Computational Methods and Stochastic Models in Proteomics

Bogusław Kluge

Faculty of Mathematics, Informatics and Mechanics

University of Warsaw

A thesis submitted for the degree of

Doctor of Philosophy

March 2011

Advisor: dr hab. Anna Gambin

Author's declaration:
Aware of legal responsibility I hereby declare that I have written this dissertation myself and all the contents of the dissertation have been obtained by legal means.

date Author's signature

Supervisor's declaration:
The dissertation is ready to be reviewed.

date Supervisor's signature

Abstract

The thesis addresses three problems arising from mass spectrometry (MS) data processing. It describes computational methods for solving them and stochastic models that formalize some of them.

The first problem is redundancy elimination in liquid chromatography MS (LC-MS) images of peptides. An algorithm for isotopic envelopes detection based on the sweeping method is presented. It consists of grouping peaks corresponding to different isotopic versions of the same particle kind and automatic determination of the charge of the group. A dynamic programming algorithm is given that proposes amino acid composition for a given weight of a peptide which helps to asses the quality of isotopic envelopes.

Two solutions are presented to the second problem — the problem of LC-MS spectra alignment. The first one estimates retention time shifting and scaling with a Metropolis-Hastings algorithm. The second one uses a two stage clustering approach consisting of a DBSCAN algorithm pass and gaussian mixture model based clustering (estimation is based on the Expectation-Maximization algorithm).

The last problem is inferring peptidase activity from LC-MS data. Firstly, a bayesian model based on the chemical master equation for exopeptidases is presented together with a Metropolis-Hastings algorithm for parameter estimation. The model is tested on synthetic and real datasets. Then an extended version is developed that handles also endopeptidases and integrates knowledge from the MEROPS peptidase database. Parameter estimation involves solving non-linear least squares problem.

Key words: stochastic modeling, proteomics, Metropolis-Hastings algorithm, Expectation-Maximization, liquid chromatography mass spectrometry, mass spectra alignment, peptidase activity, chemical master equation, triangular matrix exponentiation

ACM classification: J.3 Biology and genetics

Acknowledgements

Firstly, I would like to express my gratitude to Anna Gambin — my advisor — for guiding me, showing patience and understanding for my non-scientific passions and continuous encouragement to complete this project. I also thank all my co-authors, for without them this work would not have been possible. I am grateful to all my friends and colleagues whom I work with for creating such great atmosphere. I have really enjoyed the time spent with you. Finally, I thank my family for the support they have been giving me throughout this journey.

The research described here was supported by the Ministry of Science and Higher Education grants PBZ-KBN-088/P04/2003, KBN-3-T11F-021-28, PBZ-MNiI-2/1/2005, N N301 065236, N N206 356036 and the Integrated Regional Operational Programme financed the European Social Fund and the Polish Government. The computational resources were provided by the CoE BioExploratorium project: WKP 1/1.4.3/1/2004/44/44/115.

Contents

Chapter 1

Introduction

According to (RNCOS, 2010) the market for bioinformatics will grow annually by 24% during years 2011–2013. The report indicates that proteomics will become a significant contributor to this growth as a result of the rise of interest in personalized medicine.

One of the technologies that help in understanding proteomics is *mass spectrometry* (MS). It offers the possibility of performing exhaustive analyses of complex mixtures containing thousands of molecules in a single experiment. A typical mass spectrometer ionizes the molecules forming the mixture being analyzed, then separates them according to their mass-to-charge ratio by an electromagnetic field and finally measures their quantity using a detector. Mass spectrometers differ in the implementations of these stages (Fenn et al., 1989; March, 2000; Marshall et al., 1998). Such experiments produce large amounts of data and therefore can only be interpreted with help of a computer running specialized algorithms. This thesis presents solutions for various problems that come along the road from raw mass spectrometry data to biological conclusions.

The results of this thesis are two-fold:

- Computational methods for a set of problems related to mass spectrometry data were developed using the frameworks of Expectation Maximization, Markov chain Monte Carlo Metropolis-Hastings sampling and dynamic programming.

- The biological process of peptide proteolysis was captured into a stochastic model with input from mass spectrometry data and biological databases. It is formalized as a variant of the *chemical master equation*.

As a byproduct also some insights into the triangular matrix exponentiation are presented that can be turned into a dynamic programming algorithm.

Interdisciplinary studies, such as this require interaction of people with different backgrounds ranging from physicians, bioinformaticians and biostatisticians with experience in various methods of data processing and analysis to wet laboratory technicians. While working on my thesis I had a great opportunity to cooperate with leading experimental labs — with the team led by prof. Michał Dadlez from the Institute of Biochemistry and Biophysics of Polish Academy of Sciences and the team led by prof. Jerzy Ostrowski from the Maria Skłodowska-Curie Institute of Oncology. All the MS data used in this thesis come from one of these two labs.

The first part deals with MS data preprocessing which consists of noise reduction and redundancy elimination. A raw spectrum can be thought of as a function $\mathrm{R} \to \mathrm{R}$ (see Fig. 2.1) or $\mathrm{R}^2 \to \mathrm{R}$ (LC-MS, i.e. *liquid chromatography mass spectrum*; see Fig. 2.3) while useful information is the position and height of peaks of this function. The first step is therefore producing a list of peaks (it is often called *peak picking*) and grouping peaks corresponding to the same particles differing in charge or isotopic version. Chapter 2, based on (Gambin et al., 2007), proposes procedures for solving these problems. After an introduction to LC-MS data and dealing with the peak picking problem with the use of NMR (*Nuclear Magnetic Resonance*) software package NMRPipe (Delaglio et al., 1995) our approach to isotopic envelopes detection (i.e. grouping peaks from different isotopic versions of the same particle) is described. The novelty of this approach lies in looking at a 2D spectrum (LC-MS) as a whole. It is a substantial generalization of the THRASH method (Horn et al., 2000) which was designed for 1D data. Besides being involved in the conceptual design of our procedure (a variant of the *sweeping method*) I created a dynamic programming algorithm that proposes amino acid composition for a given weight of a peptide. The method described here was implemented as the *mz2m* software tool available at http://mz2m.sourceforge.net (Krzysztof Kowalczyk is the main author) which was used by the MS laboratory of the Institute of Biochemistry and Biophysics of Polish Academy of Sciences.

Chapter 3 deals with the problem of mass spectra alignment. Formal statements of this problem may vary but the essence is to provide a way to compare two or more mass spectra. In order to do that signals coming from the same particles in different spectra must be identified. It is a crucial step in medical diagnostics applications. Lange et al. (2008) present an overview and comparison of LC-MS alignment procedures. Some approaches bypass the feature extraction stage (i.e. pick peaking) and work with raw LC-MS data by warping the retention time dimension (Bylunda et al., 2002; Prince and Marcotte, 2006), the majority however try to align discrete objects (peaks) (Lange et al., 2007; Li et al., 2005; Smith et al., 2006). The main source of problems is large between-experiment variability of retention times due to imperfections in liquid chromatography technology. Two solutions to the LC-MS alignment problem are presented in this thesis: affine transformations of the retention times and clustering based on gaussian mixtures.

In the first method I use $f_k(t) = a_k t + b_k$ functions indexed by the spectra being aligned to correct the retention times. I define a criterion for optimization over the a and b parameters which favors the transformations where peaks have many neighboring peaks from other spectra. The Metropolis-Hastings algorithm (Hastings, 1970) is used as the optimization procedure. We applied this method in (Kluge et al., 2009), the paper however did not describe the details of the algorithm.

The second method (Gambin et al., 2006; Łuksza et al., 2009) is as far as we know the first application of *model based clustering* to LC-MS proteomic data. Such methodology was previously used with transcriptomic data (Yeung et al., 2001), but we were facing a significantly harder task due to much larger size of the computational problem. We decided to break clustering into two stages. In the first stage we obtain preliminary clusters from the DBSCAN (Ester et al., 1996) algorithm and in the second stage we tackle smaller problems with gaussian mixtures, i.e. we estimate parameters of the mixture using the *Expectation-Maximization* algorithm (Dempster et al., 1977; Minka, 1998). We compare 9 variants with different parametrizations of the covariance matrix for the gaussian distributions and the method from the XCMS package (Smith et al., 2006). I implemented some of the variants and was involved in conceptual work on the model.

It is difficult to judge the quality of the alignments because we do not know the ground truth. Besides visual comparison we employ the *False Discovery Rates* concept (Benjamini and Hochberg, 1995) to assess different alignments in the context of patient classification problem.

