827 - 0.992 to 0.882 - 0.994, the ranges of RSDs from 5.8 - 20.7 % to 4.2 - 20.7 %.

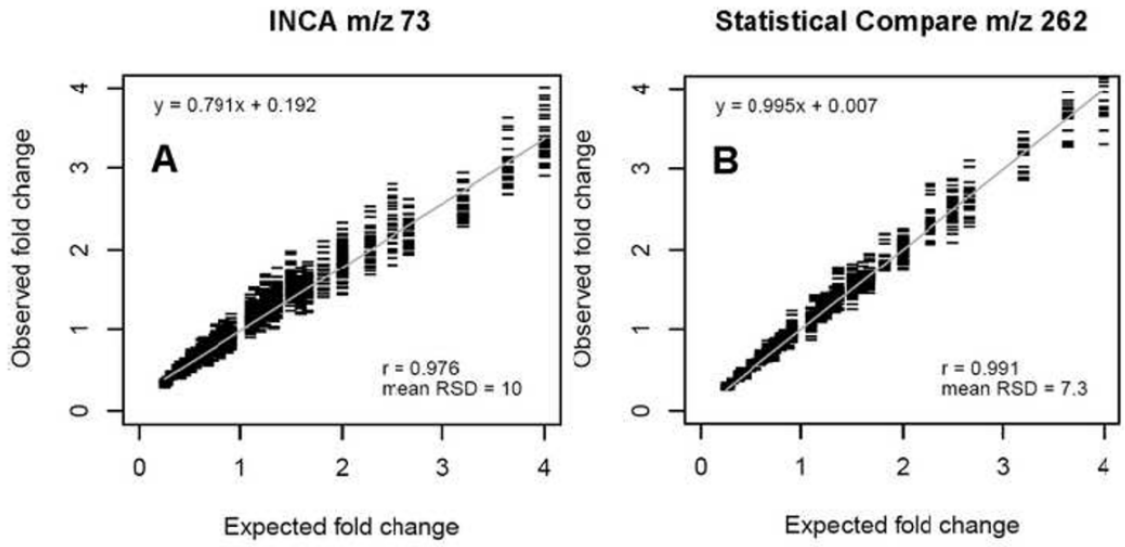


Figure 3.7: Linear dependency between expected and observed fold changes of triethanolamine for both (A) the universal m/z 73 fragment ion trace used by INCA and the analyte specific fragment ion trace m/z 262 extracted by Statistical Compare. All possible pairwise fold changes are plotted.

3.4 Applications

Comparing an E. coli wild type with a mutant strain The metabolic fingerprints of the *E. coli* wild type MG1655 and the *E. coli* MG1655 double-mutant ΔUdhA-ΔPntAB are compared by cultivating each strain in three separate flasks and by filtrating each culture in triplicates. Nine samples per strain are measured in random order to avoid a systematic error using GC×GC-TOF-MS.

The data are preprocessed, and combined by INCA using the tolerance parameters optimized by the spike-in experiment. After alignment, 2259 metabolite candidates remain. Candidates detected in at least 50% of all samples (9 out of 18) are further analyzed. This cutoff is selected with respect to the group size of 9 to ensure that metabolites not present in one group but in the other are not excluded from the list. After filtering, 25% zero values are among the 398 metabolites. A t-test with equal variances identifies a list of 48 metabolites that are likely to differ significantly between the two strains with an estimated false discovery rate of <0.05. In other words, less than 3 false positives are expected among the 48 identified metabolites. Among the 48 peaks, 27 true metabolites are identified manually (see Figure 3.8). The

double mutant strain has a reduced activity of the glycolytic and pentose phosphate pathways. Biologically, the 27 metabolites make sense since they are all part of these pathways.

Differential abundant metabolites do not imply that the strains can be correctly classified into wild type and mutant through their metabolite fingerprints. Thus, metabolic signatures are learned using the shrunken centroid classifier by Tibshirani *et al.* [60]. In a leave-one-out cross-validation procedure, 100% of the metabolic fingerprints are classified correctly. Most of the compounds are intermediates of the citrate cycle as expected from previous analysis of the two *E. coli* strains employing CE-TOF-MS [61]. The latter method confirmed that the additional differential abundant metabolites discovered by GC×GC-TOF-MS are true positives.

Comparison of serum versus plasma collection in GC-TOF-MS Bovine serum, EDTA-plasma and EDTA-plasma fortified with acetylsalicylic acid (ASA) as antioxidant are compared with regard to their suitability for metabolomic studies. INCA was adjusted to align based on one retention time and the tolerances for the mass spectrum. Metabolic fingerprints are generated from GC-TOF-MS data using the Leco Chroma- TOF software in combination with the in-house retention time correction and data alignment tool INCA. A total of 6, 9 and 21 differentially abundant metabolites with a false discovery rate of <0.05 are identified upon comparing EDTA- versus EDTA-ASA-plasma, EDTA-plasma versus serum and EDTA-ASA-plasma versus serum, respectively. To confirm that the observed signal intensities in the GC-TOF-MS fingerprints reflect true metabolite abundances, 19 amino acids, glucose and 6 organic acids are quantified by means of GC-MS using stable-isotope-labeled internal standards. As observed with the fingerprints, only the concentrations of lactate and citrate are found to be significantly lower in EDTA-plasma and serum, respectively, whereas the concentrations of the other metabolites are similar among the three sample types investigated.
Details of results and conclusions are found in Dettmer *et al.* [19].

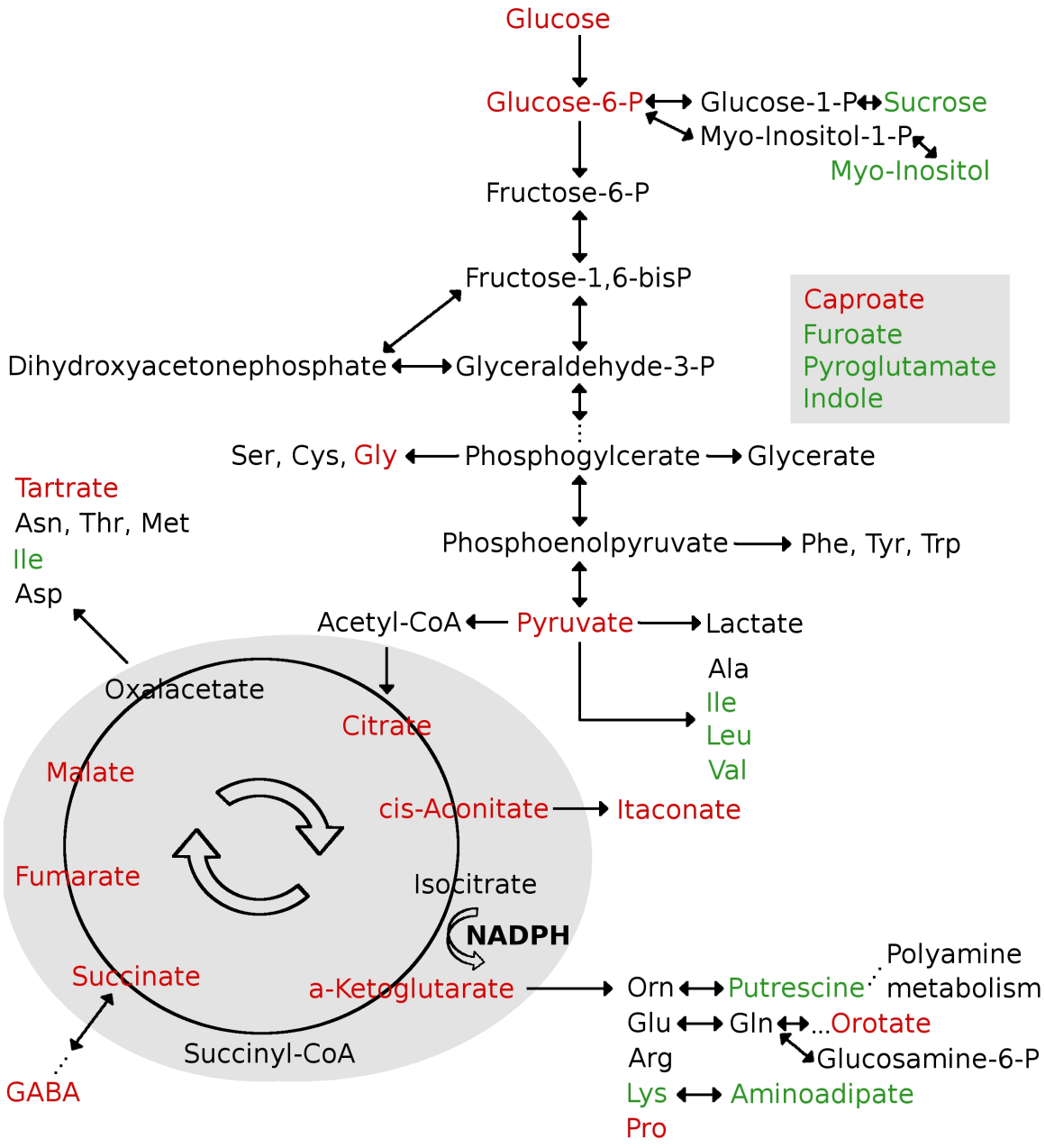


Figure 3.8: Metabolite candidates for an FDR < 0.05 identified between an *E. coli* wild type and a mutant strain are colored in red or green. The metabolite is upregulated in the mutant strain if it is colored in red, and downregulated if it is marked in green.

3.5 Conclusion and outlook

Before comparing metabolic fingerprints, retention time shifts have to be corrected. Assuming that the shifts occur linearily across the whole range of the chromatogram, reference points can be used to fit a linear model for each retention time. However, it has not been shown yet that retention time shifts always occur linearly. The odd-numbered fatty acids as reference points cover the entire retention time window. Thus, they can be used to fit non-linear regression models or linear models locally like in Wang *et al.* [65]. First, Wang *et al.* identify so called landmark peaks across the whole region of the chromatogram. Then, their algorithm uses a local partial linear fitting function to interpolate the retention time of non-landmark peaks located between two landmark peaks in each retention time dimension.

The developed peak alignment module INCA combines peak lists into a single data matrix based on first and second dimension retention times and spectral information. The module is validated by a spike-in experiment. In combination with supervised classification algorithms like shrunken centroid classification, GC×GC-TOF-MS based metabolic fingerprinting provides an effective and simple means to classify samples according to metabolite abundances making it a valuable tool in medical diagnosis and prognostication.

The novel Leco ChromaTOF SC algorithm for GC×GC-TOF-MS data is validated by an spike-in experiment. Results are compared to the developed INCA alignment tool. The spike-in experiment yields less zero values for the SC as compared to the INCA alignment. This is caused by the ability of SC to exclude conflicts during the peak grouping and to access raw data beyond the alignment process. However, peaks present in one of two possible pathological conditions also will be excluded. The improvements of alignment and quantification are depicted in ROC curves and observed-versus-expected fold change plots. Particularly, small fold changes (1.25 - 2.0) show better results with the SC alignment. INCA exploits the characteristic fragmentation behavior of silylated metabolites using the universal m/z of 73 as a quantitative measure. SC constantly assigns the same unique mass trace to an identical compound in different samples and use it for quantification. Thus, the extraction of unique mass traces from mass spectra of aligned metabolites has to be improved for both, INCA and SC, to make it a better tool for metabolic fingerprinting.

Finally, comprehensive GC×GC-TOF-MS is successfully applied to the metabolic

fingerprinting of a mutant and a wild type strain of *E. coli* and INCA is adapted for GC-TOF-MS data and applied to a serum versus plasma collection.

Chapter 4

A new method for estimating class probabilities

In the clinical classification of disease we are not only interested in the prediction of the correct disease type but also in the uncertainties associated with an individual classification. While the performance of classification algorithms has been analyzed in great detail, little attention has been given to the usefulness of probability estimates provided by some classification algorithms. I developed two novel methods for estimating smoothed local probabilities (Sections 4.1 and 4.2), and critically compared them to existing estimation methods on a recently published metabolomics dataset of patients with kidney disease (Section 4.3).

4.1 Smooth Local Error Frequencies (LEF(Smooth))

A drawback of the binning approach described in Chapter 2.3 is its coarseness. All cases in a bin receive the same local error frequency estimate $F_k(x)$ regardless where they fall in the bin, and scores in the same bin can vary substantially. While the monotone regression step is partly compensating for this artefact, I argue that it can not fully adjust for the binning effect and more smooth estimates of $F_k(x_j)$ are needed. Next I will describe two modifications of the LEF concept that combine monotone regression with smooth estimates of local error frequencies.

I propose to use gaussian smoothing kernels

$$K(s(x_j)) = \frac{1}{\sqrt{2\pi\lambda^2}} \exp\left(-\frac{(s(x_j) - s(x))^2}{\lambda^2}\right)$$

to estimate the local error frequencies $F_k(x)$ by

$$F_k(x) = \frac{\sum_{j \in C_k} K(s(x_j))}{\sum_j K(s(x_j))}.$$

The bandwidth λ is the same for all cases x_j (one fits all approach). It is a tuning parameter whose calibration will be discussed in the next section. Note that the kernels are centered at $s(x)$ and that the classification accuracy of the algorithm for cases with scores similar to $s(x)$ still determines the local error frequencies. Different to the binning approach the actual distances of neighboring cases are now taken into account. Once the local error rates are estimated I proceed like in the binning method of [71]. I use monotone regression on the $F_k(x_j)$ employing the PAVA algorithm to achieve class probabilities. Figure 4.1A shows class probability functions with increasing λ indicated by rainbow colors changing from red to violet. The function with minimal negative log-likelihood of true classes $-\log(L)$ and corresponding λ is chosen (see Figure 4.1B).

This constant λ assumption is problematic if the density of scores $s(x_j)$ is far from uniform. In this case, local error frequency estimates are supported by many scores in regions where the scores fall densly which makes them reliable, while in less dense regions estimates are more unreliable. For these situations I propose an adaptive estimator of $F_k(x_j)$.

4.2 Adaptive Local Error Frequencies (LEF(Adapt))

I propose to use the neighbourhood adaptive gaussian smoothing kernels

$$K_{x_j,l}(s(x)) = \frac{1}{\sqrt{2\pi\lambda(x,l)^2}} \exp\left(-\frac{(s(x_j) - s(x))^2}{\lambda(x,l)^2}\right)$$

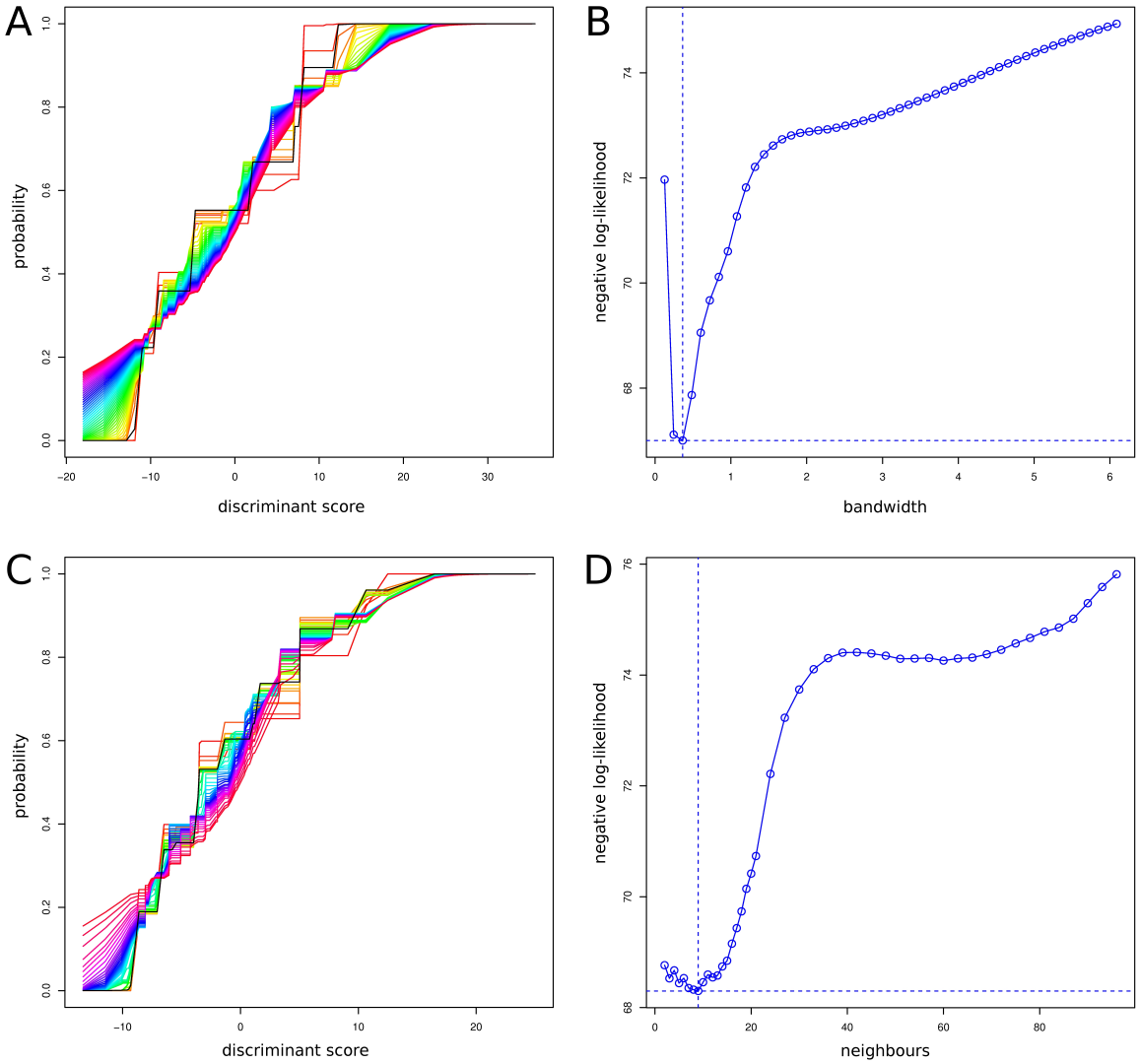


Figure 4.1: A Class probability functions for different values of the bandwidth λ of the gaussian smoothing kernel to estimate local error frequencies LEF(Smooth). The color changes from red to violet with increasing λ. The black line indicates the CPF with minimal negative log-likelihood. **B** The negative log-likelihoods are plotted for different λs. The dotted line indicates the minimum. **C** Class probability functions for different numbers of neighbours to estimated local error frequencies LEF(Adapt). **D** Negative log-likelihoods for varying number of neighbours. The plots are based on a fold of the innermost cross-validation loop from ADPKD patients (top) and healthy donors (bottom).