Chapter 4 concentrates on modeling proteolytic activity, i.e. the process of cutting peptides by enzymes. It is an important topic due to the role of this process in neoplastic diseases (Villanueva et al., 2006b).

Medical diagnostics based on mass spectra of blood samples is difficult, because of large variability of images caused by *ex-vivo* proteolysis. Paradoxically, one can try to take advantage of the differences in the proteolytic activity between samples. To this aim we built a two component bayesian model (Kluge et al., 2009) that formalizes amino acid chain cleaving. The first component describes peptide degradation resulting from *exopeptidase*[1] activity, while the second one handles acquisition of the concentrations of peptides from LC-MS data. It is assumed that the process is in the stationary state. We design and implement a Metropolis-Hastings algorithm for sampling from the model's posterior distribution. We test it on synthetic datasets and real colorectal cancer datasets.

The second part of Chapter 4 is based on (Gambin and Kluge, 2010) and describes a new version of the proteolysis model that handles cuts at arbitrary sites of the amino acid chains (i.e. it covers both exopeptidases and *endopeptidases*[2]) and describes the system in time (process stationarity is not assumed). Moreover peptidase cleavage patterns from the MEROPS database (Rawlings and Barrett, 2000) are integrated into the model. The estimation procedure and the error model for LC-MS readings is simplified — we use the L-BFGS-B algorithm (Lu et al., 1994) in order to solve a non-linear least squares problem.

[1] Exopeptidases are enzymes that cleave a single amino acid from an end of an amino acid chain.

[2] Endopeptidases cleave somewhere in the middle (i.e. not at the ends) of an amino acid chain.

One of the subroutines used by the estimation procedure is triangular matrix exponentiation that solves a linear differential equations system. We give a recursive formula for the entries of the resulting matrix, which can be translated into a dynamic programming algorithm. It could be especially useful in symbolic computations[3]. We did not see such characterization of the triangular matrix exponentiation in the literature.

I was involved in all stages of the proteolysis modeling project. I implemented our methods and computational experiments. To the best of our knowledge these are the first models built specifically with peptidase activity in mind for LC-MS data with potential applications to medical diagnostics.

[3] In our implementation a function from one of the R software package (Team, 2009) libraries is used.

Chapter 2

Mass spectra preprocessing

Raw data files produced by mass spectrometers are far from being interpretable biologically or chemically. These are simply lists of mass-to-charge ratios of particles interacting with a detector (hitting it or passing near it depending on the mass spectrometry technology used). If the machine is fitted with a liquid chromatography column, then also particles' passage time through the column (known as the *retention time*) is recorded, which corresponds to hydrophobicity. In order to save space, this information is usually binned along the mass-to-charge ratio and the retention time dimension and only counts of particles in each bin are produced.

If these data are to shed light on biological processes, they have to be presented from a biologically meaningful point of view. This may include determining what peptide sequences are present in a sample or grouping signals that correspond to the same peptide.

Section 2.1 introduces and describes LC-MS data. Section 2.2 deals with pick peaking which is the problem of converting a sampled continuous signal into a list of peaks. A brief description is presented of how we cope with this problem using the NMRPipe (Delaglio et al., 1995) package. For the isotopic envelopes detection problem custom software was developed by Krzysztof Kowalczyk with parts written by me. The results were published in (Gambin et al., 2007) and are described in Section 2.3.

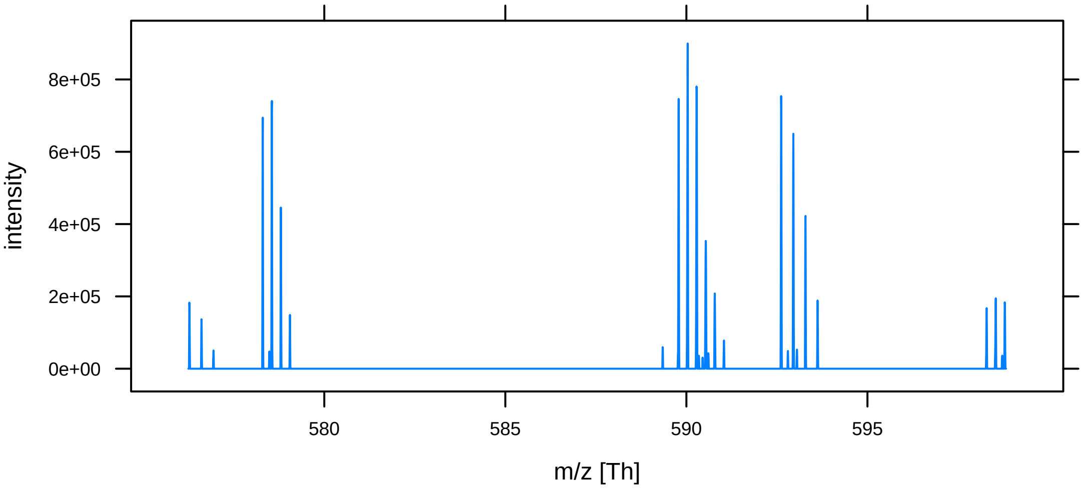


Figure 2.1: A 1-dimensional mass spectrum. On the horizontal axis is the mass-to-charge ratio (in thomsons), on the vertical – the intensity, i.e. the number of particles (in unspecified units). Peaks can be seen corresponding to particles with specific mass-to-charge ratio. Only a part of the spectrum is shown (about 1.5% of the domain) as the mass-to-charge ratio ranges in our data from 300 Th to 2000 Th.

2.1 LC-MS data

2.1.1 1D spectra

First, a 1-dimensional mass spectrum will be described. In a raw form this is simply a list of mass-to-charge ratios of particles that interacted with the detector. The detector however is not perfect and small deviations in mass-to-charge ratios occur. Furthermore this list is binned in order to save space so only counts of particles in each bin are available. The spectrum is usually depicted as a function R → R as in Fig. 2.1 which is the result of interpolating the counts in bins. The horizontal axis corresponds to the mass-to-charge ratio given in minus *thomsons* ($1\,\mathrm{Th} = 1\,\frac{\mathrm{Da}}{\mathrm{e}}$, where 1 Da (*dalton*) is the atomic mass unit and 1 e is the electron charge). Peaks of this function correspond to particles with specific mass-to-charge ratio. Their height (or volume) roughly describes the amount of

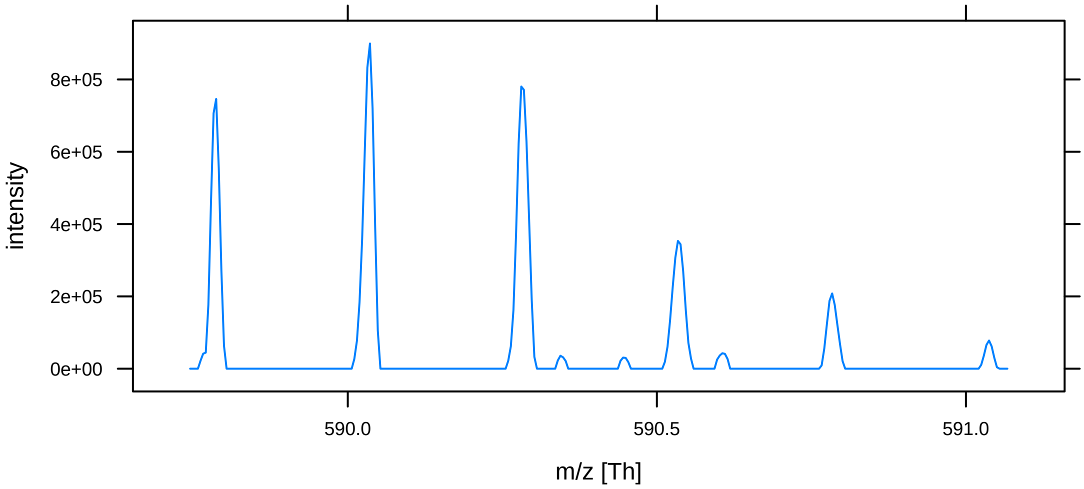


Figure 2.2: An isotopic envelope of a single peptide consisting of six peaks (additional three small peaks are noise). The leftmost peak is the monoisotopic peak, i.e. corresponding to particles with no ^{13}C atoms (^{12}C atoms only). Next peak (the largest one) corresponds to particles with exactly one ^{13}C atom. The peaks are spaced $\frac{1}{4}$Th apart indicating that the particles have charge $-4\,$e.

the particles, but we do not specify the units this amount is expressed in.

Many authors do not distinguish 1 Th from −1 Th. We also will not, since this work deals exclusively with positive (proton) charges and it will not lead to confusion.

Usually one sees groups of peaks spaced closely and regularly. These are the *isotopic envelopes* (see Fig. 2.2). Each peak in such group corresponds to a particle with the same atomic composition and the same charge but having different mass due to different isotopic composition.

Peptides are composed of carbon, hydrogen, nitrogen, oxygen, phosphorus and sulfur atoms. All of them come in multiple isotopic versions differing in the number of neutrons and mass in consequence. Many of those versions occur rarely in nature and can be ignored. A particle that consists only of the most abundant (principle) isotopes is called *monoisotopic*. Most common carbon isotopes are ^{12}C with 6 neutrons (98.9% of carbon atoms in nature) and ^{13}C with 7 neutrons

(1.1% of carbon atoms in nature).

Shape of each envelope can be inferred given the atomic composition of a particle. Assume a particle with n carbon atoms is considered. The peaks in its isotopic envelope are shaped like a binomial distribution with parameters n (the number of trials) and 0.011 (success probability), i.e. a particle can be treated as composed of independently chosen isotopic versions of each carbon atom. This scheme can be generalized to include more isotopes, however it is not needed in applications considered in this thesis.

Peaks in an isotopic envelope are spaced $\frac{1}{\mathrm{ch}}\,\mathrm{Th}$ apart, where ch is the number of additional protons attached (each having charge $-1\,\mathrm{e}$) while the particle was passing through the spectrometer. This number can vary depending on the mass spectrometry technology used. In data sets considered here one can observe particles with charges in range $-8\,\mathrm{e}$ to $-1\,\mathrm{e}$. Therefore a single particle type (the same peptide) can have a few isotopic envelopes, each corresponding do a different charge. The mass-to-charge ratios of peaks in those envelopes can be computed easily given the atomic composition of a particle, as long as one remembers to take into account masses of additional protons.

The mass-to-charge ratio range of the data depends on the spectrometer settings. The range for the data presented in this section's figures was set to 300 Th to 2000 Th. The data was obtained from human blood plasma samples and was provided by the Mass Spectrometry Laboratory of the Institute of Biochemistry and Biophysics of Polish Academy of Sciences.

2.1.2 2D spectra

Previous paragraphs describe a 1-dimensional mass spectrum. When dealing with a complex peptide mixture such as blood plasma a 1-dimensional spectrum can be hard to interpret, because images of different particles are crowded together. In order to decrease the overlap one can stratify the mixture according to particles' hydrophobicity and produce a separate spectrum for each stratum. To accomplish this task the spectrometer can be fitted with a liquid chromatography column. The passage time of a particle through the column is highly correlated with its hydrophobicity. As particles leave the column, mass spectrometer repeatedly

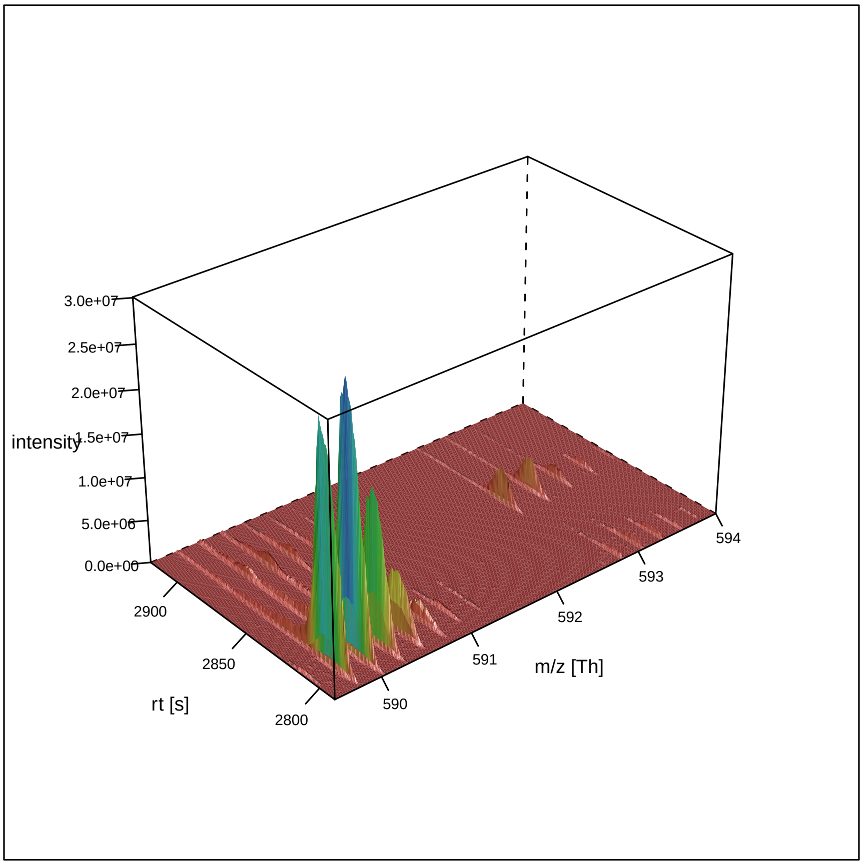


Figure 2.3: A 2-dimensional liquid chromatography mass spectrum. Only a part of the spectrum is shown (about 0.007% of the domain) as the retention time (particles' passage time through the liquid chromatography column) ranges from 1 s to 4920 s and the mass-to-charge ratio ranges in our data from 300 Th to 2000 Th.

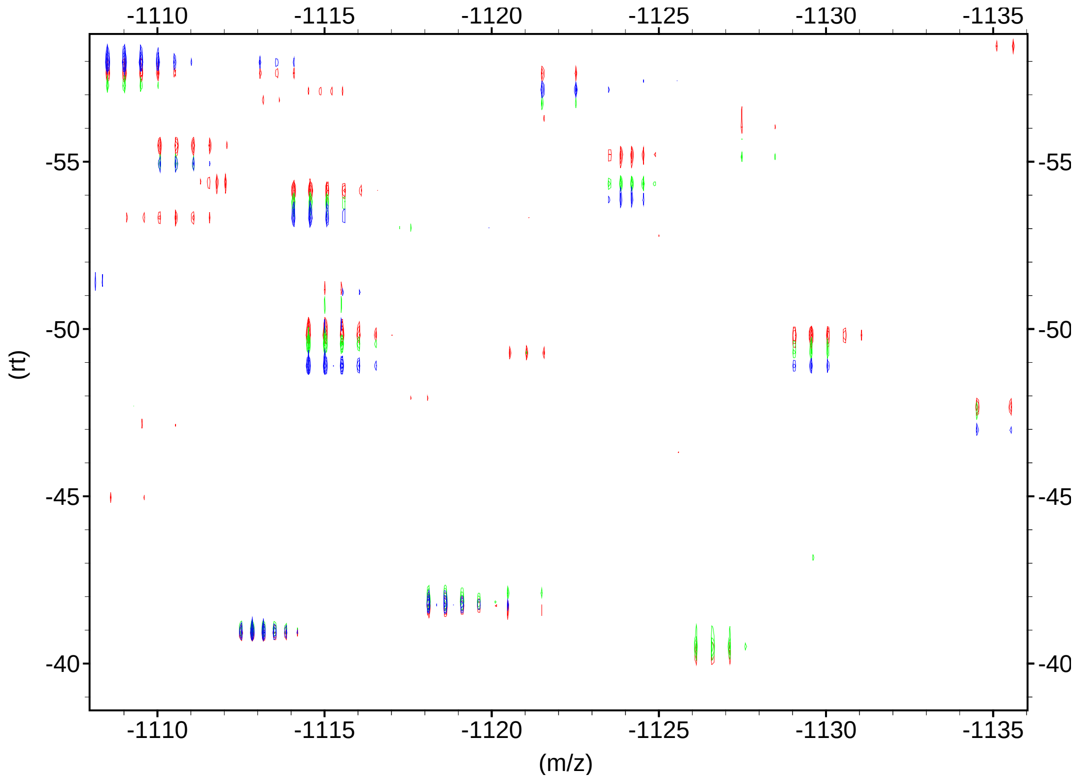


Figure 2.4: Distortions along the retention time (vertical) axis. Three independently produced spectra from the same sample (red, green and blue colors) are shown overlayed. Groups of peaks corresponding to the same peptides in different samples are clearly visible. Samples cannot be aligned perfectly by linear transformations along the retention time axis. In the upper left part and the lower part blue peaks are above red while in the middle part the order is reversed.

produces 1-dimensional spectra, which can be combined to form a 2-dimensional liquid chromatography mass spectrum (LC-MS, see Fig. 2.3).