to estimate the local error frequencies $F_k(x)$ by

$$F_k(x) = \frac{\sum_{j \in C_k} K_{x,l}(s(x_j))}{\sum_j K_{x,l}(s(x_j))}.$$

Note that the kernels are centered at $s(x)$ and that their bandwidths depend on a tuning parameter l and vary across cases. The bandwidth λ is adapted to the local density of scores around $s(x)$. It is narrower in regions where there are many cases with similar scores $s(x_j)$. I achieve this by setting $\lambda(x, l)$ equal to the empirical variance of the l nearest neighbors of $s(x)$. I will discuss tuning of the parameter l in the next section. The adaptation of the kernel bandwidths ensures that the local error frequencies are supported by roughly the same number of neighbouring cases. In addition, the actual similarity of these cases with x is taken into account. Again, once the local error rates are estimated I proceed as in the binning method of Zadrozny *et al.* [71]. I use monotone regression on the $F_k(x_j)$ employing the PAVA algorithm to achieve class probabilities. Figure 4.1C shows class probability functions with increasing number of neighbours l indicated by rainbow colors changing from red to violet. The function with minimal negative log-likelihood of true classes $-\log(L)$ and corresponding l is chosen (see Figure 4.1D).

4.3 Comparing probability estimators

The class probability estimators described in Section 2.3 are compared in the context of a recently published metabolomic profiling study on kidney diseases [27]. Autosomal dominant polycystic kidney disease (ADPKD) is a frequent cause of kidney failure. Due to a lack of reliable laboratory tests early in the disease it is usually diagnosed at a progressed stage of renal cystic transformation. The study addresses the challenge to diagnose ADPKD based on metabolomic fingerprints.

The data set comprises 168 urine samples measured using 1D nuclear magnetic resonance (NMR) spectroscopy. 54 samples were obtained from patients with autosomal polycystic kidney disease. These need to be separated from samples taken from healthy volunteers (46 samples), and samples from patients with compromised kidney function but no ADPKD (52 samples from diabetes mellitus patients and 16 samples from patients 3 months after renal transplantation). More details on the composition of cases in the study can be found in Table 4.1. NMR 1D spectra are

Description	Index	Size
ADPKD	1	54
ADPKD with medication	1A	35
ADPKD without medication	1B	19
Healthy	2	46
Other CKD		
Renal transplant without rejection	3	16
Diabetes with microalbuminuria	4	30
Diabetes without microalbuminuria	5	22

Table 4.1: Patient groups defined within the ADPKD dataset

split into 701 equally sized buckets and globally normalized to the signal of the CH_2 group of creatinine to ensure sample to sample comparability. Furthermore compatibility across metabolites is ensured by applying the glog transformation [51]. For full details of sample preparation and data preprocessing see Gronwald *et al.* [27].

The focus of this section is the assessment and comparison of class probability function (CPF) estimators. In the original publications all estimators are used in combination with different classifier learning algorithms. Heterogeneity of classifiers can severly confound a comparison of CPF estimators. Moreover, none of the estimators appears to be tailored to a special classification approach. They can be easily adapted so that all use the same classification algorithm. Here the popular shrunken centroid classifier is used [60].

4.3.1 Modification of classification performance

Qualitative classifications can be obtained directly from the linear classifier. No CPF estimation is necessary. Nevertheless, once a CPF is estimated it is natural to assign cases with a class probability $p_k(x_j)$ above 0.5 to class k and those with probabilities below 0.5 to the other class. This might lead to a reassignment of some cases, as in the example shown in Figure 4.2 where 14 cases were reassigned. Moreover, since the sparseness parameter Δ needs to be calibrated for all estimators independently, there is an indirect effect of the choice of CPF estimator on the overall classification performance.

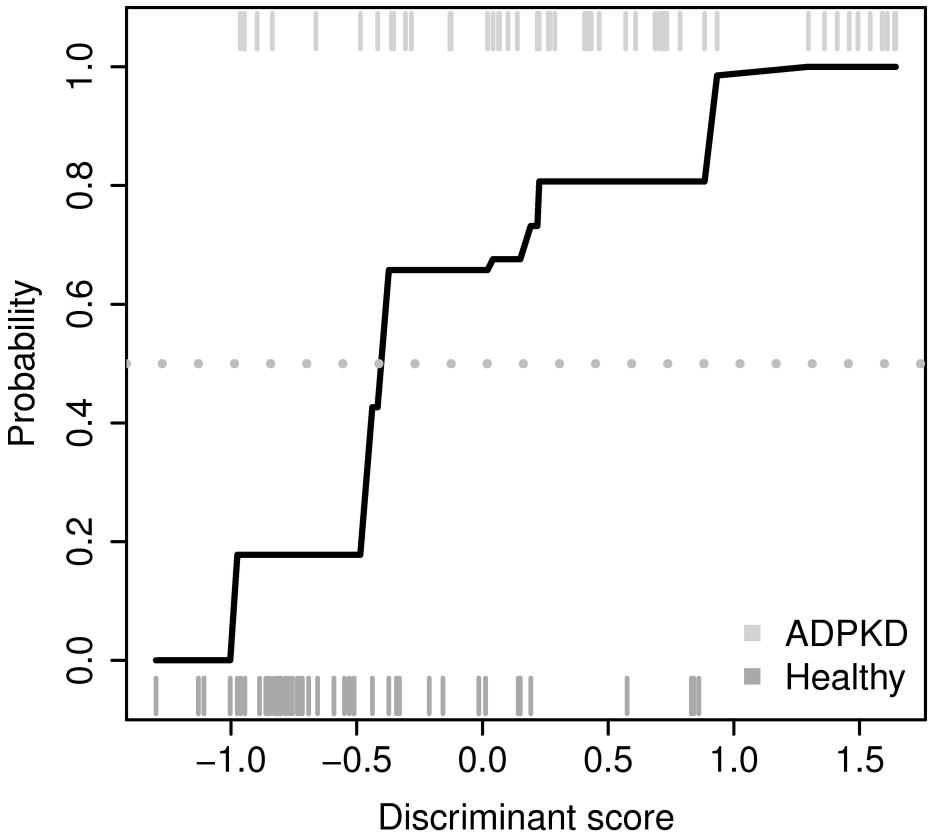


Figure 4.2: The Figure shows estimated classification probabilities for cross-validated scores for method LEF(Adapt) exemplarily. The stripes on the x-axis show the cross-validated classification scores $s(x_j)$ from ADPKD patients (top) and healthy donors (bottom).

Table 4.2 shows the global classification performance in several pairwise classifications using the plain shrunken centroid classifier (equal to the Naive Bayes (NB) estimator) and its modifications resulting from the different CPF estimators. Groups 1A and 1B correspond to ADPKD patients with and without medication for arterial hypertension, group 2 consists of healthy volunteers. Patients 3 months after renal transplantation without rejection are assigned to group 3 and diabetes mellitus type 2 patients with and without microalbuminuria are in groups 4 and 5, respectively. For group sizes see Table 4.1. Classification accuracies on the various training sets of the outer cross-validation loop can be found in the Appendix A.3.1.

The local error frequency method LEF(Adapt) reaches the highest average perform-

Group comparison		Classification performance					
		NB	CB	BReg	Bin	Smooth	Adapt
1	2	76.0	75.0	76.0	71.0	74.0	76.0
1	3	97.1	95.7	92.9	97.1	97.1	98.6
1	4	82.1	82.1	85.7	88.1	85.7	85.7
1	5	82.9	88.2	86.8	85.5	84.2	85.5
1	2,3,4,5	75.0	78.6	76.8	76.2	76.8	76.2
1A	2	86.4	86.4	86.4	85.2	80.2	84.0
1B	2	63.1	60.0	66.2	64.7	64.6	66.2

Table 4.2: Classification performances of the outer cross-validation loop for the six probability estimation methods, Naive Bayes (NB), Compound Bayes (CB), binary regression (BReg) and the local error frequency methods using binning (Bin) and smoothing (Smooth/ Adapt). For each patient group comparison (rows) performances are listed for each estimation method. The assignment of patient groups to the indices can be found in Table 4.1.

ance of 81.7%, followed by binary regression (BReg) and LEF(Bin) with 81.5% and 81.1% respectively. Overall the classification accuracies of all CPF estimators differ 3.6 to 5.7% depending on the pair of groups. Table 4.2 also shows that the performance of a given method depends on the investigated pair of groups. Therein, Compound Bayes (CB), binary regression (BReg) and LEF(Adapt) won most frequently.

Gronwald *et al.* [27] used support vector machines for classification which may explain the differences in performance. Therefore, we also run the classification using scores from support vector machines. Therein, the error rates were 5 to 10% smaller and were comparable to the error rates in Gronwald *et al.* [27] (see Appendix A.3.2).

4.3.2 Sparseness Bias

As explained above, I optimize the number of classifier features of each CPF estimator seperately. Nevertheless, class probability estimation should be possible for any number of features. The estimated probabilities should not depend on the number of features except for reflecting shifts in the overlaps of scores. Figure 4.3 shows discriminant scores and a PAM-based CPF (NB) of a 5 feature and a 200 feature classifier.

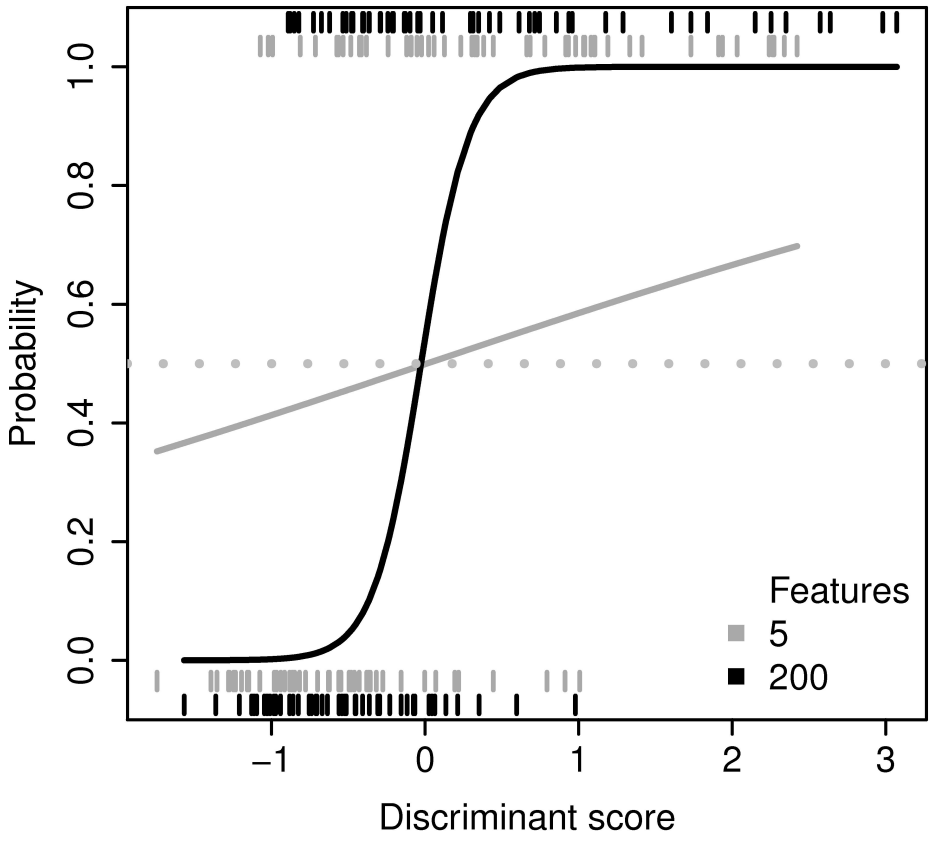


Figure 4.3: Scores of a 5 feature classifier (gray) and a 200 feature estimator (black) together with a CPF estimated by the PAM approach are shown. Scores on the x-axis were standardized to mean zero and standard deviation 1 for comparing CPFs directly. Cross-validated scores are indicated as gray and black stripes for the comparison of ADPKD patients (top) and healthy controls (bottom).

In order to ensure comparability, discriminant scores are standardized to mean zero and standard deviation one. Both refer to a comparison of ADPKD patients with healthy donors. One can observe that the overlap of scores only marginally changes for the two classifiers. However the estimated CPFs change dramatically. For the 5 feature curve all patients receive probabilities between 0.35 and 0.70 flagging them all as unreliable. A result that does not reflect the score distributions well since scores above +1 are only reached by ADPKD patients and cases with such scores should be considered reliable ADPKD cases. In contrast, for the 200 metabolites curve all probabilities are either close to 0 or close to 1. Hence the curve considers all clas-

sifications reliable, which is misleading given the mix of classes in cases with scores between -1.1 and $+1$. The PAM estimator has an obvious sparseness bias. The more features are included in a classifier the more confident the estimator becomes regardless of how the classes overlap. I next investigate to which extent the individual methods suffer from such a sparseness bias.

Figure 4.4 shows scatter plots of estimated probabilities for classifiers including different numbers of metabolites. For each number of features, the class probabilities of all cases are subtracted by those obtained for the same patient by the 2 feature classifier. The differences (y-axis) are plotted against the number of features (x-axis, logarithmic). The density of points in the scatter plots is coded on a blue scale with dark regions indicating high density. The first plot clearly shows the sparseness bias of the PAM approach (NB). The 2 gene classifier gives probabilities around 0.5. This is kept for classifiers up to 10 features. Classifiers with many features produce probabilities near 0 or 1 leading to differences of +/- 0.5. This effect becomes manifest for classifiers with 75 features or more. None of the other methods showed this behavior. Although differences of class probabilities can reach high values, there is no systematic sparseness bias observable. For the Compound Bayes estimator and the binary regression estimator the majority of differences stay close to zero. For the local error rate based methods the differences are greater but also here I do not observe a systematic trend towards more self-confident probabilities when more features are included. Another way to look at the sparseness bias is to calculate the difference in the range for each sample. The distribution of differences are shown as boxplots in Figure 4.5 for each CPF method.

4.3.3 Calibration

A straight forward criterion to evaluate probability estimators is calibration. An estimator is well calibrated, if in the long run the relative frequency of true classifications of cases with estimated class probabilities falling in a small interval $[p_0-\epsilon, p_0+\epsilon]$ is close to the estimated probability p_0 [17]. Our ADPKD data set is not large enough to test long run performance.