The measurement accuracy of the new dimension (called the *retention time*) is not very reliable. Moreover one can witness significant non-linear distortions along this dimension when comparing two LC-MS runs of the same sample (e.g. the order of peaks' projections onto the retention time axis may change between two spectrometer runs), which is illustrated on Fig. 2.4.

2.2 Peak picking

NMRPipe (Delaglio et al., 1995) software package was designed for processing and analyzing NMR (Nuclear Magnetic Resonance) spectroscopic data (Keeler, 2005). After file format conversion however, some of its components can be applied to mass spectra. In the initial stage of data processing we used (Gambin et al., 2007) the NMRPipe package to improve the signal-to-noise ratio by eliminating high frequency signals with the 2D Fourier transform. In the next stage we took advantage of NMRPipe's 2D pick peaking procedure. The outcome was a list of coordinates of peaks in the spectrum with the signal strength (i.e. peak's height and volume).

Results obtained with NMRPipe were compared with recently proposed XCMS package (Smith et al., 2006) which also provides procedures for filtering and peak picking as well as matching peaks across samples and retention time correction. The XCMS package is included in the Bioconductor open source software project for the R programming language (Gentleman et al., 2004). We compared the efficiency and sensitivity of these two approaches, i.e. XCMS and NMRpipe (data not shown). Our observations are in favor of the NMR tool because of the more flexible visualization functions and the suitability for high-throughput processing. Further work is based on peak lists generated with the NMRPipe tool.

We assume that after the peak picking stage an LC-MS data set is a list of triples

$$(\mathrm{mz}_p, \mathrm{rt}_p, \mathrm{int}_p)_p$$

where:

- p ranges over all peaks in the data set,
- mz_p is peak's p mass-to-charge ratio,
- rt_p is peak's p retention time,
- int_p is peak's p intensity (i.e. height or volume).

2.3 Isotopic envelopes detection

The main goal of this processing step is to reduce the list of peaks present in a single LC-MS dataset into a list of the monoisotopic peptide masses. In the peak list each peptide is described by several peaks corresponding to different charge states of the peptide and different isotopic compositions. Hence, initially the list contains much redundancy. We eliminate the redundancy by determining monoisotopic mass and charge of each peptide signal. This process consists of the following steps:

1. noise filtering,
2. clustering into isotopic envelopes,
3. automated charge determination,
4. monoisotopic mass calculation,

which were implemented in our *mz2m* program (Gambin et al., 2007) available at `http://mz2m.sourceforge.net`.

There is a large body of research dealing with automated charge determination. Zhang and Marshall (1998) proposed Z-Score algorithm, which uses a scoring scheme to assign charges to ions. The other method described by Senko et al. (1995b) is based on the Patterson and Fourier transforms for selected areas of the spectrum. For deconvolution (i.e. finding out which peaks correspond to the same molecules) Senko et al. (1995a) proposed the *averagine* method, based on fitting an isotopic distribution from spectrum to theoretical distribution. This method is appropriate for large molecules (10-20 kDa). Horn et al. (2000) proposed the algorithm THRASH based on both Patterson and Fourier transforms and the averagine method. For small peptides the problem of grouping isotopic peaks is relatively easy because the monoisotopic peak is usually the highest or the second highest. In our method we incorporate the ideas from (Senko et al., 1995b) and (Horn et al., 2000) and adapt them to a two dimensional setting.

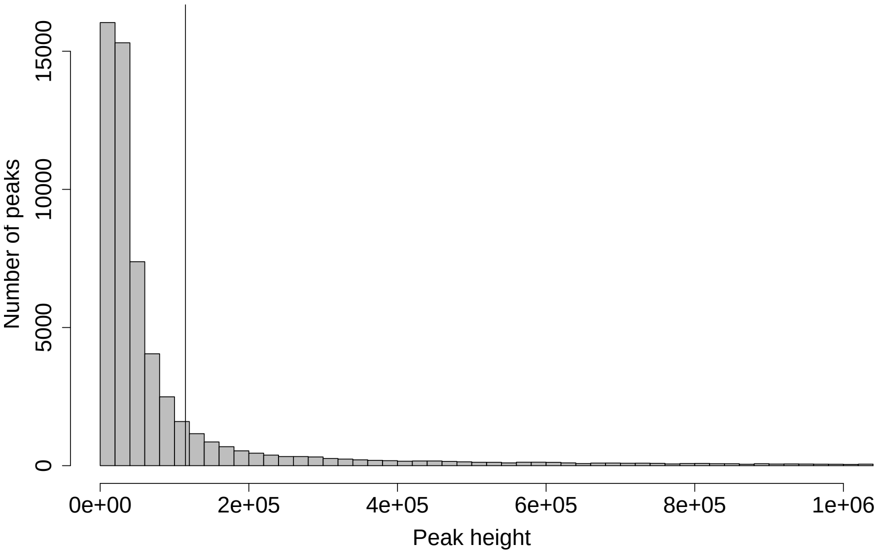


Figure 2.5: Distribution of signal abundance in a single blood serum peptidome: the cutoff threshold is calculated to filter out overrepresented small peaks.

2.3.1 Noise Filtering

Noise filtering performed during preprocessing phase has to be refined at the beginning of our algorithm to eliminate peaks corresponding to spurious signals. To this aim we simply discard peaks with height below the threshold T. The value of T is established based on analysis of the distribution of peaks' height in a given sample (Fig. 2.5).

The threshold is fixed to filter out small peaks with comparable height. To this aim we approximate the density function f of peaks' height distribution and set T to be the solution of the equation $f'(T) = -1$. Figure 2.5 is a typical example of f's shape so in practice one may assume that such T is well defined.

2.3.2 Isotopic Cluster Identification

At this stage our algorithm operates on the list of peaks computed during the preprocessing phase. Peaks are represented by several parameters: mass-to-charge ratio, retention time, intensity (i.e. height or volume). During peak clustering we take into account general properties of peptide isotopic clusters. We scan our 2D input dataset from left to right (direction of increasing mass-to-charge ratios) and examine peaks in the vertical stripe of 1 Th width (a variant of *sweeping method*, see Fig. 2.6). The position of the right border of the stripe is determined by the mass-to-charge ratio of the peak being examined (we call it the active peak).

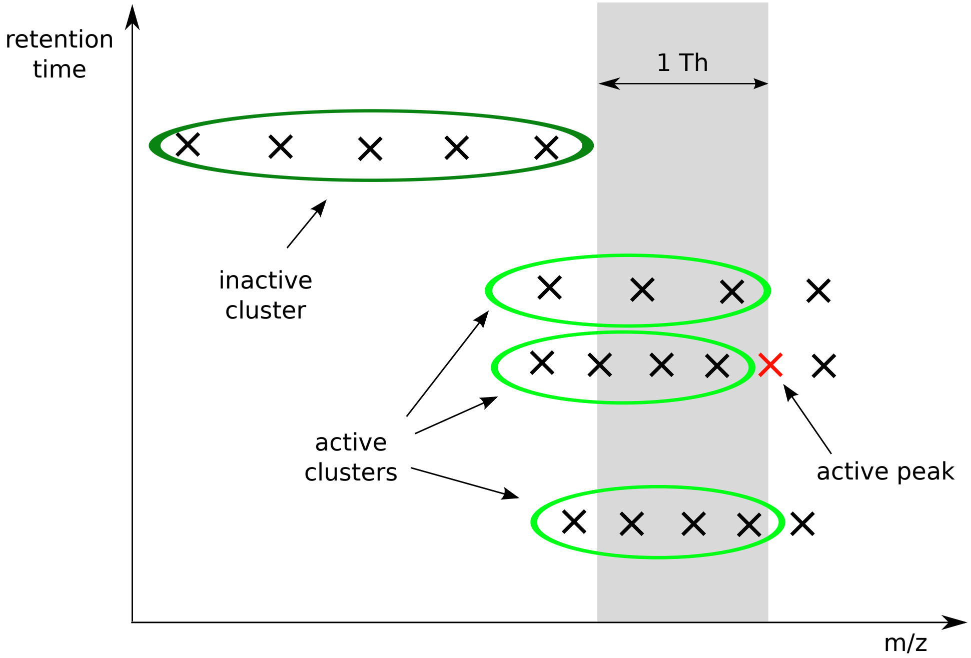


Figure 2.6: Sweeping method. Clusters of peaks marked with light green (active clusters) are candidates for extending with the active peak (marked with red). The cluster marked with dark green (inactive cluster) is outside the 1 Th stripe and cannot be extended.

We assume that all peaks to the left of the active peak (i.e. having lower mass-to-charge ratio) have already been clustered. All peaks to the right of the active peak will be considered in the next steps.

We call a cluster active when its last peak (i.e. the one with the highest mass-to-charge ratio) is in the currently considered stripe. We maintain the data structure containing the list of active isotopic clusters.

Our goal is to assign an active peak to one of the active clusters. The first step is to select from the set of the active clusters those which could be extended by our active peak.

The criteria for the peak fitting to an isotopic cluster are the following:

- distance between the peak and the cluster (i.e. between the peak and the rightmost peak in the cluster) in the domain of retention time should be smaller than the predefined threshold,
- the shape of the isotopic cluster extended by the active peak should pass user predefined filter (see below; by shape we mean the relative height, positions and the number of peaks in the cluster).

As the filter for the isotopic cluster we investigate here the proportions of the height of two neighboring peaks. We compute the possible extreme values for these proportions by considering polyserine and polyphenyloalanine peptides and we filter out clusters having these proportions outside computed values. In fact these extreme values are further relaxed to encompass sulphur containing peptides. Our algorithm is designed to deal with small peptides. Hence only two possibilities for monoisotopic peak position are considered by the algorithm: either the first or the second peak in the isotopic cluster can be the highest one.

The behavior of the algorithm depends on the number of candidate active isotopic clusters:

- if there is no candidate cluster for the active peak, we form a new cluster containing this peak,
- if here is exactly one candidate cluster we extend it with the active peak,
- if more than one candidate cluster exists we assign the active peak to the cluster whose monoisotopic peak is the highest (such a situation is quite rare but it happens when the signal coming from one peptide is artificially split in the domain of the retention time (c.f. Fig. 2.7)).

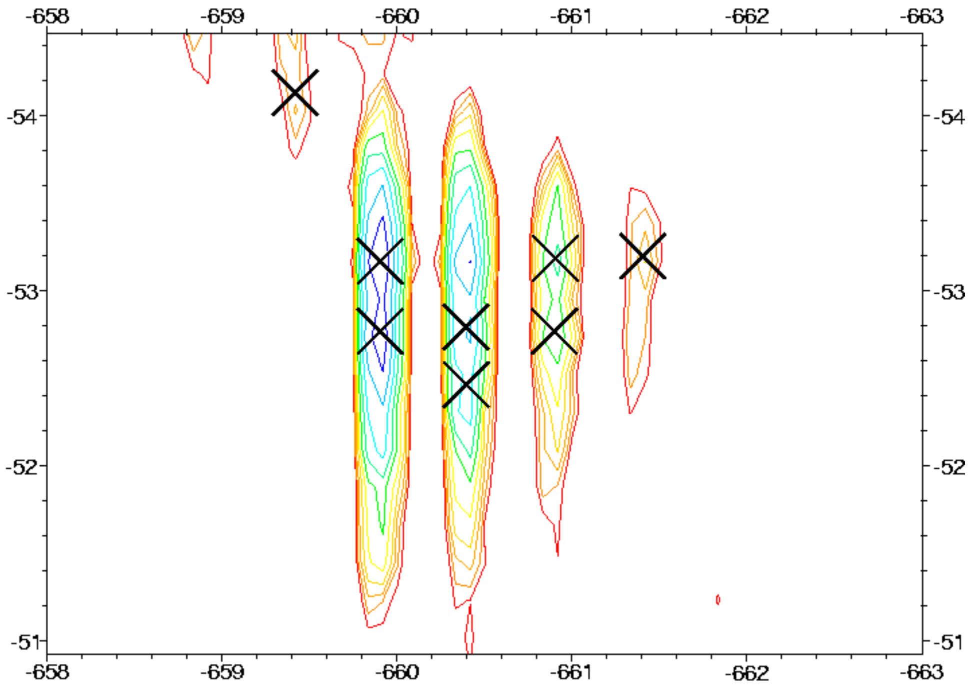


Figure 2.7: Artificial peak separation in the retention time domain. Isotopic envelopes visualized by the Sparky tool (Goddard and Kneller, 2006). Peaks found by NMRPipe are depicted as black Xes. Horizontal axis — mass-to-charge ratio, vertical axis — retention time, height is color coded increasing from red to blue.

2.3.3 Automated Charge Determination

We have implemented two versions of this step, simple and fast and a more sophisticated one. The simple version uses only information from the peak spacing in the isotopic clusters as prepared in the previous step and it can be viewed as a variation of the Z-Score method from (Zhang and Marshall, 1998). We assume that the charge is simply the reciprocal of the distance between two adjacent peaks in the isotopic cluster. We count results for each possible space interval and choose the most frequent value as the charge.

This method is very fast but also susceptible to errors especially when there are artifacts and split peaks in the spectrum. This method also cannot determine charges for overlapping isotopic envelopes.

The second method is a variation of the method from (Senko et al., 1995b). It operates on a list of clusters and a raw mass spectrum. The original method is designed for 1D spectra. Here we use the isotopic clusters found in the previous step to approximate the coordinates of isotopic clusters in the spectrum. We perform the projection of the isotopic cluster in the direction of the retention time and use the combination of Patterson and Fourier transform for the mass-to-charge ratios of the isotopic cluster to determine the charge.

First we compute the Patterson transform of the part of the projected spectrum f containing the cluster:

$$P_f(\Delta M) = \sum_i f(M_i - \frac{\Delta M}{2}) * f(M_i + \frac{\Delta M}{2})$$

where ΔM is the inverse of charge, M_i are mass-to-charge ratios and the sum ranges over the location of the isotopic cluster. If a clear maximum can be found, we output $\frac{1}{\Delta M}$ as the charge, otherwise we look for a maximum in the Fourier transformed data.

2.3.4 Isotopic model (mass decomposition)

Recall that our procedure is designed to deal with small peptides (up to 5000 Da). For such data the *averagine* model by Senko et al. (1995a) might not be suitable. In order to estimate the significance of the cluster we fit its group of peaks to the estimated theoretical isotopic distribution calculated for the given monoisotopic mass. This step is coupled with the mass and charge determination.

We start with the assumption that the first visible peak in the isotopic cluster corresponds to the monoisotopic mass. If this assumption turns out to be false, the first visible peak in the spectrum is assumed to be the second in the theoretical envelope (i.e. corresponding to the isotope containing one neutron more than the monoisotopic version).

From the putative mass-to-charge ratio and the charge determined in the previous step we calculate the monoisotopic mass. Then we perform mass decomposition, i.e. we guess the atomic composition for the given mass.

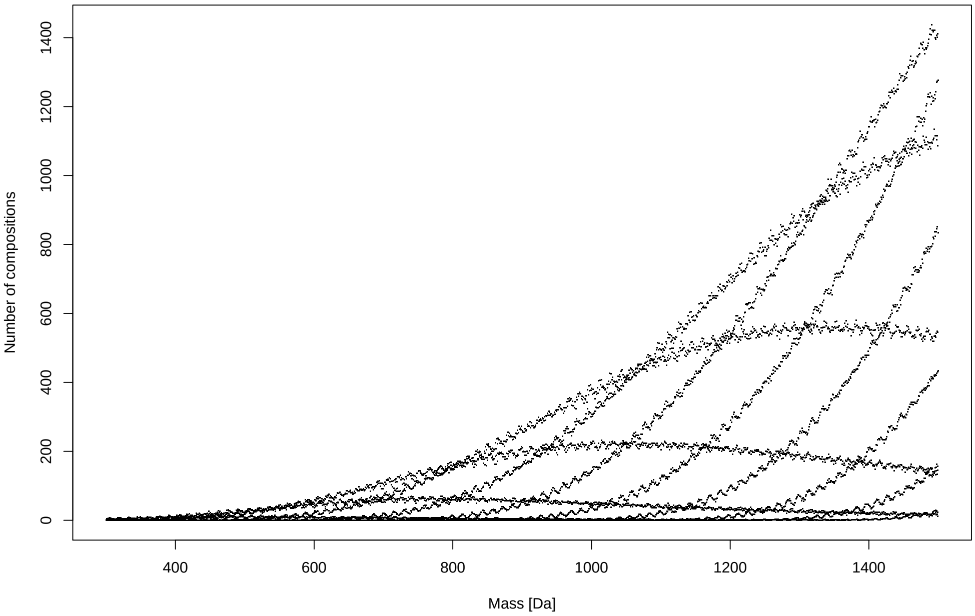


Figure 2.8: The distribution of peptides' abundance for a given monoisotopic mass and precision $\epsilon = 0.01$. This graph does not seem to be a function, but it is in fact — for each mass there exists exactly one point corresponding to the number of different atomic compositions. This function behaves very non-continuously — different visible curves arise from combinatorial properties of mass decomposition problem, namely, some masses have much larger amount of possible atomic compositions than others.