That is why we simulated data with structures similar to the ADPKD data for the comparison of the ADPKD patients and healthy controls. Therein, pairs of samples (x_1, x_2) were drawn randomly from the original data set and sample x_1 was shifted

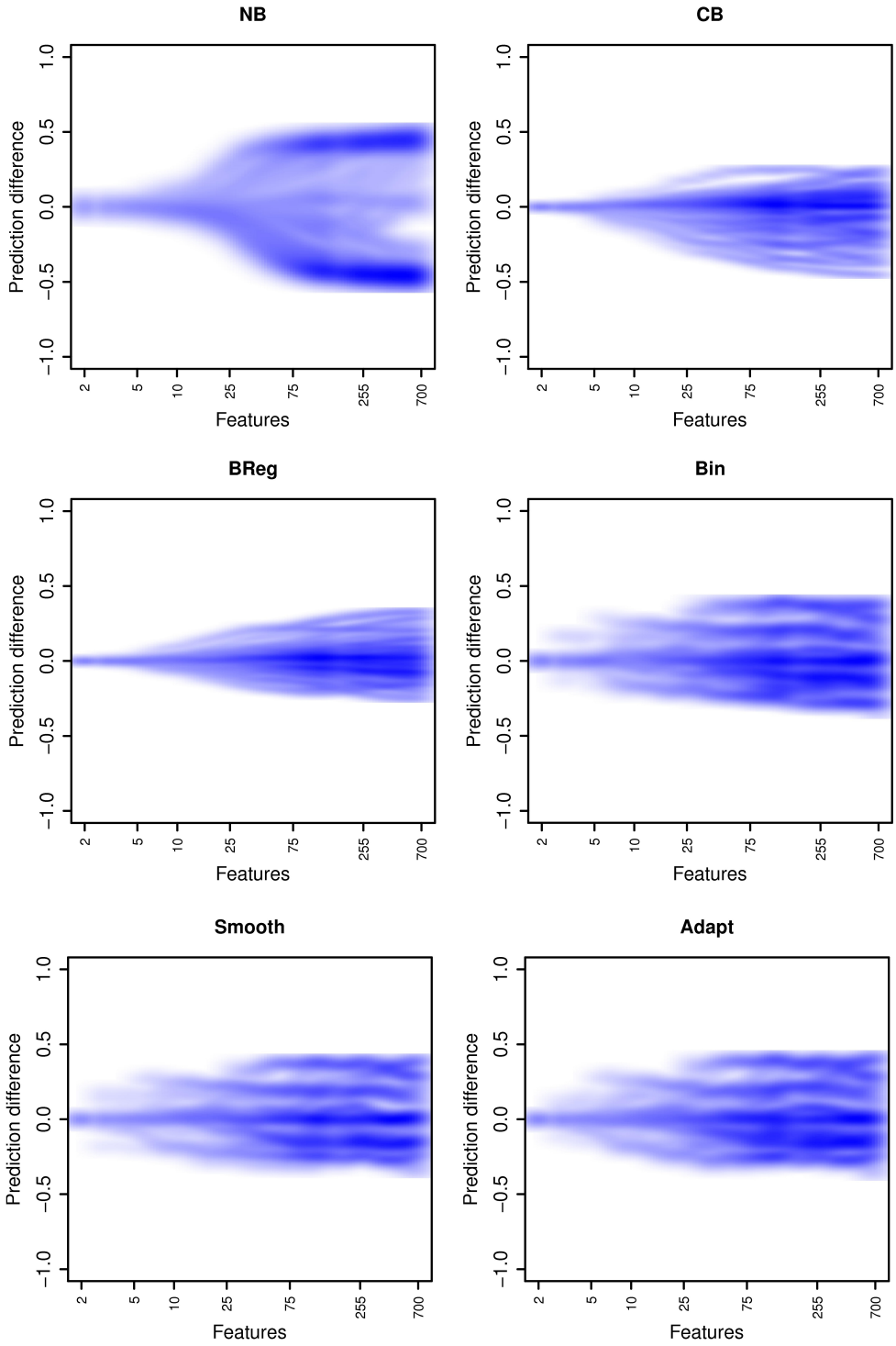


Figure 4.4: Smoothed scatter plots of estimated probabilities for classifiers including different numbers of metabolites. For each patient the class probability of the 2 feature classifier was subtracted from probabilities for all numbers of metabolites for the given patient. The differences (y-axis) are plotted against the number of features (x-axis, logarithmic). The density of points in the scatter plots is coded on a blue scale with dark regions indicating high density.

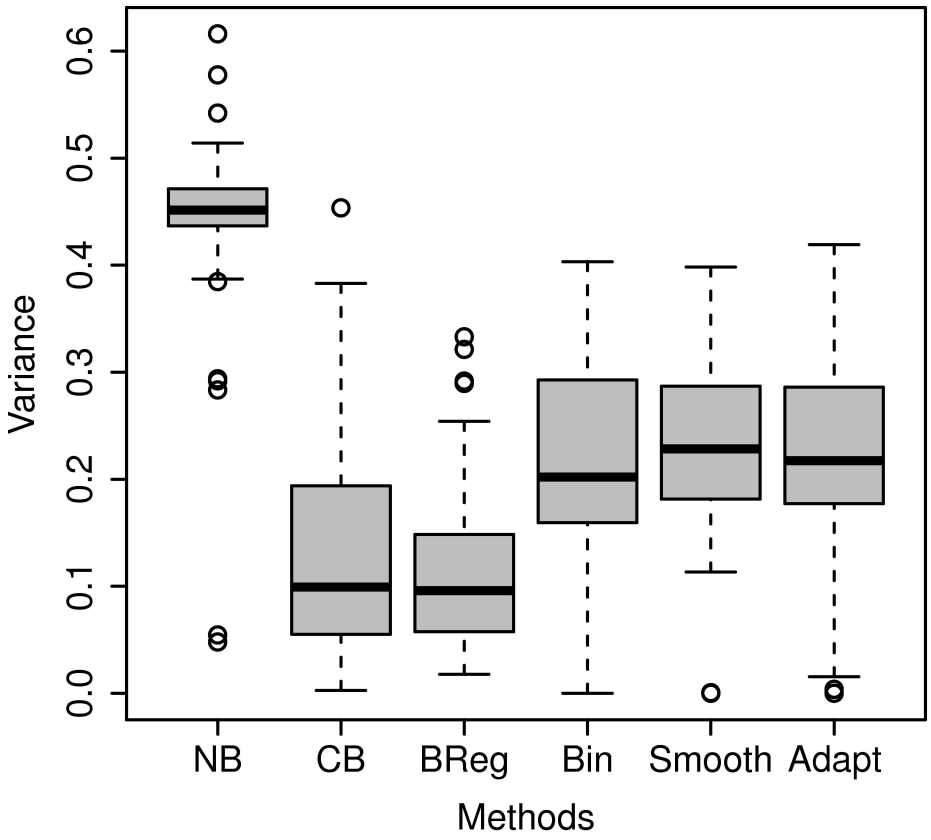


Figure 4.5: Differences in the range of estimated probabilities for classifiers including different numbers of metabolites. For each patient the difference in the range of class probabilities among all numbers of features is calculated. The distribution of differences (y-axis) is plotted for all CPF methods as boxplots.

on the line spanned by samples x_1 and x_2 such that $x_1' = x_1 + \beta \cdot (x_2 - x_1)$ where $\beta \in [-1, 1]$. This procedure was repeated, for each class separately, as long as the simulated data set was of the same size as the original one. Finally, 30 data sets were used for further analyses. Because the distances of the new data points were smaller within classes and larger between classes compared to the original data, learning a classifier was easier and the misclassification rate decreased by more than 10%. With decreasing learning complexity the classification probabilities tend to zero and one. This was compensated by drawing samples from the data set with probability of 99% for the currently simulated class.

Figure 4.6 compares estimated class probabilities to long run classification accuracies.

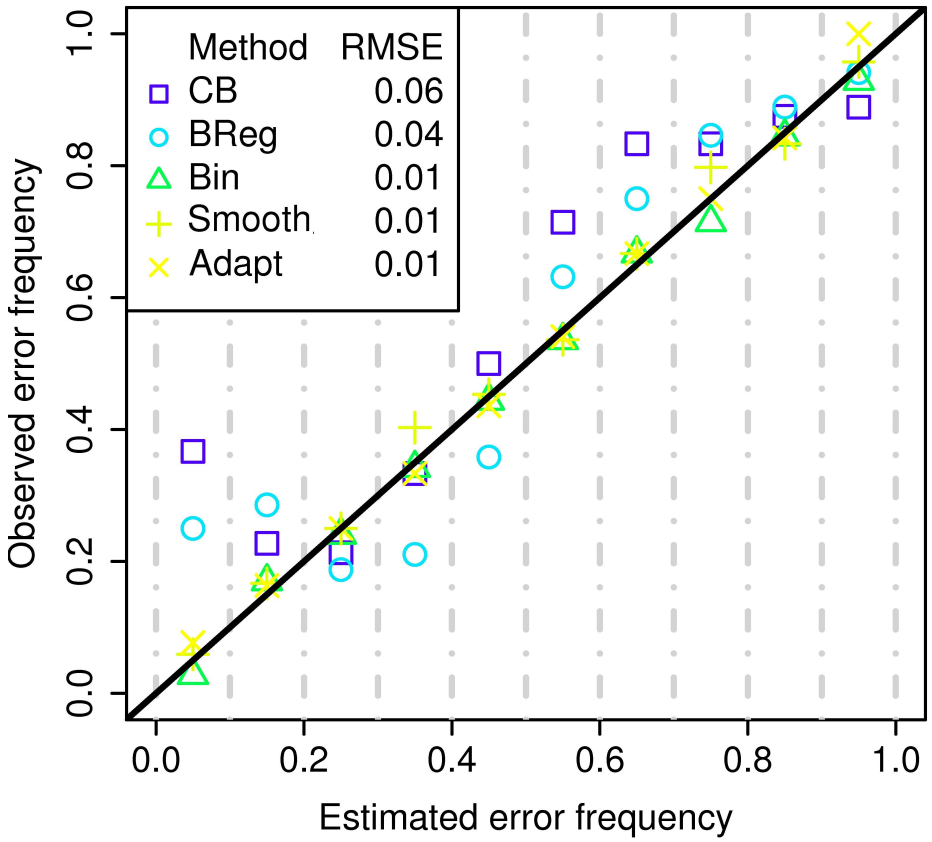


Figure 4.6: Estimated class probabilities to long run classification accuracies are shown in a reliability diagram. Both estimated error probabilities (x-axis) and observed error frequencies (y-axis) were collected in bins of width 0.1. If the classifier is well calibrated, all points fall close to the $x = y$ line. Root mean square errors (RMSE) from this line quantify the calibration quality of the estimator.

Both estimated error probabilities and observed error frequencies were collected in bins of width 0.1. If the classifier is well calibrated, all points fall close to the $x = y$ line. Root mean square errors (RMSE) from this line quantify the calibration of the estimator. I found that the RMSE was large for the Compound Bayes estimator and for binary regression while it was decreased by a factor of up to 6 for the local estimation methods, indicating that the local estimators are better calibrated. To test whether calibration depends on the number of samples used for estimator training, I rerun our evaluation for sample sizes between 100 and 3200. Figure 4.7 shows that in the study the RMSE values remain relatively constant, and that local estimators outperform the competing methods independent of the sample size, con-

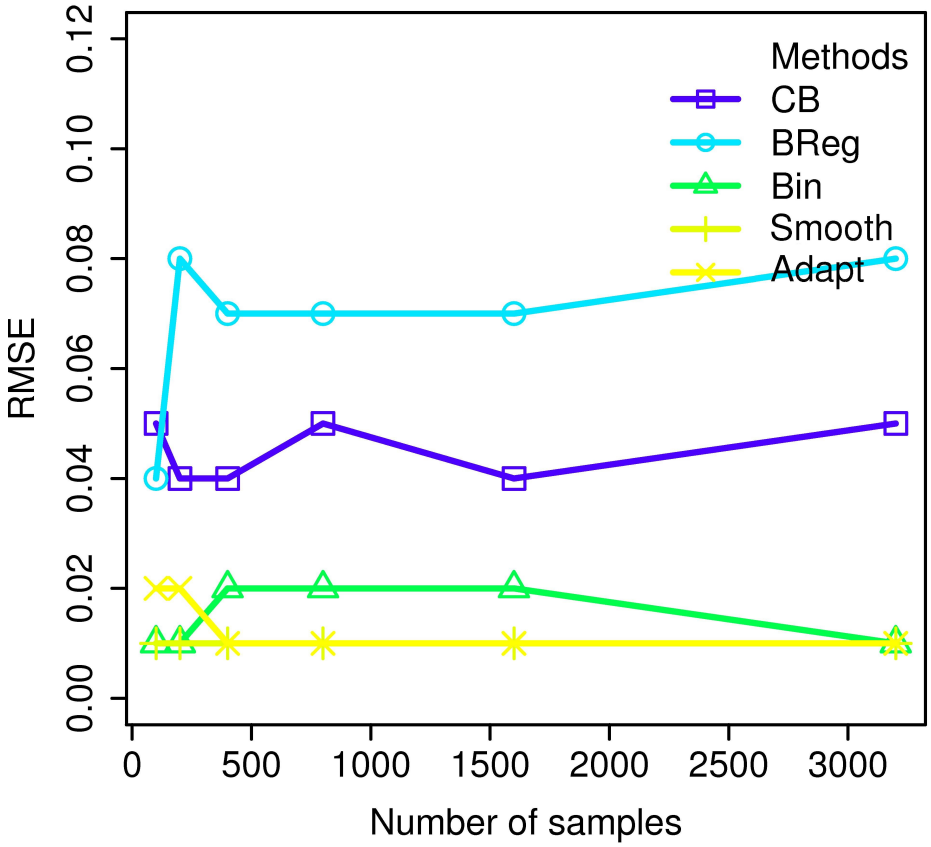


Figure 4.7: Calibration of CPFs with increasing sample size. To test whether calibration depends on the number of samples used for estimator training, the evaluation of calibration was repeated for sample sizes between 100 and 3200 and RMSEs were plotted for all CPFs with increasing sample size.

trary to Niculescu-Mizil *et al.* [47] who report a superior performance of the binary regression estimator for data sets with less than 1000 samples in the domain of text classification.

4.3.4 Variance of estimators

Another classical criterion for estimator evaluation is the variance of the estimator with respect to sampling. Typically, flexibility of an estimator comes at the price of increased variance. The binary regression approach is the most rigid in our collection with only two adjustable parameters. Compound Bayes has four parameters, while

the non-parametric local estimators seem to be the most flexible ones. Here I assess the variance of probability estimators across multiple simulation runs. For this purpose the original data of ADPKD patients and healthy controls were enclosed as test set in the outer loop of the cross-validation scheme within all simulation runs (from Section 4.3.3). Hence, each case was predicted number of outer folds times simulated data sets for each CPF method. For each fold a variance was calculated for each case among simulations. Finally, the median variance of each case among folds and methods was evaluated. Thus, I assess the variance for each sample individually. Figure 4.8 shows box plots of the variance of estimated class probabilities across all samples. I observed, that the variance was smallest for the binary regression estimator followed by the Compound Bayes method and larger for the local error estimators, which is not surprising given the increased flexibility of these estimators. Moreover, I observed that the local estimators displayed a wider range of variances across samples showing high variance for samples in the gray zones between classes.

4.3.5 Identifying reliable classifications

The ultimate goal of class probability estimation is the identification of those samples that can be reliably classified. Misclassifications should be rare among samples with high class probabilities. This property of an estimator is related to calibration in that I relate long run misclassification rates with estimated probabilities. However, the focus here is on extreme probabilities only. If an estimator is poorly calibrated for probabilities around 0.5 this is less of a problem since clinicians would not base treatment decisions on classifications that are labeled unreliable. If however, an estimated probability is close to 1, it must be reliable since a clinician might want to adjust treatment decisions based on this diagnostic result. Moreover, there is a trade-off between the reliability of a diagnosis and the number of samples that receive class probabilities close to 1. An estimator might assign extreme probabilities only to a small number of cases thus obtaining very low misclassification rates among these cases. However, this estimator might also miss many cases that could actually be reliably classified. Figure 4.9 shows the trade-off between the percentage of correct classifications and unclassified cases. Therein, a confidence threshold α is varied from 0.5 to 1. Samples below the threshold are left unclassified (x-axis), whereas the percentage of correct classifications is computed among samples above α (y-axis).

I observed that the percentage of correct classifications of the local methods does not

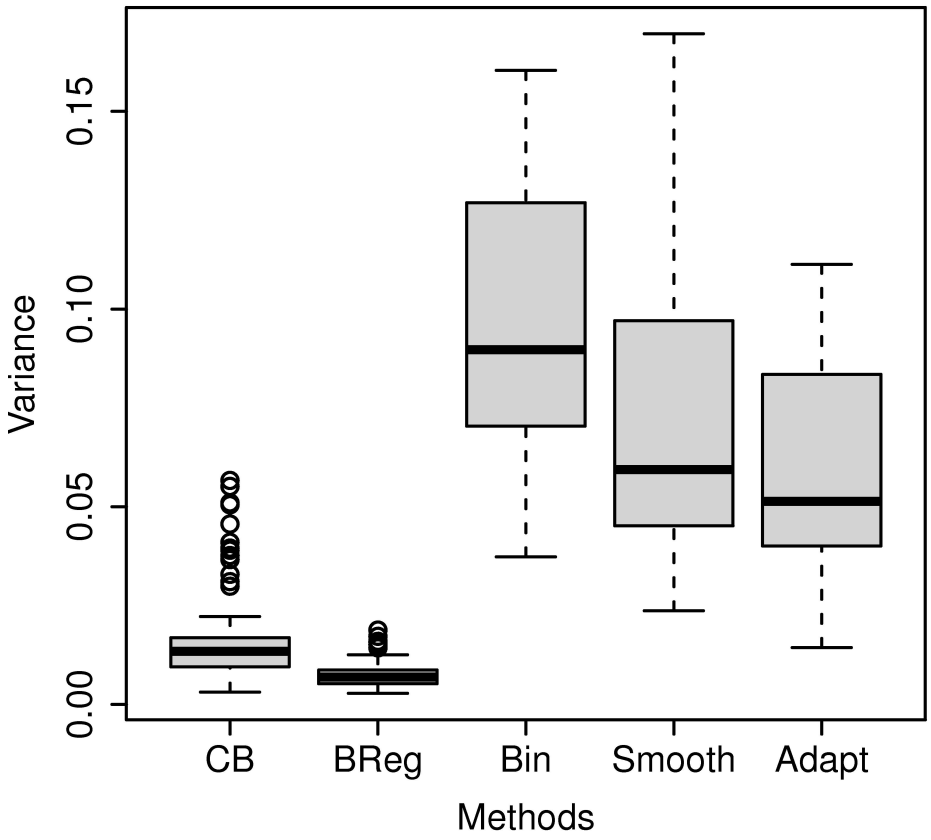


Figure 4.8: Variance of probability estimators across multiple simulation runs. Original data of ADPKD patients and healthy controls were enclosed as test set in the outer loop of the cross-validation scheme within all simulation runs. Each case was predicted number of outer folds times simulated data sets for each CPF method. For each fold a variance was calculated for each case among simulations. Finally, the median variance of each case among folds and methods was used for plotting.

fluctuate as much as that of the Compound Bayes and binary regression. Furthermore, the local methods reached 100% correct classifications faster than the others. For $\alpha = 0.90$ (indicated by the vertical dashed lines in Figure 4.9), Compound Bayes left 76% of samples unclassified with 87.5% correct classifications, 2.5% below the confidence level, and binary regression left 85% unclassified being 100% sensitive. The local methods left 78 - 81% cases unclassified and classified the remaining samples correctly.

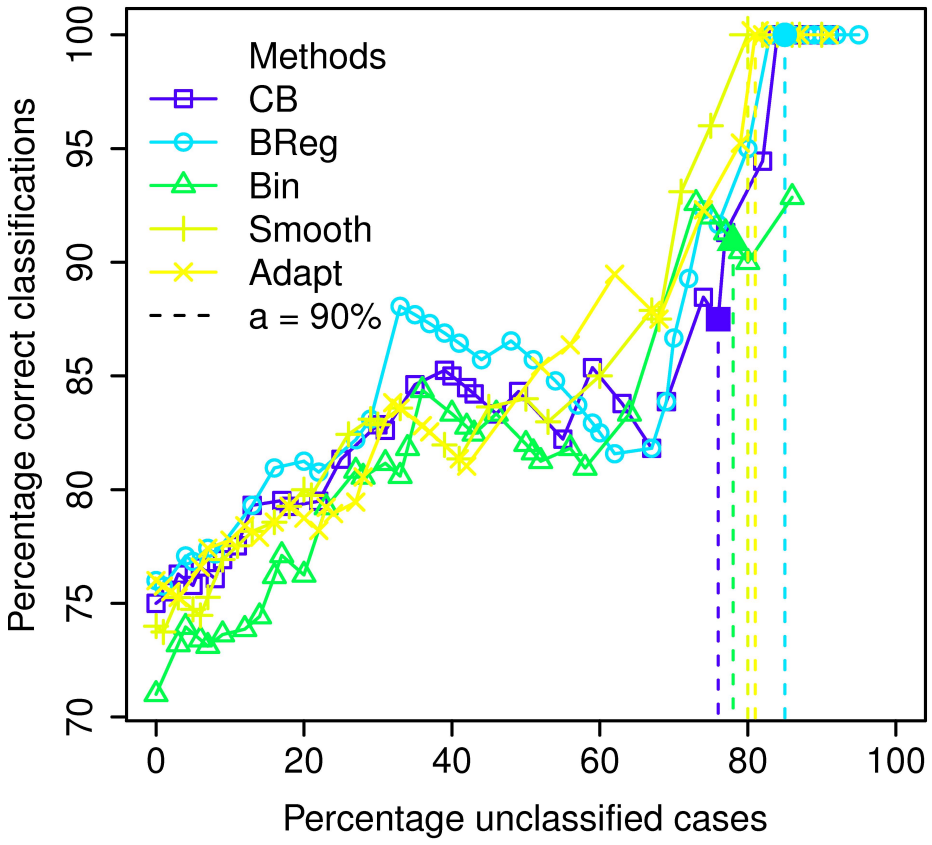


Figure 4.9: Reliability of classifications with increasing confidence level α. α is varied from 0.5 to 1 estimated probability. Samples below the threshold are left unclassified (x-axis), whereas the percentage of correct classifications is computed among samples above α (y-axis). The vertical dashed lines indicate the 90% confidence level for the different CPF estimators.

4.4 Conclusions and outlook

Before a clinician decides on the treatment of a patient, the reliability of the diagnosis must be known. In machine learning reliability can be expressed in terms of classification probabilities. Since little attention has been given to the usefulness of probability estimates so far, I have compared a selection of class probability estimators including Bayes methods, binary regression and local error based methods in the context of metabolomics based diagnosis of disease. I found that local error based estimators perform superior for instance to more widely used methods, the PAM program, binary regression and Compound Bayes classifiers. Strikingly, the PAM approach displayed a strong sparseness bias. I do not recommend its use in clinical diagnosis at all. Binary regression based estimators display the least variance but are inferior with respect to all other criteria evaluated here. A collection of three local error based estimators performed best overall with only marginal differences between the individual implementations. I conclude that this type of approach is the method of choice.

I evaluated class probabilities for binary decisions. However, clinical diagnosis includes more possible entities. Statnikov et al. [59] have evaluated a number of multi-classification algorithms for gene expression data. They report classification accuracies between 70% and 80% when trying to distinguish up to 14 different tumor types. They also observe that the more disease types considered in a classification task, the more difficult the classification is. Mukherjee [45] argues that in problems with more classes, the classification algorithm has to determine a larger number of separation boundaries. A common approach for constructing a multi-class classifier is to determine separation boundaries between each pair of classes using a two-class classification algorithm and integrate the results to generate a single classification. Hastie et al. [33] suggest a coupling scheme to summarize pairwise class probabilities into a class probability per class for the given sample.

The ADPKD patients of the metabolomics dataset are grouped into patients with and without medication. These two groups can not be separated by a classifier (classification accuracies not shown). In a multi-class setting these groups lower the overall classification accuracy.

Future work could improve multi-class classification by either merging undistinguishable classes as long as this leads to an overall gain in accuracy, or reject classification

in case of doubt. New patients may be borderline as well as do not belong to any class used for training. Hanczar and Dougherty [32] reject patient samples such that the overall classification accuracy satisfies the user-defined accuracy. This approach is limited to two-class problems and may be extended to multi-class settings.

Appendix A

Appendix

A.1 Probability transformation of Naive Bayes (NB) estimates

Generally, the discriminat score for a nearest centroid classifier

$$\delta_k(x) = \sum_{i=1}^{p} \frac{(x_i - \bar{x}_{ik})^2}{\sigma_i^2} - 2 \cdot log\pi_k$$

is calculated independently for each class, where the sum runs over all genes with non-zero weights after shrinkage Δ, σ_i is the pooled within-class standard deviation of gene i and π_k the estimated proportion of cases from class k in the entire population. A case is then assigned to the group k with minimal $\delta_k(x)$. Classification probabilities are also derived from the $\delta_k(x)$ by assuming independence and normality with equal within classes variance of all p classifier genes. Under these assumptions Bayes theorem yields for two class classification probabilities

$$p_k(x) = \frac{e^{-\frac{1}{2}\cdot\delta_k(x)}}{e^{-\frac{1}{2}\cdot\delta_1(x)} + e^{-\frac{1}{2}\cdot\delta_2(x)}}. \tag{A.1}$$

Note that the nearest shrunken centroid classification rule defines a separating hyperplane with normal vector $w_i = \frac{2\cdot(\bar{x}_{i1}-\bar{x}_{i2})}{\sigma_i^2}$, and Equation 2.1 can be translated to

$$p_k(x) = \frac{1}{1 + e^{-\frac{1}{2}\cdot s(x)}}. \tag{A.2}$$

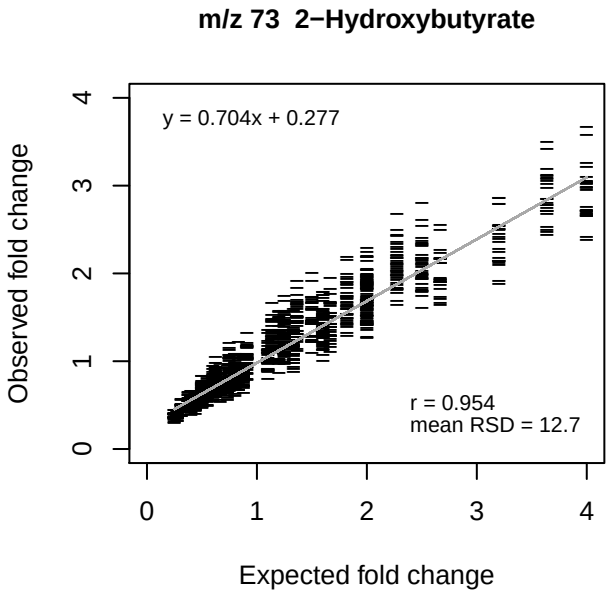
Proof of Equation A.2:
The right side of Equation A.1 divided by $e^{-\frac{1}{2}\delta_1(x)}$ yields

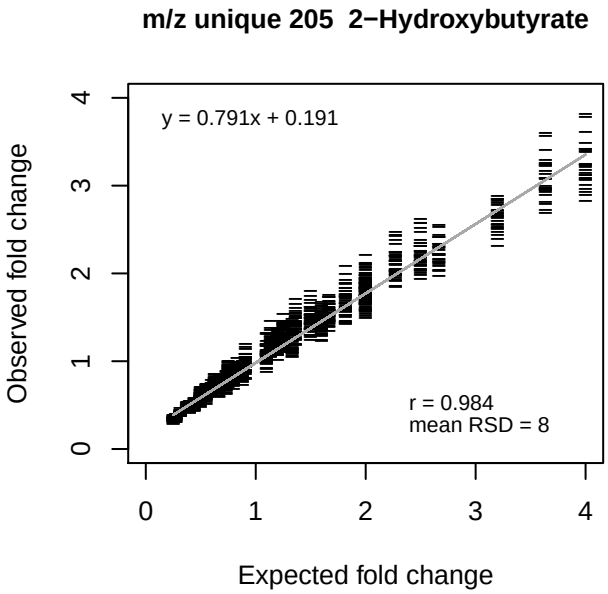
$$\frac{1}{1+\frac{e^{-\frac{1}{2}\cdot d_2(x)}}{e^{-\frac{1}{2}\cdot d_1(x)}}} = \frac{1}{1+e^{-\frac{1}{2}\cdot\underbrace{(d_2(x)-d_1(x))}}}.$$

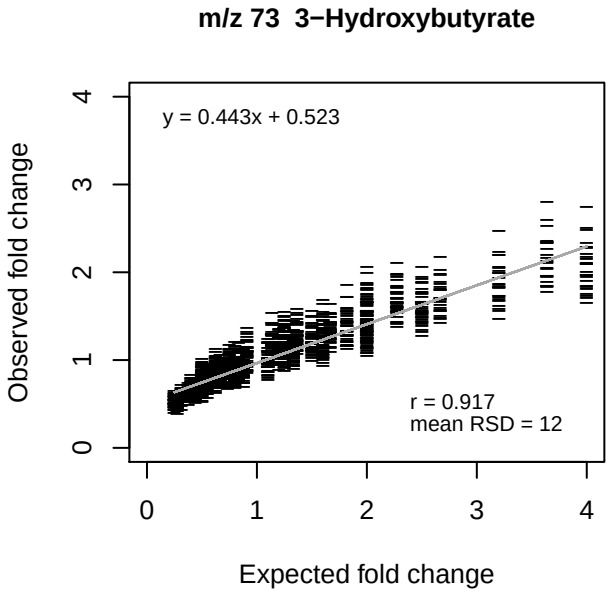
The exponent of the second summand in the denominator can be further translated
$d_2(x) - d_1(x) =$
$= \sum_i \frac{(x_i-\bar{x}_{i2})^2}{\sigma_i^2} - 2\cdot log_2(\pi_2) - \sum_i \frac{(x_i-\bar{x}_{i1})^2}{\sigma_i^2} + 2\cdot log_2(\pi_1)$
$= \sum_i \left[\frac{(x_i-\bar{x}_{i2})^2}{\sigma_i^2} - \frac{(x_i-\bar{x}_{i1})^2}{\sigma_i^2}\right] - 2\cdot(log_2(\pi_2) - log_2(\pi_1))$
$= \sum_i \frac{(x_i-\bar{x}_{i2})^2-(x_i-\bar{x}_{i1})^2}{\sigma_i^2} - 2\cdot log_2\frac{\pi_2}{\pi_1}$
$= \sum_i \frac{x_i^2-2\cdot x_i\bar{x}_{i2}+\bar{x}_{i2}^2-x_i^2+2\cdot x_i\bar{x}_{i1}-\bar{x}_{i1}^2}{\sigma_i^2} - 2\cdot log_2\frac{\pi_2}{\pi_1}$
$= \sum_i \frac{x_i\cdot 2\cdot(\bar{x}_{i1}-\bar{x}_{i2})+\bar{x}_{i2}^2-\bar{x}_{i1}^2}{\sigma_i^2} - 2\cdot log_2\frac{\pi_2}{\pi_1}$
$= \sum_i x_i\cdot\underbrace{\frac{2\cdot(\bar{x}_{i1}-\bar{x}_{i2})}{\sigma_i^2}}_{w} + \underbrace{\sum_i \frac{\bar{x}_{i2}^2-\bar{x}_{i1}^2}{\sigma_i^2} - 2\cdot log_2\frac{\pi_2}{\pi_1}}_{b}.$

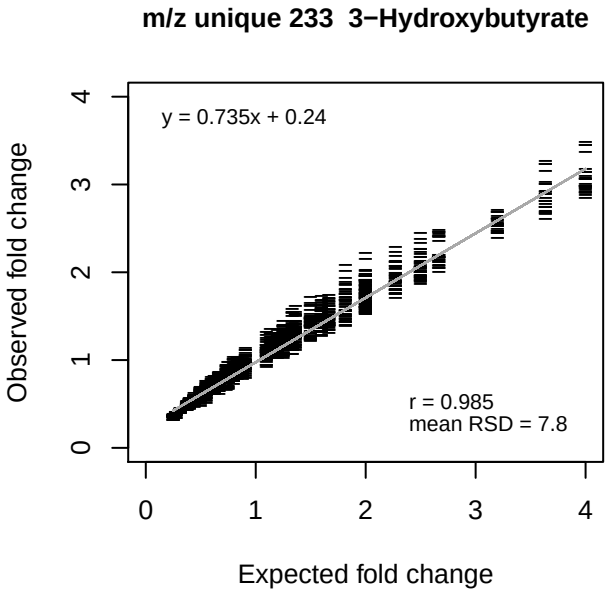
The last line implies a hyperplane with normal vector w and distance to the origin b.