Our mass decomposition procedure works as follows. Let m be a monoisotopic mass of a peptide. First, we find candidates for atomic compositions of this peptide: each candidate can be represented as a vector of length 5, storing the numbers of atoms of C, H, N, O and S. Mass of each candidate can differ from m by at most ϵ and has to represent a chain of amino acids.

In order to be able to efficiently find compositions of masses up to M, we perform the following preprocessing. Let m_h be the mass of the heaviest amino acid considered. We generate all compositions of peptides with mass not exceeding $\frac{M}{2} + m_h$, sort them by mass and store in a vector v. To answer a query m, for

mass	ϵ	# candidate compositions
1428.65	0.0001	1
	0.001	14
	0.01	123
	0.1	1157

Table 2.1: The number of candidate atomic compositions for lysozyme peptide mass measured with different precision (ϵ).

each element of v we check if there exists an element in v, such that the sum of masses of those two elements differs from m by at most ϵ (precision parameter).

For small peptides our procedure gives a reasonable number of candidate atomic compositions (c.f. Table 2.1 and Fig. 2.8).

For each candidate atomic composition of a given monoisotopic mass we calculate theoretical isotopic distribution using dynamic programming technique and fit the experimental data (i.e. isotopic cluster determined in the previous step).

To estimate the quality of fit the figure-of-merit (FOM) value is calculated as follows (Horn et al., 2000):

$$\text{FOM} = \frac{k}{\sum_n [(A_n - \omega I_n) + (\omega V)^2]} \tag{2.1}$$

where:

- A_n is the abundance of then nth peak in the theoretical isotopic distribution,
- I_n is the observed signal intensity at the point corresponding to the nth isotopic peak,
- V is the maximum value in the valley between adjacent peaks (in the interval from $\frac{1}{3}$ to $\frac{2}{3}$ of the distance between consecutive peaks),
- ω is the normalization factor,
- k is the number of values compared (i.e. all peaks and valleys in the isotopic cluster).

type of sample	# of samples	min	max	mean	stddev
CF (peaks)	59	25613	108831	63032	16147
CF (masses)	59	1341	5244	3361	821
CC (peaks)	40	57719	213178	124225	53629
CC (masses)	40	2657	8227	5250	2250

Table 2.2: Mass and peak statistics. CF – cystic fibrosis dataset, CC – colorectal cancer dataset.

The normalization factor scales the intensities such that the average intensity from three most abundant peaks in the theoretical distribution equals the average intensity for three corresponding peaks from experimental spectrum. The exception is made for very small masses when the first peak, being the most abundant one, is used to scale the distribution.

The best fit is selected for further analysis. Its FOM has to exceed some user-predefined threshold.

2.3.5 Mass and Volume Calculation

To determine the monoisotopic mass one needs to know the charge of the isotopic cluster and the coordinates of the monoisotopic peak. Both these values are determined in the previous step by the best fit to the theoretical isotopic model (i.e. the fit with the greatest FOM value). The volume, corresponding to the abundance of the peptide, is calculated as the sum of volumes for all peaks in the isotopic cluster.

2.3.6 Automated interpretation results

The goal of our *mz2m* program is to calculate the list of peptides in the sample identified by their mass and retention time. The program has been tested on several datasets. Before starting analysis of complex peptide mixtures many relatively simple samples have been processed for calibration of the whole procedure. These samples include tryptic digest of bovine serum albumin (BSA), lysozyme and cytochrome C.

To demonstrate the application of the LC-MS for large-scale analysis the following sets of complex peptide mixtures were processed (c.f. Table 2.2):

- blood plasma samples from cystic fibrosis (CF) children and their healthy family members (59 samples),

- blood serum samples from colorectal cancer (CC) patients and healthy donors (40 samples).

To estimate the quality of our algorithm several tests were performed. However, the main goal was to verify the following two aspects which are crucial for medical applications.

- how many peptide signals were missed by the automated processing,

- how many peptide signals were interpreted incorrectly.

The best way of validation here was visual inspection of the results (Fig. 2.9). For the fragments of some samples all peptide signals were manually counted and interpreted with the assistance of the Sparky visualization tool (Goddard and Kneller, 2006). The result has been compared to the program output. Table 2.3 presents the number of interpreted peptide signals and the number of errors. We consider four types of errors: false positives, missing peptides, incorrect charge and incorrect mass calculation. The program misses about 6% of all peptides and returns about 6% peptides with incorrect mass. The incorrect interpretation of the charge is closely related to very strong signal deformation (c.f. Fig. 2.7). We want to emphasize that the peptide list generated by our algorithm contains only about 2% of false positives (possibly experimental and software artifacts).

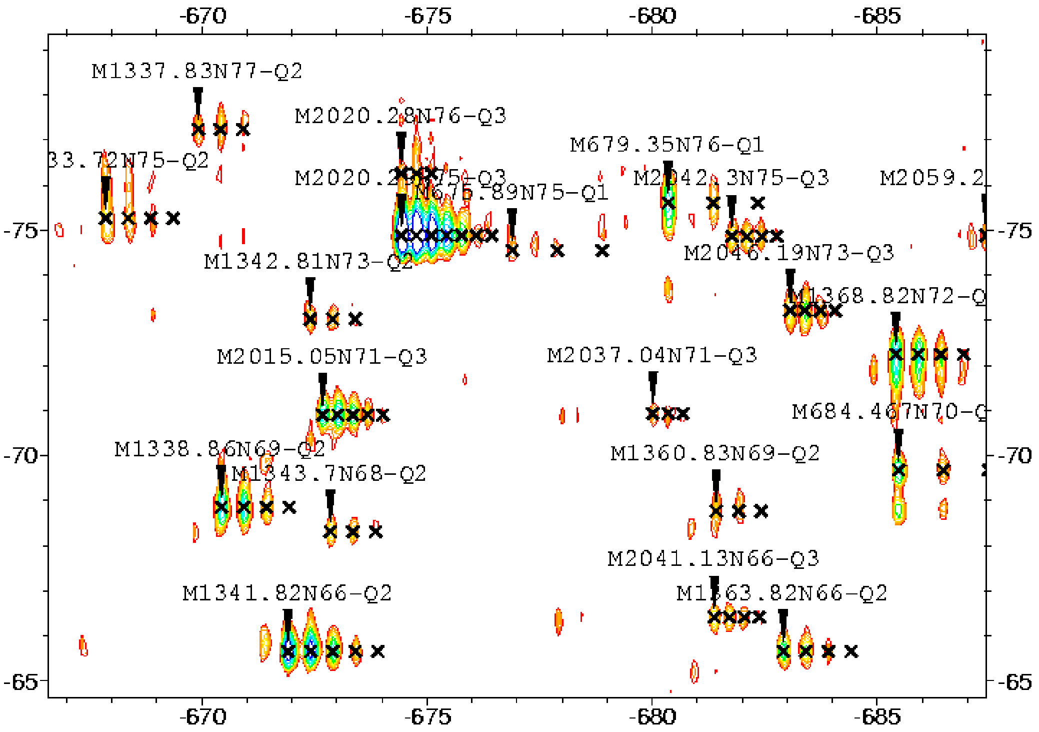


Figure 2.9: Masses and charges calculated by our *mz2m* program for the fragment of the spectrum. Peaks are marked as black crosses, small arrow denotes the monoisotopic peak in each isotopic cluster, the monoisotopic mass (M) and charge (Q) are given for each identified peptide.

	mean	variance
peptides manually counted	377	14
peptides correctly described by the program	321	10
false positives	8	7
incorrect charges	5	3
incorrect masses	25	1.4
peptides not found	26	1.4

Table 2.3: Error statistics for 3 manually tested datasets.