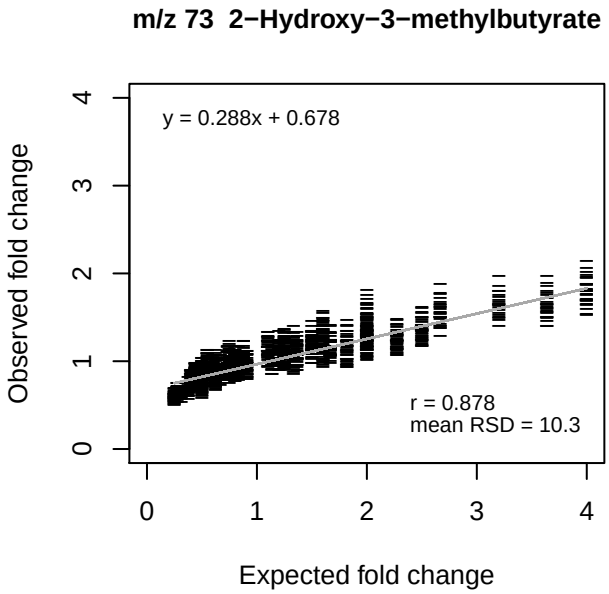
A.2 Quantitative reproducability of spiked-in fold changes

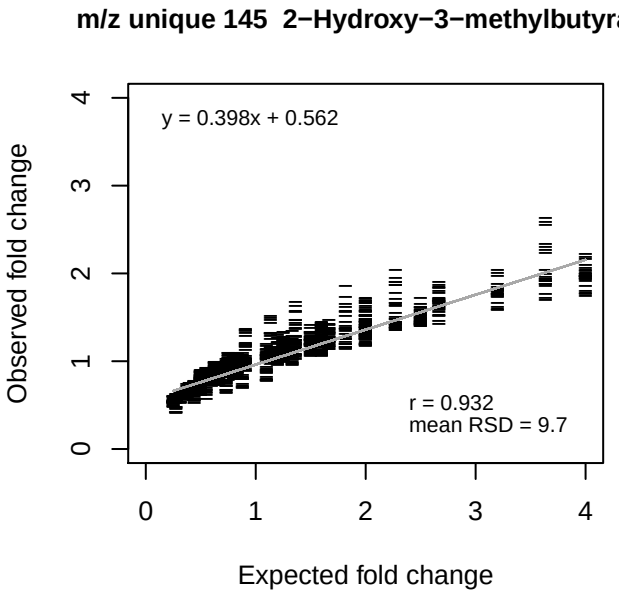

m/z 73 2−Hydroxybutyrate
y = 0.704x + 0.277
r = 0.954
mean RSD = 12.7
Observed fold change
Expected fold change
0
1
2
3
4

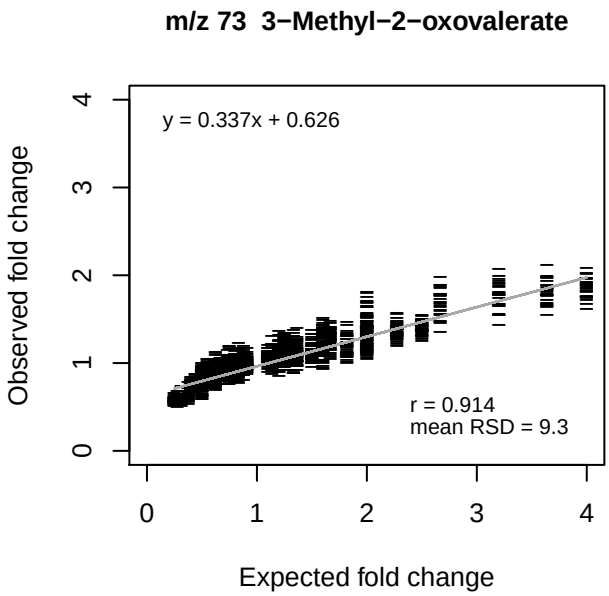

m/z unique 205 2−Hydroxybutyrate
y = 0.791x + 0.191
r = 0.984
mean RSD = 8
Observed fold change
Expected fold change
0
1
2
3
4

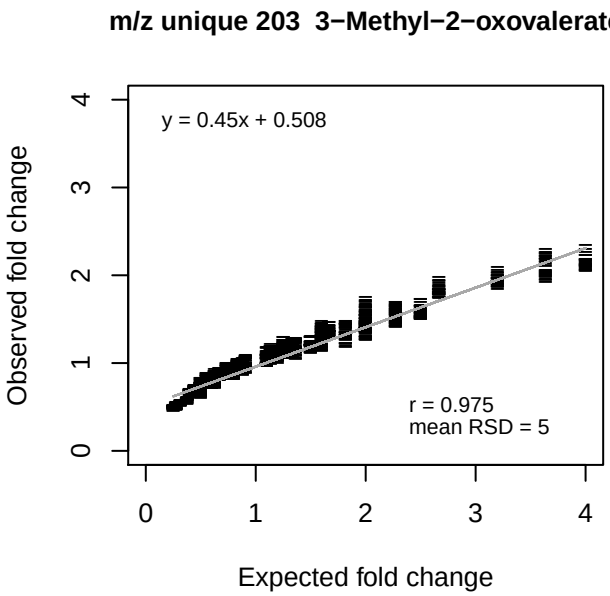

m/z 73 3−Hydroxybutyrate
y = 0.443x + 0.523
r = 0.917
mean RSD = 12
Observed fold change
Expected fold change
0
1
2
3
4

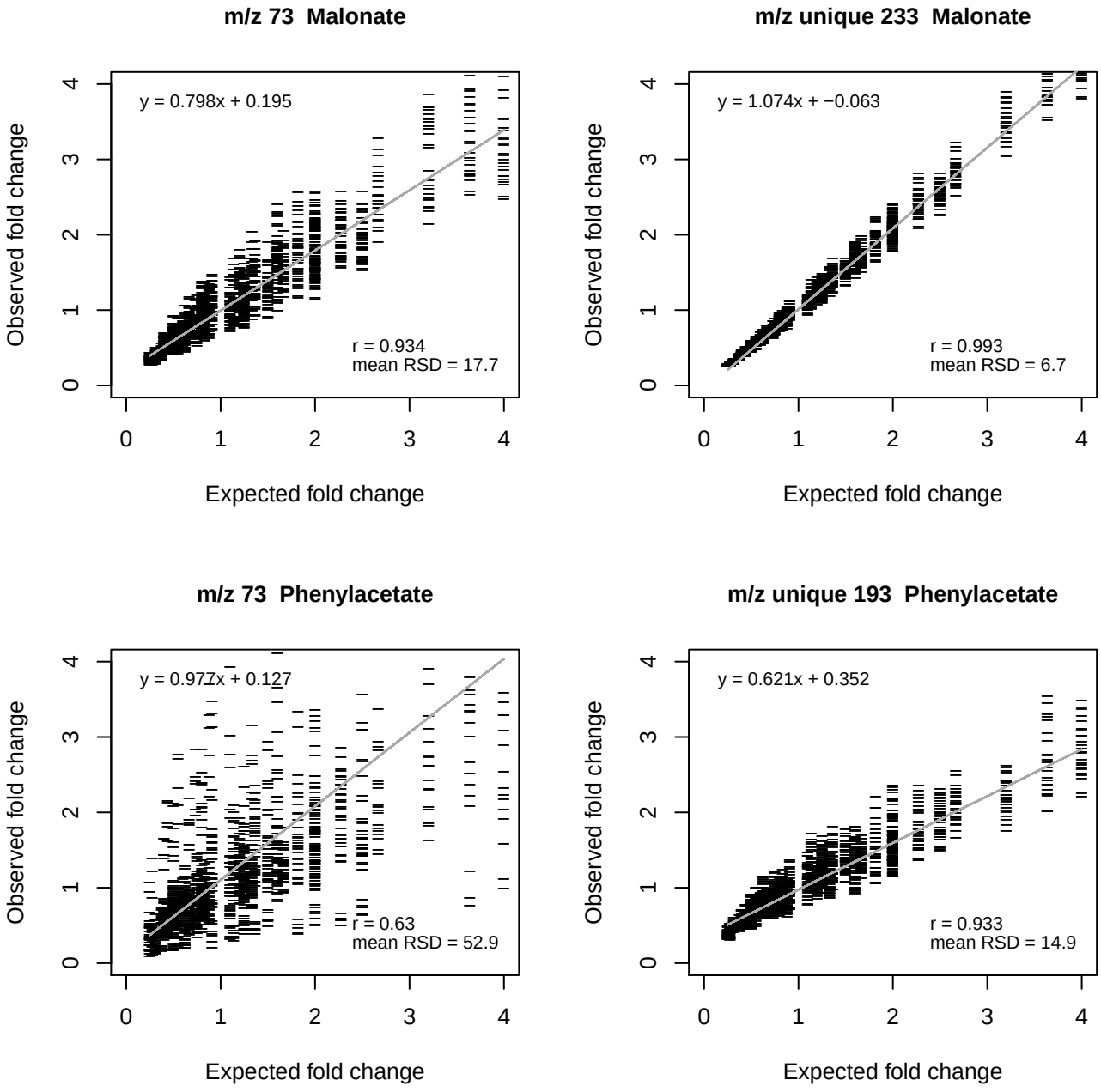

m/z unique 233 3−Hydroxybutyrate
y = 0.735x + 0.24
r = 0.985
mean RSD = 7.8
Observed fold change
Expected fold change
0
1
2
3
4

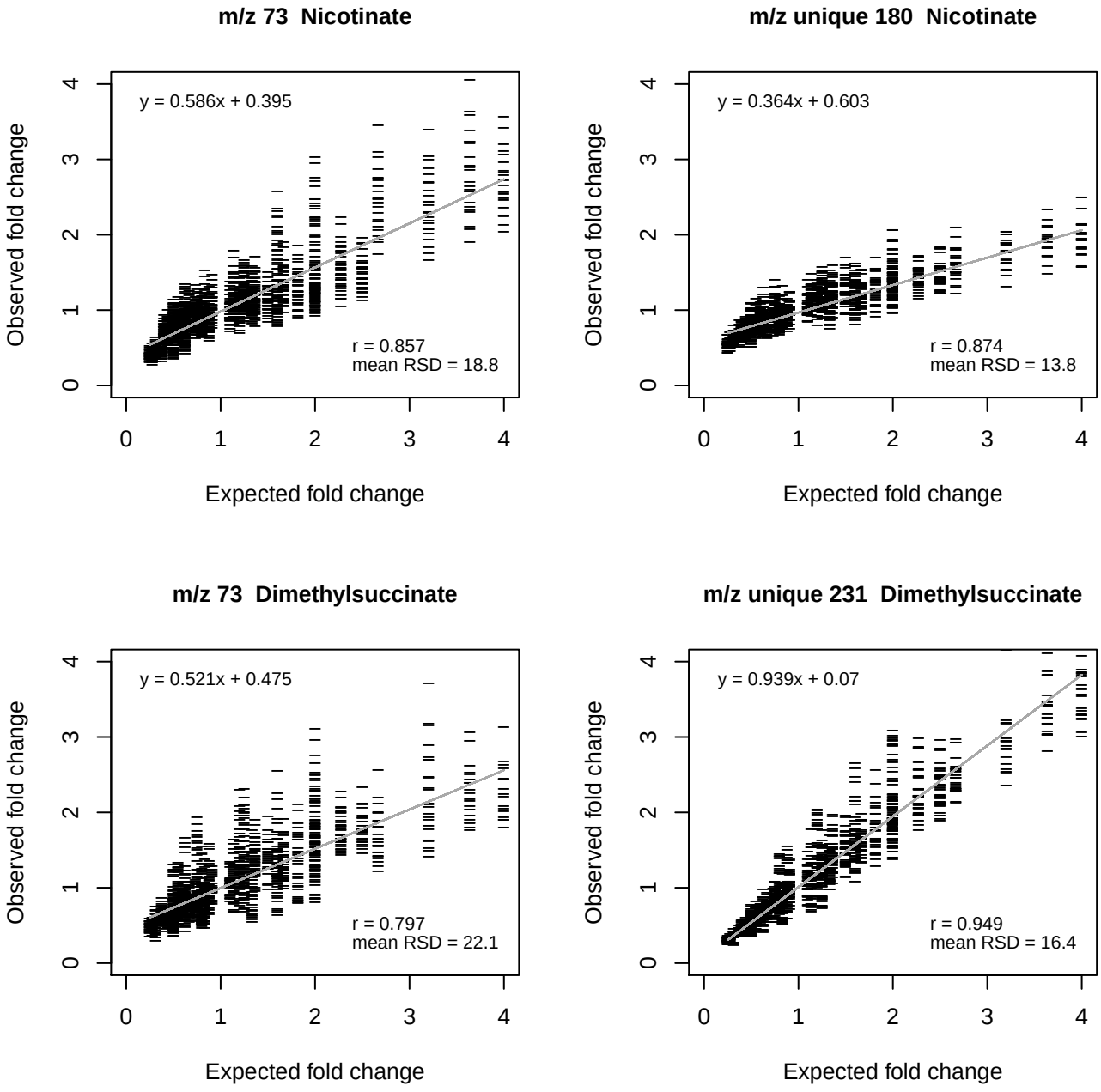

m/z 73 2−Hydroxy−3−methylbutyrate
y = 0.288x + 0.678
r = 0.878
mean RSD = 10.3
Observed fold change
Expected fold change
0
1
2
3
4

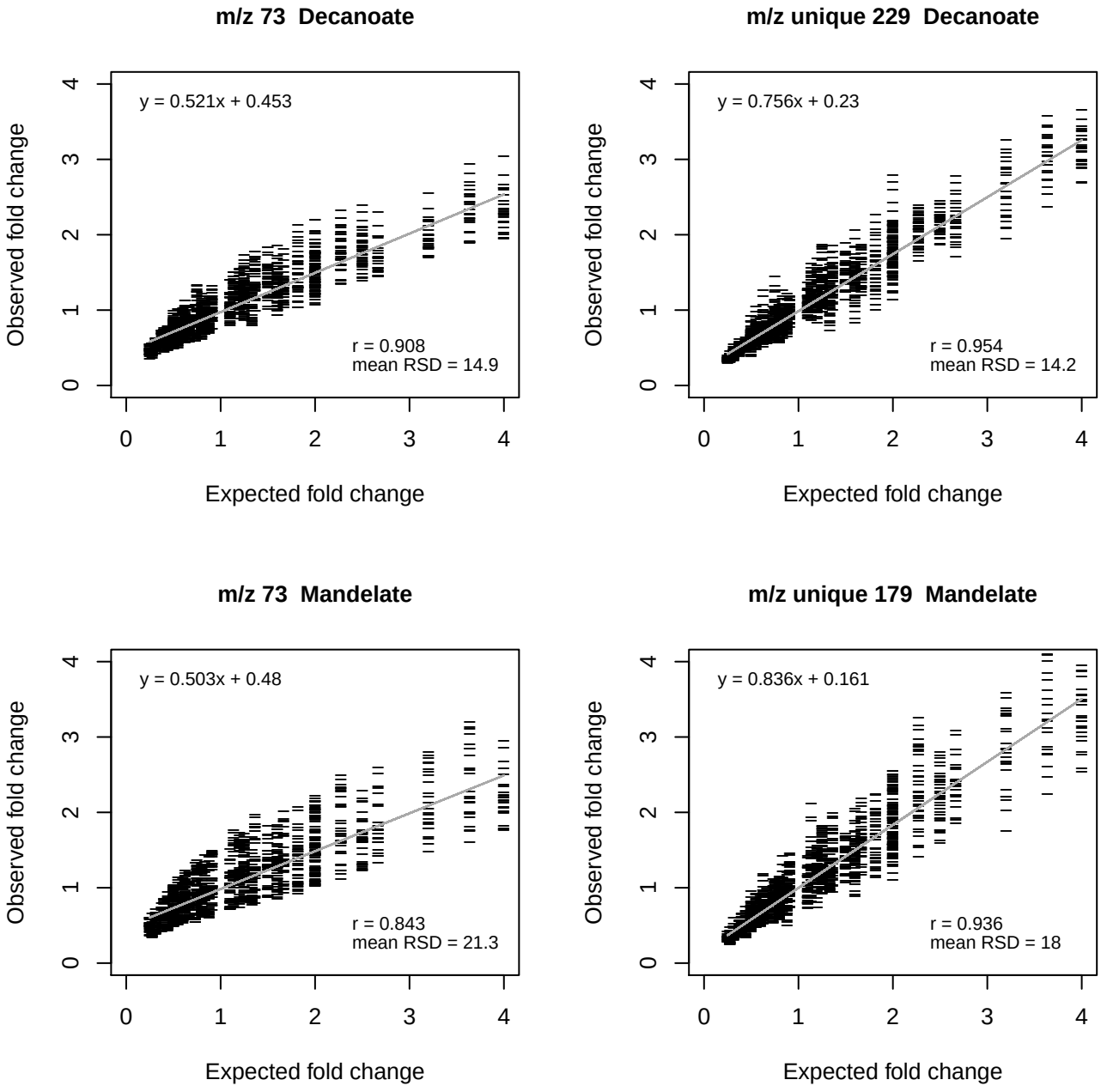

m/z unique 145 2−Hydroxy−3−methylbutyrate
y = 0.398x + 0.562
r = 0.932
mean RSD = 9.7
Observed fold change
Expected fold change
0
1
2
3
4

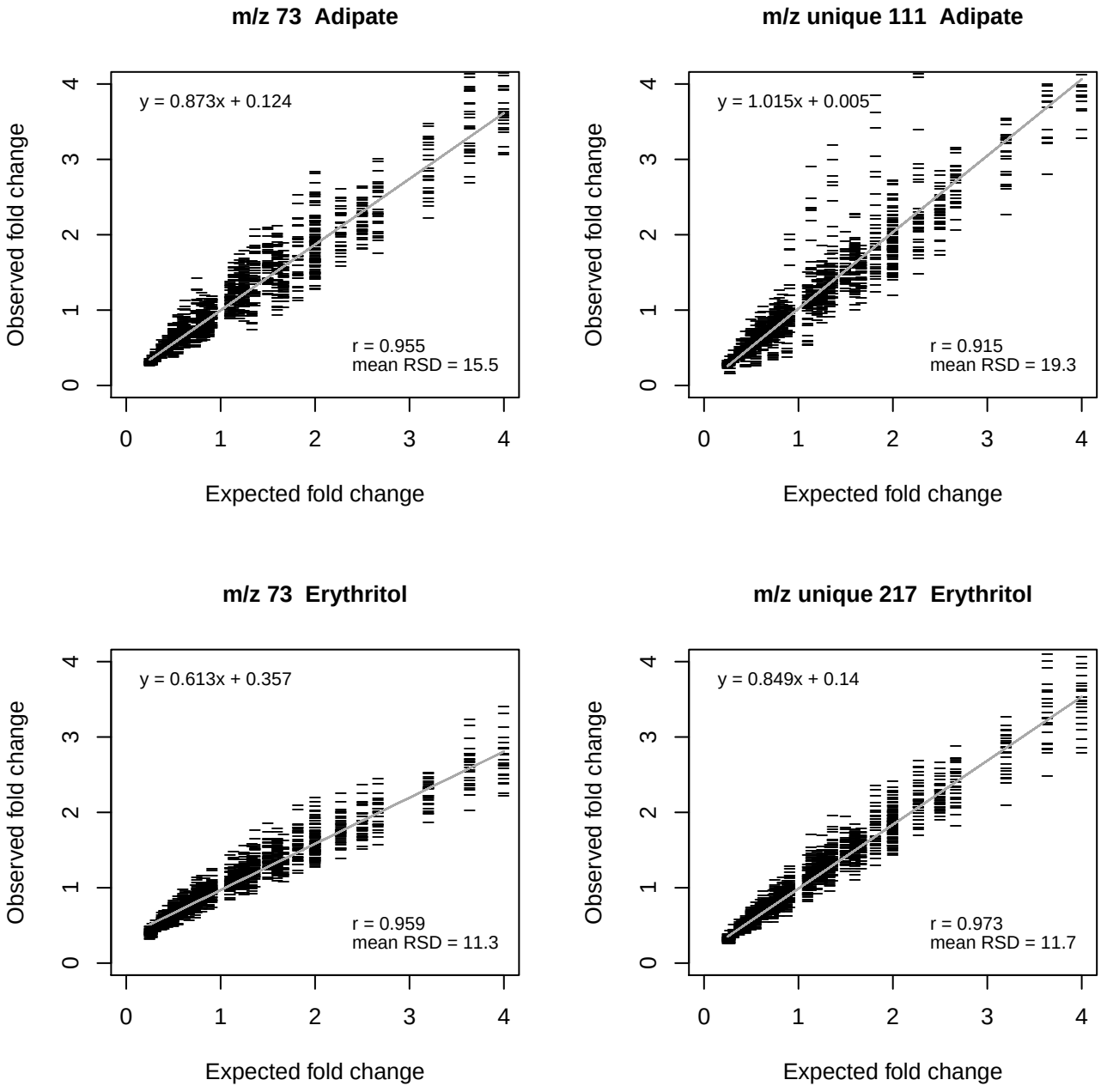

m/z 73 3−Methyl−2−oxovalerate
y = 0.337x + 0.626
r = 0.914
mean RSD = 9.3
Observed fold change
Expected fold change
0
1
2
3
4

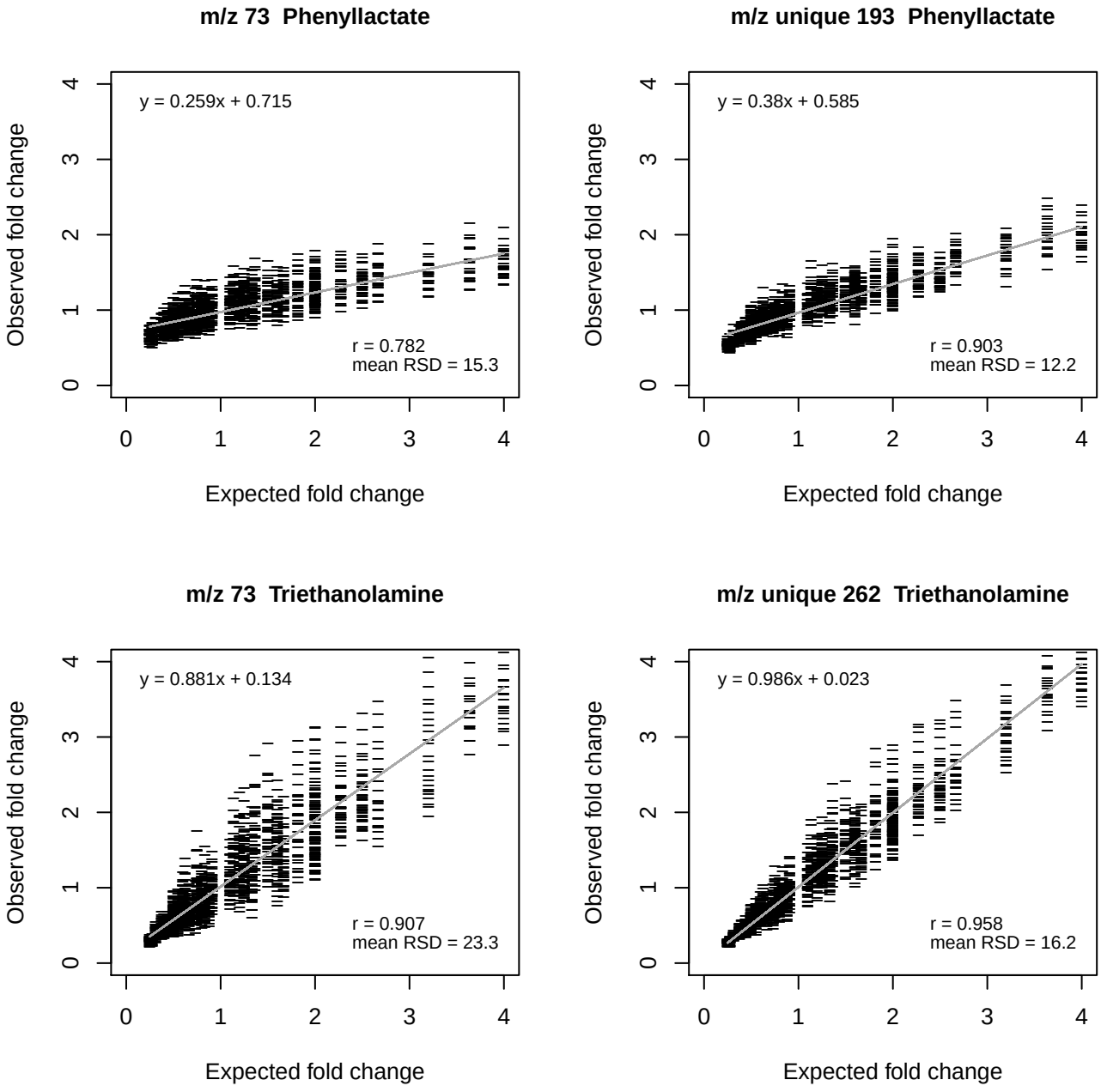

m/z unique 203 3−Methyl−2−oxovalerate
y = 0.45x + 0.508
r = 0.975
mean RSD = 5
Observed fold change
Expected fold change
0
1
2
3
4

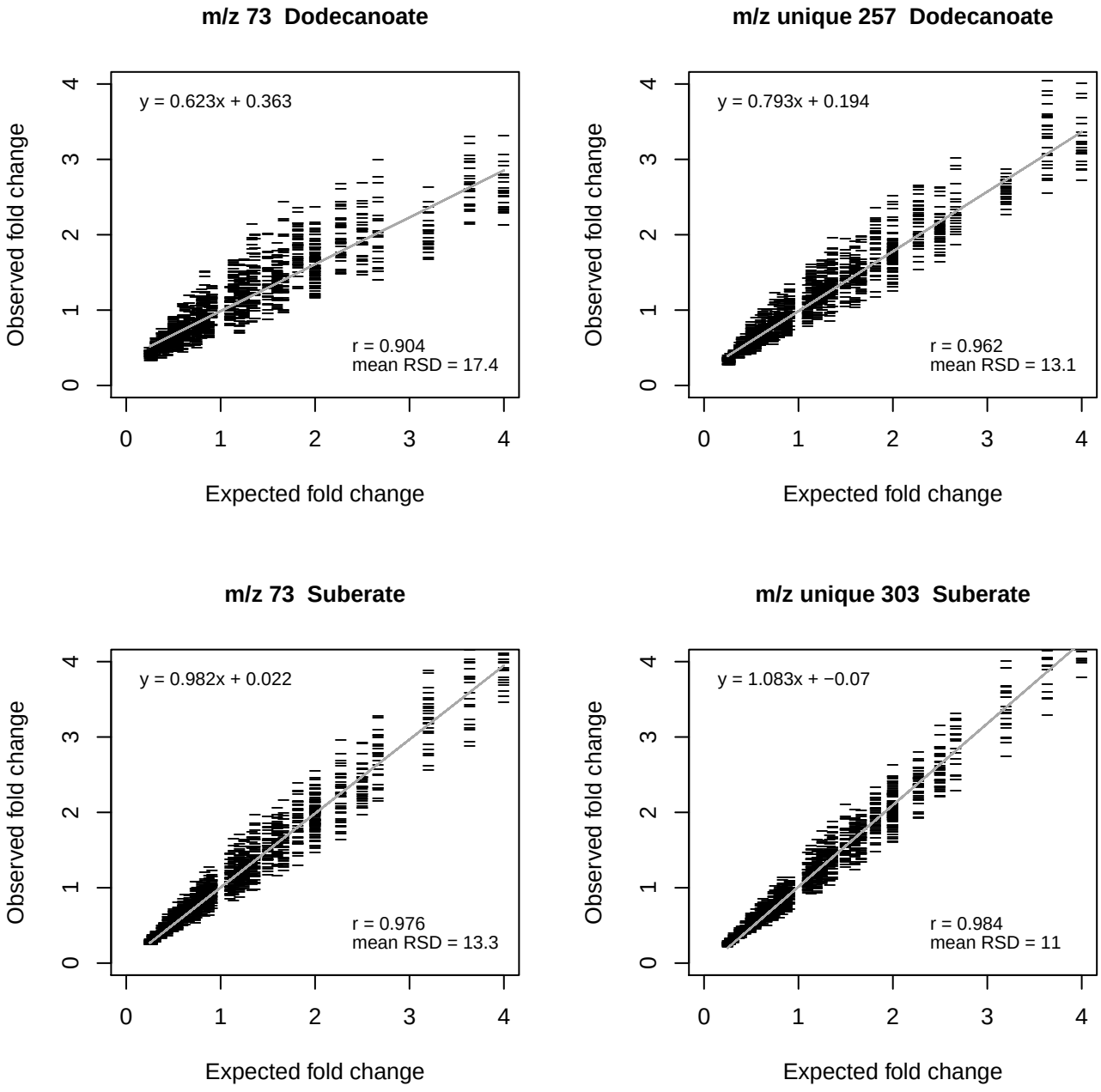

m/z 73 Malonate
y = 0.798x + 0.195
r = 0.934
mean RSD = 17.7
m/z unique 233 Malonate
y = 1.074x + −0.063
r = 0.993
mean RSD = 6.7
m/z 73 Phenylacetate
y = 0.977x + 0.127
r = 0.63
mean RSD = 52.9
m/z unique 193 Phenylacetate
y = 0.621x + 0.352
r = 0.933
mean RSD = 14.9
Observed fold change
Expected fold change
0
1
2
3
4

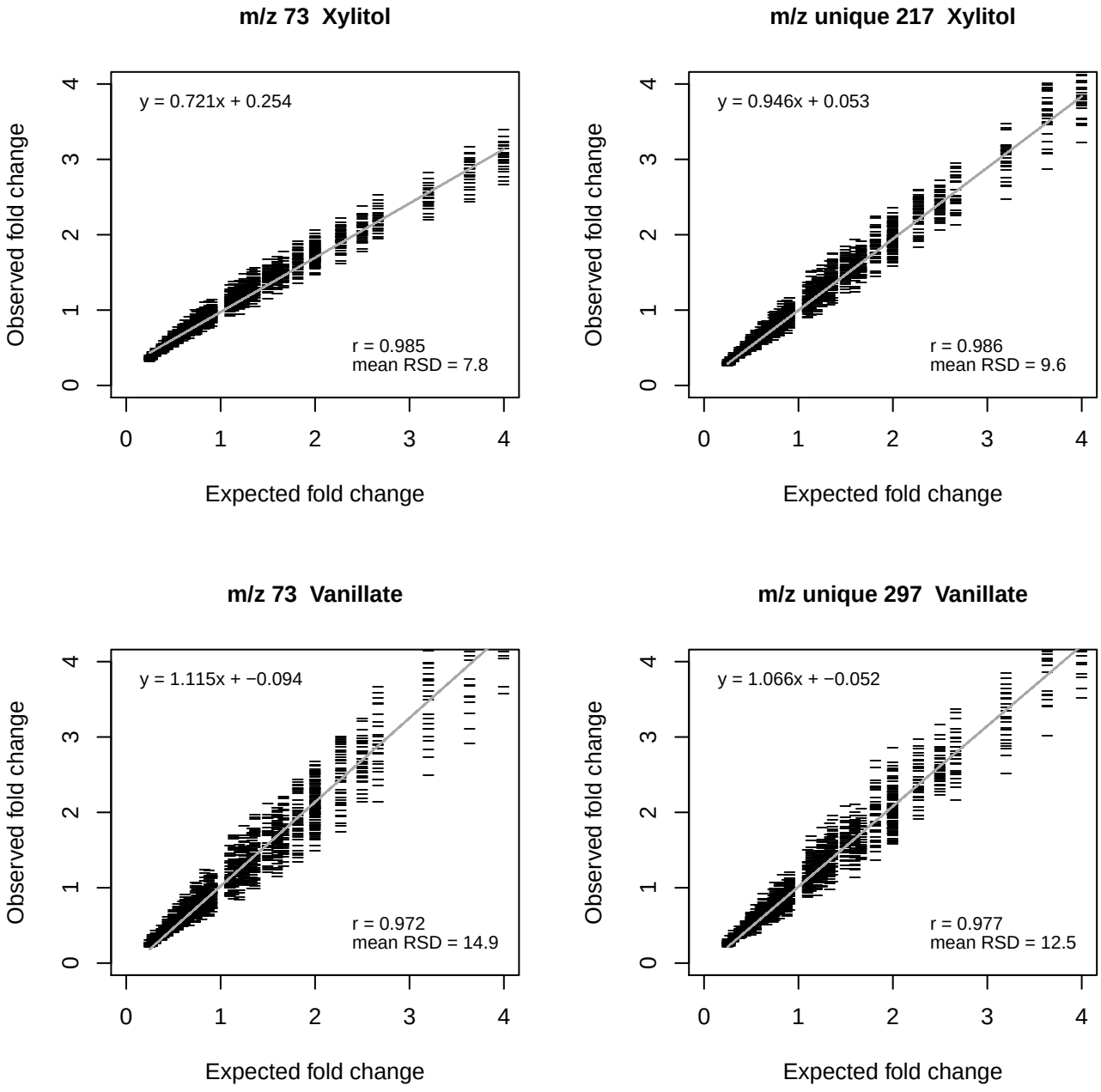

m/z 73 Nicotinate
y = 0.586x + 0.395
r = 0.857
mean RSD = 18.8
m/z unique 180 Nicotinate
y = 0.364x + 0.603
r = 0.874
mean RSD = 13.8
m/z 73 Dimethylsuccinate
y = 0.521x + 0.475
r = 0.797
mean RSD = 22.1
m/z unique 231 Dimethylsuccinate
y = 0.939x + 0.07
r = 0.949
mean RSD = 16.4
Observed fold change
Expected fold change
0
1
2
3
4