Our procedure can efficiently process LC-MS spectra with sensitivity sufficient for medical applications such as searching for biomarkers for diagnostics (screening and prognostic tests).

An interesting open question is how to deal with more dimensions. A challenging problem is to design a framework for multidimensional mass spectrometry based on different separation techniques (Janini et al., 2005; Shvartsburg et al., 2005).

The method can also be useful for LC-MS based differential proteomics, in which quantitative comparison of protein levels in two samples is made possible by labeling peptides in one of the samples by a stable isotope (Figeys et al., 2001; Twigger et al., 2005; Valkenborg and Burzykowski, 2011), e.g. it allows to differentiate between ^{16}O and ^{18}O labeled species in an automated way and to quantitate peptide ratios.

Chapter 3

Mass spectra alignment

One of the most common tasks one needs to perform on multiple mass spectra is to point peaks (or monoisotopic peaks or isotopic envelopes depending on the form of input) corresponding to one another in different spectra. This step is crucial in medical diagnostics applications where blood plasma or serum samples from many individuals are analyzed separately by a mass spectrometer. Even when multiple analyses of the same sample are conducted the results differ as each spectrometer run produces some drifts and distortions both along the mass-to-charge ratio axis and the retention time axis (the latter causing much more trouble). Lange et al. (2008) present an overview and comparison of LC-MS alignment procedures.

Two approaches to this problem are presented in Sections 3.1 and 3.2. The first one performs retention time stretching and shifting while the second one employs peak clustering via a two stage procedure consisting of a DBSCAN (Ester et al., 1996) pass and gaussian mixture model based clustering. Results from Section 3.2 were published in (Gambin et al., 2006) and (Łuksza et al., 2009). The method from Section 3.1 was used in (Kluge et al., 2009) but was not described in detail.

3.1 Alignment via retention time transformation

Errors on the mass-to-charge ratio axis are orders of magnitude lower than on the retention time axis, therefore only the retention time axis will be considered in this section. The idea presented here is to apply a linear transformation along this axis separately for each sample (mass spectrum), so that each peak has near neighbors in other samples. Linear transformations are certainly not enough to obtain a perfect alignment. Looking closely into two LC-MS data sets one can find matching pairs of peaks that are ordered along the retention time axis in a way that one sample cannot be obtained from the other by applying a linear transformation. In many situations however a linear transformation gives a great improvement in the alignment.

A specific situation that we have in mind here is the HPLC (High Performance Liquid Chromatography) column replacement. Even at first sight samples before and after the replacement look very different. Applying our procedure roughly corrects the discrepancies. We will use a data set comprising LC-MS samples from 3 batches separated by the HPLC column replacement and one additional sample with peptide list from a tandem LC-MS experiment[1]. The list can be ignored when thinking about the problem — it is just a convenient way to assign peptides (amino acid sequences) to peaks when one can form a set of peptides expected in the samples. It will find use in Section 4.1.

3.1.1 Problem statement

Let P be the set of all peaks from the set of all samples S. We write mz_p, rto_p, rt_p, ch_p, s_p for $p \in \mathrm{P}$ to denote mass-to-charge ratio, original retention time, corrected retention time, charge and peak's p sample respectively. We assume that rt is a linear function of rto, i.e.:

$$\mathrm{rt}_p = a_{\mathrm{s}_p} \,\mathrm{rto}_p + b_{\mathrm{s}_p}.$$

[1] Tandem MS (also called MS/MS) is a technique where particles undergo at least two separation steps with fragmentation in-between. It can be used to sequence proteins (i.e. obtain a sequence of amino acids composing a protein).

We define a criterion to be optimized over the a and b parameters. In a perfect alignment peaks corresponding to the same isotopic versions of the same peptides with the same charges in different samples should coincide. For each peak, we therefore try to count peaks from other samples in its neighborhood and favor the alignments with greater total counts.

For $p \in \mathrm{P}$, $s \in \mathrm{S}$ and a parameter $w > 0$ let $\mathrm{N}_w(p, s)$ be the set of peak's p w-neighbors along the mass-to-charge ratio axis with matching charge in sample s, i.e.:

$$\mathrm{N}_w(p, s) = \{q \in \mathrm{P} \mid \mathrm{s}_q = s, \mathrm{ch}_q = \mathrm{ch}_p, |\,\mathrm{mz}_q - \mathrm{mz}_p\,| \leq w\}$$

and $\mathrm{D}_w(p, s)$ be the distance on the retention time axis to the nearest neighbor in sample s:

$$\mathrm{D}_w(p, s) = \begin{cases} \min\{|\,\mathrm{rt}_q - \mathrm{rt}_p\,| \mid q \in \mathrm{N}_w(p, s)\}, & \text{if } \mathrm{N}_w(p, s) \neq \emptyset, \\ 0, & \text{if } \mathrm{N}_w(p, s) = \emptyset. \end{cases}$$

After experimenting with different values of the w parameter, we set it to 0.075.

Low $\mathrm{D}_w(p, s)$ value indicates that peak p has a corresponding peak in sample s. Informally speaking we set our objective to minimizing the expression:

$$F(a, b) = -\sum_{p \in \mathrm{P}, s \in \mathrm{S}} \arctan(\mathrm{D}_w(p, s))$$

over a and b. The arctan function is used to dampen the effect of distant peaks on the score.

In order to make the above problem well defined we:

- fix $\sum_{s \in \mathrm{S}} a_s = |S|$ as otherwise a would approach 0,
- add the $G(b) = -\frac{1}{2d} \sum_{s \in S} b_s^2$ term to the objective expression as otherwise adding an arbitrary constant to b would keep the objective invariant.

3.1.2 Solution through the Metropolis–Hastings algorithm

In practice we will not try to find an exact minimum but be content with sampling (a, b) from the distribution with density proportional to $\exp\left(F(a, b) + G(b)\right)$.

Note that technically, this procedure can be viewed as sampling from a posterior distribution with density $f(a, b \mid \mathrm{rto})$ implied by defining:

- the likelihood density as

$$f(\text{rto} \mid a, b) \propto \exp\left(F(a,b)\right),$$

- the prior distribution on $\frac{a}{|S|}$ as a uniform distribution on a simplex,
- the prior distribution on b_s as a normal distribution with mean 0 and variance d for each $s \in S$ independently.

We apply the Metropolis–Hastings algorithm (Hastings, 1970) to perform sampling. The move proposals are generated by executing with equal probability one of the two following steps:

1. changing a:
   - generate $s_1, s_2 \in S, s_1 \neq s_2$ uniformly,
   - generate $(a'_{s_1}, a'_{s_2}) \sim \text{Dirichlet}\left(c\frac{a_{s_1}}{|S|}, c\frac{a_{s_2}}{|S|}\right)|S|$, where c is a parameter of the algorithm and was set to 1000 after some tuning,
   - set a'_s to a_s for $s \notin \{s_1, s_2\}$,
   - propose transition $(a, b) \mapsto (a', b)$,
2. changing b:
   - generate $s_1 \in S$ uniformly,
   - generate $b'_{s_1} \sim \text{Normal}\,(b_{s_1}, d)$,
   - set b'_s to b_s for $s \neq s_1$,
   - propose transition $(a, b) \mapsto (a, b')$.

A standard acceptance rule is used, i.e. transition proposal $(a, b) \mapsto (a', b')$ is accepted with probability

$$\min\left\{\frac{f(a', b' \mid \text{rto})}{f(a, b \mid \text{rto})}\,\frac{p((a,b) \mapsto (a', b'))}{p((a', b') \mapsto (a, b))}, 1\right\},$$

where p denotes the proposal density.

3.1.3 Alignment results

The procedure was used to align 126 LC-MS blood plasma spectra. Additionally a list of peptide sequences annotated with mass, charge and retention time was included as the 127th sample. The list was based on a separate MS/MS experiment. This addition will be helpful at a later stage (Section 4.1), where peaks corresponding to specific peptide sequences will be needed.

The spectra came in 3 batches having 29, 30 and 67 samples respectively. The batches were collected at different time periods separated by the HPLC column replacement.

The procedure was run for $2 * 10^6$ steps. Figure 3.1 documents this run by showing the score (the F function), a and b in successive steps. Figure 3.2 shows the final values of a and b. Estimated parameters stabilized after 10^6 steps. One can easily identify batches of samples evolving together and producing similar final parameter values. This behavior shows that an HPLC column replacement produces significant changes in the LC-MS spectrum which is consistent with our experience.