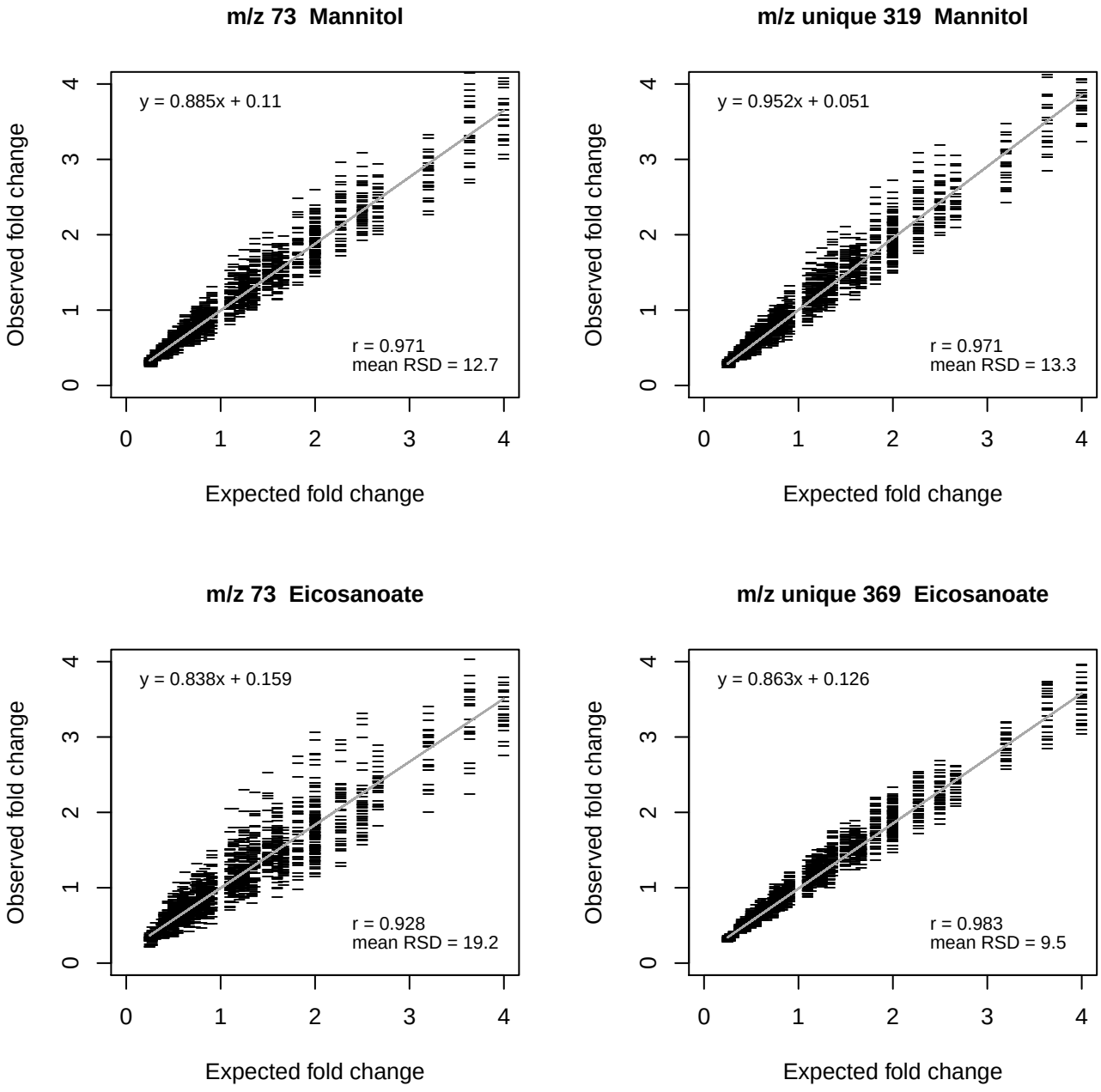

m/z 73 Decanoate
y = 0.521x + 0.453
r = 0.908
mean RSD = 14.9
m/z unique 229 Decanoate
y = 0.756x + 0.23
r = 0.954
mean RSD = 14.2
m/z 73 Mandelate
y = 0.503x + 0.48
r = 0.843
mean RSD = 21.3
m/z unique 179 Mandelate
y = 0.836x + 0.161
r = 0.936
mean RSD = 18
Observed fold change
Expected fold change
0
1
2
3
4


m/z 73 Adipate
y = 0.873x + 0.124
r = 0.955
mean RSD = 15.5
m/z unique 111 Adipate
y = 1.015x + 0.005
r = 0.915
mean RSD = 19.3
m/z 73 Erythritol
y = 0.613x + 0.357
r = 0.959
mean RSD = 11.3
m/z unique 217 Erythritol
y = 0.849x + 0.14
r = 0.973
mean RSD = 11.7
Observed fold change
Expected fold change
0
1
2
3
4


m/z 73 Phenyllactate
y = 0.259x + 0.715
r = 0.782
mean RSD = 15.3
m/z unique 193 Phenyllactate
y = 0.38x + 0.585
r = 0.903
mean RSD = 12.2
m/z 73 Triethanolamine
y = 0.881x + 0.134
r = 0.907
mean RSD = 23.3
m/z unique 262 Triethanolamine
y = 0.986x + 0.023
r = 0.958
mean RSD = 16.2
Observed fold change
Expected fold change
0
1
2
3
4


m/z 73 Dodecanoate
y = 0.623x + 0.363
r = 0.904
mean RSD = 17.4
m/z unique 257 Dodecanoate
y = 0.793x + 0.194
r = 0.962
mean RSD = 13.1
m/z 73 Suberate
y = 0.982x + 0.022
r = 0.976
mean RSD = 13.3
m/z unique 303 Suberate
y = 1.083x + −0.07
r = 0.984
mean RSD = 11
Observed fold change
Expected fold change
0
1
2
3
4


m/z 73 Xylitol
y = 0.721x + 0.254
r = 0.985
mean RSD = 7.8
m/z unique 217 Xylitol
y = 0.946x + 0.053
r = 0.986
mean RSD = 9.6
m/z 73 Vanillate
y = 1.115x + −0.094
r = 0.972
mean RSD = 14.9
m/z unique 297 Vanillate
y = 1.066x + −0.052
r = 0.977
mean RSD = 12.5
Observed fold change
Expected fold change
0
1
2
3
4


m/z 73 Mannitol
y = 0.885x + 0.11
r = 0.971
mean RSD = 12.7
Observed fold change
Expected fold change
m/z unique 319 Mannitol
y = 0.952x + 0.051
r = 0.971
mean RSD = 13.3
Observed fold change
Expected fold change
m/z 73 Eicosanoate
y = 0.838x + 0.159
r = 0.928
mean RSD = 19.2
Observed fold change
Expected fold change
m/z unique 369 Eicosanoate
y = 0.863x + 0.126
r = 0.983
mean RSD = 9.5
Observed fold change
Expected fold change

A.3 Average classification accuracies of the outer cross-validation loop

A.3.1 Average classification accuracies on training sets for linear scores

Group comparison		Average classification accuracy					
		NB	CB	BReg	Bin	Smooth	Adapt
1	2	76.9 ± 0.9	76.5 ± 0.9	76.9 ± 0.7	77.6 ± 2.6	78.8 ± 1.2	78.7 ± 1.3
1	3	96.9 ± 0.7	95.7 ± 0.7	93.0 ± 0.6	96.5 ± 1.5	96.8 ± 1.3	97.0 ± 1.2
1	4	84.4 ± 1.1	84.9 ± 1.1	89.1 ± 1.0	89.5 ± 1.7	89.6 ± 1.0	89.7 ± 1.1
1	5	83.4 ± 2.1	86.8 ± 0.8	88.3 ± 1.1	87.4 ± 3.2	88.9 ± 1.3	88.4 ± 1.5
1	2,3,4,5	77.1 ± 0.8	78.5 ± 0.4	78.3 ± 0.6	78.1 ± 1.5	78.1 ± 0.5	78.4 ± 0.4
1A	2	86.4 ± 1.2	86.2 ± 1.1	86.5 ± 0.8	85.9 ± 2.1	85.9 ± 1.0	85.9 ± 0.9
1B	2	63.7 ± 2.9	61.2 ± 6.0	71.7 ± 2.9	72.6 ± 4.7	74.0 ± 3.1	72.8 ± 3.6

Table A.1: Average classification accuracies of the outer cross-validation loop on training sets for the six probability estimation methods, Naive Bayes (NB), Compound Bayes (CB), binary regression (BReg) and the local error frequency methods using binning (Bin) and smoothing (Smooth/ Adapt). For each patient group comparison (rows) performances are listed for each estimation method. The assignment of patient groups to the indices can be found in Table 4.1. Since within the outer cross-validation loop 2 samples are left out as test set, there are 32 to 50 training sets for the different group comparisons. Thus, classification accuracies are given as average plus/minus standard deviations from the average.

A.3.2 Average classification accuracies on training and test sets for SVM scores

Group comparison		Average classification accuracy					
		NB	CB	BReg	Bin	Smooth	Adapt
1	2	82.4 ± 2.2	81.9 ± 1.9	82.4 ± 2.2	81.0 ± 3.4	82.3 ± 2.7	82.8 ± 2.0
1	3	7.1 ± 0.5	97.9 ± 1.5	99.2 ± 1.1	77.1 ± 0.6	77.1 ± 0.6	77.1 ± 0.6
1	4	94.5 ± 1.6	92.2 ± 2.5	94.4 ± 2.1	93.9 ± 2.7	94.3 ± 2.2	94.4 ± 2.1
1	5	96.3 ± 1.8	95.8 ± 1.4	96.0 ± 2.3	95.4 ± 2.5	96.4 ± 2.0	96.3 ± 2.0
1	2,3,4,5	84.0 ± 1.8	82.9 ± 1.8	84.4 ± 1.7	84.2 ± 2.0	84.7 ± 1.6	84.8 ± 1.7
1A	2	93.3 ± 1.7	93.0 ± 1.9	92.9 ± 1.9	92.6 ± 2.8	92.8 ± 2.1	93.0 ± 2.2
1B	2	80.3 ± 2.9	73.6 ± 5.7	80.1 ± 3.3	81.2 ± 2.3	81.1 ± 2.6	80.6 ± 2.7

Table A.2: Average classification accuracies of the outer cross-validation loop on training sets for the six probability estimation methods based on SVM scores. For each patient group comparison (rows) performances are listed for each estimation method. The assignment of patient groups to the indices can be found in Table 4.1. Since within the outer cross-validation loop 2 samples are left out as test set, there are 32 to 50 training sets for the different group comparisons. Thus, classification accuracies are given as average plus/minus standard deviations from the average.

Group comparison		Classification performance					
		NB	CB	BReg	Bin	Smooth	Adapt
1	2	78.0	80.0	80.0	78.0	77.0	79.0
1	3	7.1	97.1	97.1	75.7	75.7	75.7
1	4	94.0	89.3	91.7	91.7	91.7	86.9
1	5	90.8	92.1	94.7	90.8	89.5	90.8
1	2,3,4,5	78.6	77.4	81.0	76.8	79.8	75.6
1A	2	88.9	90.0	88.9	85.1	88.9	86.4
1B	2	72.3	69.2	80.0	73.8	70.8	70.8

Table A.3: Classification performances of the outer cross-validation loop for the six probability estimation methods based on SVM scores. For each patient group comparison (rows) performances are listed for each estimation method. The assignment of patient groups to the indices can be found in Table 4.1.

Bibliography

[1] M. Adhchour, J. Beens, and U. A. T. Brinkman. Recent developments in the application of comprehensive two-dimensional gas chromatography. *J Chromatogr A*, 1186:67–108.

[2] M. Almstetter, K. Dettmer, and P. Oefner. Comprehensive two-dimensional gas chromatography in metabolomics studies. *Anal Bioanal Chem*, in revision, 2011.

[3] M. F. Almstetter, I. J. Appel, K. Dettmer, M. A. Gruber, and P. Oefner. Comparison of two algorithmic data processing strategies for metabolic fingerprinting by comprehensive two-dimensional gas chromatography-time-of-flight mass spectrometry. *J Chromatogr A*, 1218:7031–7038, 2011.

[4] M. F. Almstetter, I. J. Appel, M. A. Gruber, C. Lottaz, B. Timischl, R. Spang, K. Dettmer, and P. J. Oefner. Integrative normalization and comparative analysis for metabolic fingerprinting by comprehensive two-dimensional gas chromatography-time-of-flight mass spectrometry. *Anal Chem*, 81(14):5731–5739, 2009. PMID: 19522528.

[5] C. Ambroise and G. McLachlan. Selection bias in gene extraction on the basis of microarray gene-expression data. *Proc Natl Acad Sci U S A*, 99(10):6562–6, May 2002.

[6] J. M. Amigo, T. Skov, and R. Bro. Chromathography: Solving chromatographic issues with mathematical models and intuitive graphics. *Chem Rev*, 110(8):4582–4605, 2010.

[7] I. J. Appel, W. Gronwald, and R. Spang. Estimating classification probabilities in high-dimensional diagnostic studies. *Bioinformatics*, 27:2563–2570, 2011.

[8] M. Ayer, H. D. Brunk, G. M. Ewing, W. T. Reid, and E. Silverman. An empirical distribution function for sampling with incomplete information. *Ann Math Stat*, 5(26):641–647, 1955.

[9] W. Bertsch. Two-dimensional gas chromatography. Concepts, Instrumentation, and Applications part 1: Fundamentals, conventional two-dimensional gas chromatography, selected applications. *J High Res Chromatog*, 22(12):647–665, 1999.

[10] H.-J. Boehm, M. Boehringer, D. Bur, H. Gmuender, W. Huber, W. Klaus, D. Kostrewa, H. Kuehne, T. Luebbers, N. Meunier-Keller, and F. Mueller. Novel inhibitors of DNA gyrase: 3D structure based biased needle screening, hit validation by biophysical methods, and 3D guided optimzation. a promising alternative to random screening. *J Med Chem*, 43(14):2664–2674, 2000.

[11] S. Castillo, I. Mattila, J. Miettinen, M. Oresic, and T. Hyotylainen. Data analysis tool for comprehensive two-dimensional gas chromatography/time-of-flight mass spectrometry. *Anal Chem*, 83:3058–3067, 2011.

[12] M. Cheok, W. Yang, C. Pui, J. Downing, C. Cheng, C. Naeve, M. Relling, and W. Evans. Treatment-specific changes in gene expression discriminate in vivo drug response in human leukemia cells. *Nat Genet*, 34(1):85–90, May 2003.

[13] P. Clarke, R. te Poele, R. Wooster, and W. P. Gene expression microarray analysis in cancer biology, pharmacology, and drug development: progress and potential. *Biochem Pharmacol*, 62(10):1311–1336, 2001.

[14] M. Consortium. The microarray quality control (MAQC)-II study of common practices for the development and validation of microarray-based predictive models. *Nat Biotechnol*, 28:827–838, 2010.

[15] H. J. Cortes, B. Winniford, J. Loung, and M. Pursch. Comprehensive two dimensional gas chromatography review. *J Sep Sci*, 32:883–904, 2009.

[16] J. Dalluege, J. Beens, and U. A. T. Brinkman. Comprehensive two-dimensional gas chromatography: a powerful and versatile analytical tool. *J Chromatogr A*, 1000:69–108, 2003.

[17] A. P. Dawid. The well-calibrated Bayesian. *J Am Stat Assoc*, 77:605–610, 1982.

[18] M. L. de Hoon, Y. Makita, S. Imoto, K. Kobayashi, N. Ogasawara, K. Nakai, and S. Miyano. Predicting gene regulation by sigma factors in Bacillus subtilis from genome-wide data. *Bioinformatics*, 20(suppl.1):i101–108, 2004.

[19] K. Dettmer, M. F. Almstetter, I. J. Appel, N. Nürnberger, G. Schlamberger, W. Gronwald, H. H. D. Meyer, and P. J. Oefner. Comparison of serum versus

plasma collection in gas chromatography - mass spectrometry-based metabolomics. *Electrophoresis*, 31:2365–2373, 2010.

[20] K. Dettmer, P. A. Aronov, and B. D. Hammock. Mass spectrometry-based metabolomics. *Mass Spectrom Rev*, 26(1):51–78, 2007.

[21] K. Dettmer and B. D. Hammock. Metabolomics – a new exciting field within the "omics" sciences. *Environ Health Perspect*, 112:396–397, 2004.

[22] S. Dudoit, J. Fridlyand, and T. Speed. Comparison of discrimination methods for the classification of tumors using gene expression data. *J Amer Stat Assoc*, 97:77–87, 2002.

[23] X. Fan, L. Shi, H. Fang, Y. Cheng, R. Perkins, and W. Tong. DNA microarrays are predictive of cancer prognosis: A re-evaluation. *Clin Cancer Res*, 16:629–636, 2010.

[24] C. G. Fraga, B. J. Prazen, and R. E. Synovec. Objective data alignment and chemometric analysis of comprehensive two-dimensional separations with run-to-run peak shifting on both dimensions. *Anal Chem*, 73:5833–5840, 2001.

[25] R. G. G G Harrigan. *Metabolic profiling - its role in biomarker discovery and gene function analysis*. Kluwer Academic Publishers Group, second edition, 2004.

[26] H. G. Gika, E. Macpherson, G. A. Theodoridis, and I. D. Wilson. Evaluation of the repeatability of ultra-performance liquid chromatography-TOF-MS for global metabolic profiling of human urine samples. *J Chromatogr B*, 871(2):299 – 305, 2008. Hyphenated Techniques for Global Metabolite Profiling.

[27] W. Gronwald, M. S. Klein, R. Zeltner, B.-D. Schulze, S. W. Reinhold, M. Deutschmann, A.-K. Immervoll, C. A. Bger, B. Banas, K.-U. Eckardt, and P. J. Oefner. Detection of autosomal dominant polycystic kidney disease by NMR spectroscopic fingerprinting of urine. *Kidney Int*, 79:1244–1253, 2011.

[28] X. Guo and M. E. Lidstrom. Metabolite profiling analysis of methylobacterium extorquens AM1 by comprehensive two-dimensional gas chromatography coupled with time-of-flight mass spectrometry. *Biotechnol Bioeng*, 99(4):929–940, 2008.

[29] T. Haferlach, A. Kohlmann, S. Schnittger, M. Dugas, W. Hiddemann, W. Kern, and C. Schoch. A global approach to the diagnosis of leukemia using gene expression profiling. *Blood*, 106(4):1189–98, May 2005.

[30] J. Halket and V. Zaikin. Derivatization in mass spectrometry. *Eur J Mass Spectrom (Chichester, Eng)*, 9:1–21, 2003.

[31] R. Hall, M. Beale, O. Fiehn, N. Hardy, L. Sumner, and R. Bino. Plant metabolomics: the missing link in functional genomics strategies. *Plant Cell*, 14:1437–1440, 2002.

[32] B. Hanczar and E. Dougherty. Classification with reject option in gene expression data. *Bioinformatics*, Jul 2008.

[33] T. Hastie and R. Tibshirani. Classification by pairwise coupling. *Ann Stat*, 26(2):451–471, 1998.

[34] E. Holmes, R. L. Loo, O. Cloarec, M. Coen, H. Tang, E. Maibaum, S. Bruce, Q. Chan, P. Elliott, J. Stamler, I. D. Wilson, J. C. Lindon, and J. K. Nicholson. Detection of urinary drug metabolite (xenometabolome) signatures in molecular epidemiology studies via statistical total correlation (NMR) spectroscopy. *Anal Chem*, 79(4):2629–2640, 2007.

[35] J. L. Hope, B. J. Prazen, E. J. Nilsson, M. E. Lidstrom, and R. E. Synovec. Comprehensive two-dimensional gas chromatography time-of-flight mass spectrometry detection: analysis of amino acid and organic acid trimethylsilyl derivatives, with application to the analysis of metabolites in rye grass samples. *Anal Chem*, 65:380–388, 2005.

[36] X. Huang and F. E. Regnier. Differential metabolomics using stable isotope labeling and two-dimensional gas chromatography with time-of-flight mass spectrometry. *Anal Chem*, 80:107–114, 2008.

[37] J. Iqbal, W. Sanger, D. Horsman, A. Rosenwald, D. Pickering, B. Dave, S. Dave, L. Xiao, K. Cao, Q. Zhu, S. Sherman, C. Hans, D. Weisenburger, T. Greiner, R. Gascoyne, G. Ott, H. Müller-Hermelink, J. Delabie, R. Braziel, E. Jaffe, E. Campo, J. Lynch, J. Connors, J. Vose, J. Armitage, T. Grogan, L. Staudt, and W. Chan. BCL2 translocation defines a unique tumor subset within the germinal center B-cell-like diffuse large B-cell lymphoma. *Am J Pathol*, 165(1):159–66, Jul 2004.

[38] W. J. Lemon, S. Liyanarachchi, and M. You. A high performance test of differential gene expression for oligonucleotide arrays. *Genome Biol*, 4(R67), 2003.

[39] J. C. Lindon and J. K. Nicholson. Analytical technologies for metabonomics and metabolomics, and multi-omic information recovery. *Trends Analyt Chem*, 27(3):194–204, 2008.

[40] C. Lottaz, D. Kostka, F. Markowetz, and R. Spang. Computational diagnostics with gene expression profiles. In J. M. Keith, editor, *Bioinformatics*, volume 453 of *Methods in Molecular Biology*, pages 281–296. Humana Press, 2008.

[41] M. J. Marton, J. L. DeRisi, H. A. Bennett, V. R. Iyer, M. R. Meyer, C. J. Roberts, R. Stoughton, J. Burchard, D. Slade, H. Dai, D. E. J. Bassett, L. H. Hartwell, O. P. Brown, and F. S. H. Drug target validation and identification of secondary drug target effects using DNA microarrays. *Nat Med*, 4(11):1293–1301, 1998.

[42] S. Michiels, S. Koscielny, and C. Hill. Interpretation of microarray data in cancer. *Brit J Cancer*, 96:1155–1158, 2007.

[43] R. E. Mohler, K. M. Dombek, J. C. Hoggard, K. M. Pierce, E. T. Young, and R. E. Synovec. Comprehensive analysis of yeast metabolite GC×GC-TOFMS data: combining discovery-mode and deconvolution chemometric software. *Analyst*, 132:756–767, 2007.

[44] R. E. Mohler, K. M. Dombek, J. C. Hoggard, E. T. Young, and R. E. Synovec. Comprehensive two-dimensional gas chromatography time-of-flight mass spectrometry analysis of metabolites in fermenting and respiring yeast cells. *Anal Chem*, 78:2700–2709, 2006.

[45] S. Mukherjee. *A Practical Approach to Microarray Data Analysis*, chapter Classifying microarray data using support vector machines, pages 166–185. Springer US, 2003.

[46] M. Nagata, J. Fujita, H. Ida, H. Hoshina, T. Inoue, Y. Seki, M. Ohnishi, T. Ohyama, S. Shingaki, M. Kaji, T. Saku, and R. Takagi. Identification of potentioal biomarkers of lymph node metastasis in oral squamous cell carcinoma by cDNA microarray analysis. *Int J Cancer*, 106:683–689, 2003.

[47] A. Niculescu-Mizil and R. Caruana. Predicting good probabilities with supervised learning. In *ICML '05: Proceedings of the 22nd international conference on Machine learning*, pages 625–632, New York, NY, USA, 2005. ACM.

[48] I. Nobeli, H. Postingl, E. Krissinel, and J. Thornton. A structure-based anatomy of the E.coli metabolome. *J Mol Biol*, 334:697–719, 2011.

[49] C. Oh, X. Huang, F. E. Regnier, C. Buck, and X. Zhang. Comprehensive two-dimensional gas chromatography/time-of-flight mass spectrometry peak sorting algorithm. *J Chromatogr A*, 1179:205–215, 2008.

[50] K. Oksman-Caldentey and K. Saito. Integrating genomics and metabolomics for engineering plant metabolic pathways. *Curr Opin Biotechnol*, 16:174–179, 2005.

[51] H. M. Parsons, C. Ludwig, U. L. Gnther, and M. R. Viant. Improved classification accuracy in 1- and 2-dimensional nmr metabolomics data using the variance stabilising generalised logarithm transformation. *BMC Bioinformatics*, 8:234, 2007.

[52] K. M. Pierce, L. F. Wood, B. W. Wright, and R. E. Synovec. A comprehensive two-dimensional retention time alignment algorithm to enhance chemometric analysis of comprehensive two-dimensional separation data. *Anal Chem*, 77:7735–7743, 2005.

[53] J. C. Platt. chapter Probabilistic outputs for support vector machines and comparisons to regularized likelihood methods. MIT Press, 2000.

[54] K. Pollard, S. Dudoit, and M. van der Laan. MULTTEST multiple testing procedures and applications to genomics. Division of Biostatistics, University of California, Berkeley, 2004.

[55] Y. Saeys, I. Inza, and P. Larranaga. A review of feature selection techniques in bioinformatics. *Bioinformatics*, 23(19):2507–2517, 2007.

[56] H.-G. Schmarr and J. Bernhardt. Profiling analysis of volatile compounds from fruits using comprehensive two-dimensional gas chromatography and image processing techniques. *J Chromatogr A*, 1217:565–574, 2010.

[57] R. A. Shellie, W. Welthagen, J. Zrostlikova, J. Spranger, M. Ristow, O. Fiehn, and R. Zimmermann. Statistical methods for comparing comprehensive two-dimensional gas chromatography - time-of-flight mass spectrometry results: Metabolomic analysis of mouse tissue extracts. *J Chromatogr A*, 1086:83–90, 2005.

[58] C. Sotiriou and M. J. Piccart. Taking gene-expression profiling to the clinic: when will molecular signatures become relevant to patient care? *Nat Rev Cancer*, 7:545–553, July 2007.

[59] A. Statnikov, C. Aliferis, I. Tsamardinos, D. Hardin, and S. Levy. A comprehensive evaluation of multicategory classification methods for microarray gene expression cancer diagnosis. *Bioinformatics*, 21(5):631–43, Mar 2005.

[60] R. Tibshirani, T. Hastie, B. Narasimhan, and G. Chu. Diagnosis of multiple cancer types by shrunken centroids of gene expression. *Proc Natl Acad Sci U S A*, 99(10):6567–72, May 2002.

[61] B. Timischl, K. Dettmer, H. Kaspar, M. Thieme, and P. J. Oefner. Development of a quantitative, validated capillary electrophoresis-time of flight - mass spectrometry method with integrated high-confidence analyte identification for metabolomics. *Electrophoresis*, 29(10):2203–2214, 2008.

[62] V. G. van Mispelaar, A. C. Tas, A. K. Smilde, P. J. Schoenmakers, and A. C. van Asten. Quantitative analysis of target components by comprehensive two-dimensional gas chromatography. *J Chromatogr A*, 1019:15–29, 2003.

[63] L. van 't Veer, H. Dai, M. van de Vijver, Y. He, A. Hart, M. Mao, H. Peterse, K. van der Kooy, M. Marton, A. Witteveen, G. Schreiber, R. Kerkhoven, C. Roberts, P. Linsley, R. Bernards, and S. Friend. Gene expression profiling predicts clinical outcome of breast cancer. *Nature*, 415(6871):530–6, Jan 2002.

[64] A. R. Venkitaraman. Cancer susceptibility and the functions of BRCA1 and BRCA2. *Cell*, 108:171–182, 2002.

[65] B. Wang, A. Fang, J. Heim, B. Bogdanov, S. Pugh, M. Libardoni, and X. Zhang. DISCO: Distance and spectrum correlation optimization alignment for two-dimensional gas chromatography time-of-flight mass spectrometry-based metabolomics. *Anal Chem*, 82(12):5069–5081, 2010. PMID: 20476746.

[66] L. Wessels, M. Reinders, A. Hart, C. Veenman, H. Dai, Y. He, and L. Veer. A protocol for building and evaluating predictors of disease state based on microarray data. *Bioinformatics*, 21(19):3755–62, Oct 2005.

[67] M. West, C. Blanchette, H. Dressman, E. Huang, S. Ishida, R. Spang, H. Zuzan, J. Olson, J. Marks, and J. Nevins. Predicting the clinical status of human

breast cancer by using gene expression profiles. *Proc Natl Acad Sci U S A*, 98(20):11462–7, Sep 2001.

[68] D. Wishart, C. Knox, A. Guo, R. Eisner, N. Young, B. Gautam, D. Hau, N. Psychogios, E. Dong, S. Bouatra, R. Mandal, I. Sinelnikov, J. Xia, L. Jia, J. Cruz, E. Lim, C. Sobsey, S. Shrivastava, P. Huang, P. Lui, L. Fang, J. Peng, R. Fradette, D. Cheng, D. Tzur, M. Clements, A. Lewis, A. D. Souza, A. Zuniga, M. Dawe, Y. Xiong, D. Clive, R. Greiner, A. Nazyrova, R. Shaykhutdinov, L. Li, H. Vogel, and I. Forsythe. HMDB: a knowledgebase for the human metabolome. *Nucleic Acids Res*, 37:D603–610, 2009.

[69] D. S. Wishart. Quantitative metabolomics using NMR. *Trends Analyt Chem*, 27(3):228–237, 2008.

[70] G. Wright, B. Tan, A. Rosenwald, E. H. Hurt, A. Wiestner, and L. M. Staudt. A gene expression-based method to diagnose clinically distinct subgroups of diffuse large B cell lymphoma. *Proc Natl Acad Sci U S A*, 100(17):9991–6, Aug 2003.

[71] B. Zadrozny and C. Elkan. Transforming classifier scores into accurate multiclass probability estimates. In *SIGKDD'02*, pages 694–699, 2002.

[72] M. Zervakis, M. Blazadonakis, G. Tsiliki, V. Danilatou, M. Tsiknakis, and D. Kafetzopoulos. Outcome prediction based on microarray analysis: a critical perspective on methods. *BMC Bioinformatics*, 10(53):53, Feb 2009.

[73] D. Zhang, X. Huang, F. E. Regnier, and M. Zhang. Two-dimensional correlation optimized warping algorithm for aligning GC×GC-MS data. *Anal Chem*, 80:2664–2671, 2008.

Curriculum Vitae

Inka Appel
Galgenbergstraße 11d
93053 Regensburg
Tel: (0941) 30092890
Inka.Appel@item.fraunhofer.de

geboren am 28. Juli 1983
Geburtsort: Dachau
Staatsangehörigkeit: Deutsch
Familienstand: ledig

Akademischer Grad

Diplom-Bioinformatikerin

	Mai 2007
an der	Ludwig Maximilians Universität und Technische Universität München
Betreuer:	Prof. Dr. W. Mewes
Arbeit:	Duplizierte Netzwerke in Pflanzengenomen

Ausbildung

seit 10/2011	Wissenschaftl. Mitarbeiter am Fraunhofer ITEM Regensburg, Projektgruppe *Personalisierte Tumortherapie*
06/2007–09/2011	Doktorand in der Gruppe *Computational Diagnostics* des *Instituts für Funktionelle Genomik* der Universität Regensburg
10/2002–05/2007	Studium der Bioinformatik, Ludwig Maximilian Universität und Technische Universität München
05/2002	Abitur, Gymnasium Weilheim i. OB

Auszeichnungen

Glanzlichter Biomedizinischer Forschung 2009 der Universität Regensburg für die Publikation: Almstetter MF*, Appel IJ*, Gruber MA, Lottaz C, Timischl B, Spang R, Dettmer K, Oefner PJ (2009). *Integrative normalization and comparative analysis for metabolic fingerprinting by comprehensive two-dimensional gas chromatography - time-of-flight mass spectrometry.* Anal Chem. 2009; 81(14): 5731-9. * ebenbürtig zu dieser Arbeit beigetragen

Publications

- **I. J. Appel**, W. Gronwald, R. Spang. Estimating classification probabilities in high-dimensional diagnostic studies. *Bioinformatics*, 27(18):2563–2570, Sept 2011.

- M. F. Almstetter*, **I. J. Appel***, K. Dettmer, M. A. Gruber, P. J. Oefner, * equally contributing. Comparison of two algorithmic data processing strategies for metabolic fingerprinting by comprehensive two-dimensional gas chromatography - time-of-flight mass spectrometry. *J Chromatogr A*, 1218(39):7031–7038, Sept 2011.

- K. Dettmer, M. F. Almstetter, **I. J. Appel**, N. Nürnberger, G. Schlamberger, W. Gronwald, H. H.-D. Meyer, and P. J. Oefner. Comparison of serum *versus* plasma collection in gas chromatography - Mass spectrometry-based metabolomics. *Electrophoresis*, 31:2365–2373, Apr 2010.

- M. F. Almstetter*, **I. J. Appel***, M. A. Gruber, C. Lottaz, B. Timischl, R. Spang, K. Dettmer, and P. J. Oefner, * equally contributing. Integrative Normalization and Comparative Analysis for Metabolic Fingerprinting by Comprehensive Two-Dimensional Gas Chromatography - Time-of-Flight Mass Spectrometry. *Anal Chem*, 81(14):5731–5739, July 2009.

  * contributed equally to this work

Eidesstattliche Erklärung

1. Ich erkläre hiermit an Eides statt, dass ich die vorliegende Arbeit ohne unzulässige Hilfe Dritter und ohne Benutzung anderer als der angegebenen Hilfsmittel angefertigt habe; die aus anderen Quellen direkt oder indirekt übernommenen Daten und Konzepte sind unter Angabe des Literaturzitats gekennzeichnet.

2. Weitere Personen waren an der inhaltlich-materiellen Herstellung der vorliegenden Arbeit nicht beteiligt. Insbesondere habe ich hierfür nicht die entgeltliche Hilfe eines Promotionsberaters oder anderer Personen in Anspruch genommen. Niemand hat von mir weder unmittelbar noch mittelbar geldwerte Leistungen für Arbeiten erhalten, die im Zusammenhang mit dem Inhalt der vorgelegten Dissertation stehen.

3. Die Arbeit wurde bisher weder im In- noch im Ausland in gleicher oder ähnlicher Form einer anderen Prüfungsbehörde vorgelegt.

Regensburg, 24.11.2011 Inka Appel