It is hardly possible to compute meaningful statistics describing alignment quality when the true alignment is not known, therefore we turn to visual validation of the results. Figure 3.4 shows 29 LC-MS samples from one batch before and after the linear retention time correction with our procedure. Two distant parts of the spectra are depicted before and after alignment, each showing a significant improvement. Figure 3.3 depicts alignment results across batches (2 spectra from each of the 3 batches). Clearly alignment within batches is better than the alignment between batches.

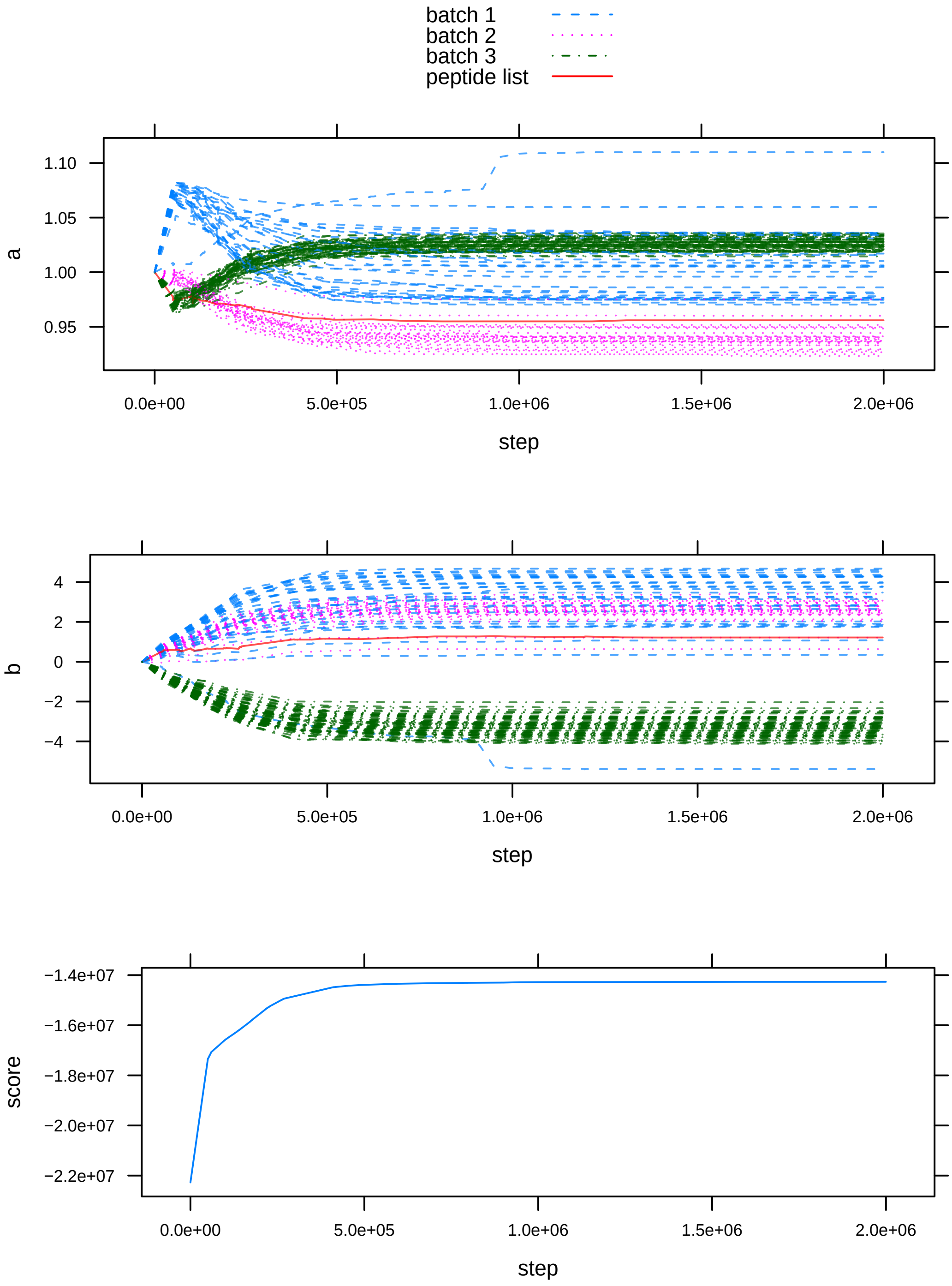


Figure 3.1: Evolution of the a (top) and b (middle) parameters during $2*10^6$ steps of the alignment algorithm run. The a and b parameters correspond respectively to scaling and shifting samples along the retention time axis. Evolution of the score (the F function) is shown at the bottom.

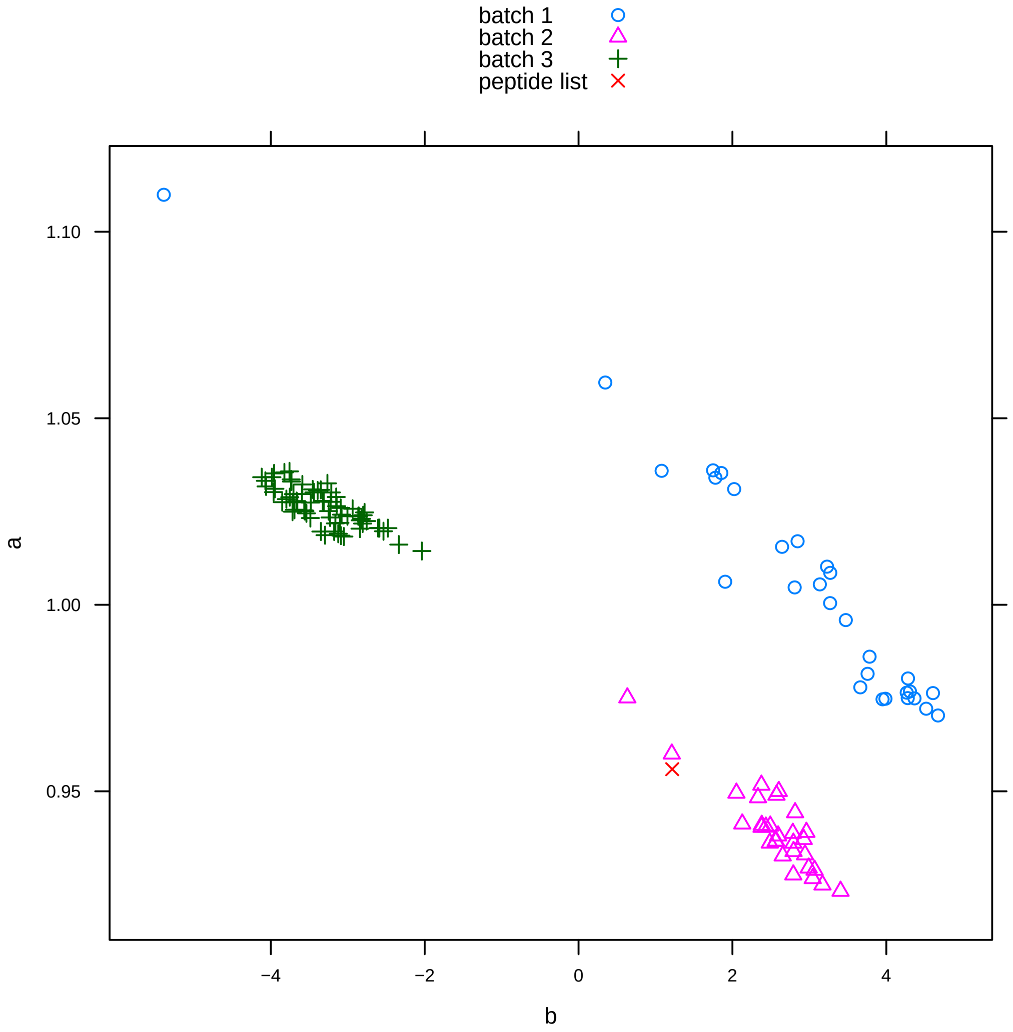


Figure 3.2: Final values after $2 * 10^6$ steps of the alignment algorithm run of the a and b parameters corresponding respectively to scaling and shifting samples along the retention time axis.

As mentioned before mass spectra alignment needs to be performed whenever one wishes to compare information from different mass spectrometer runs. This particular method was used as a preprocessing stage for applications described in Section 4.1 and the results of the computations presented here are in fact exactly the ones used in Section 4.1.

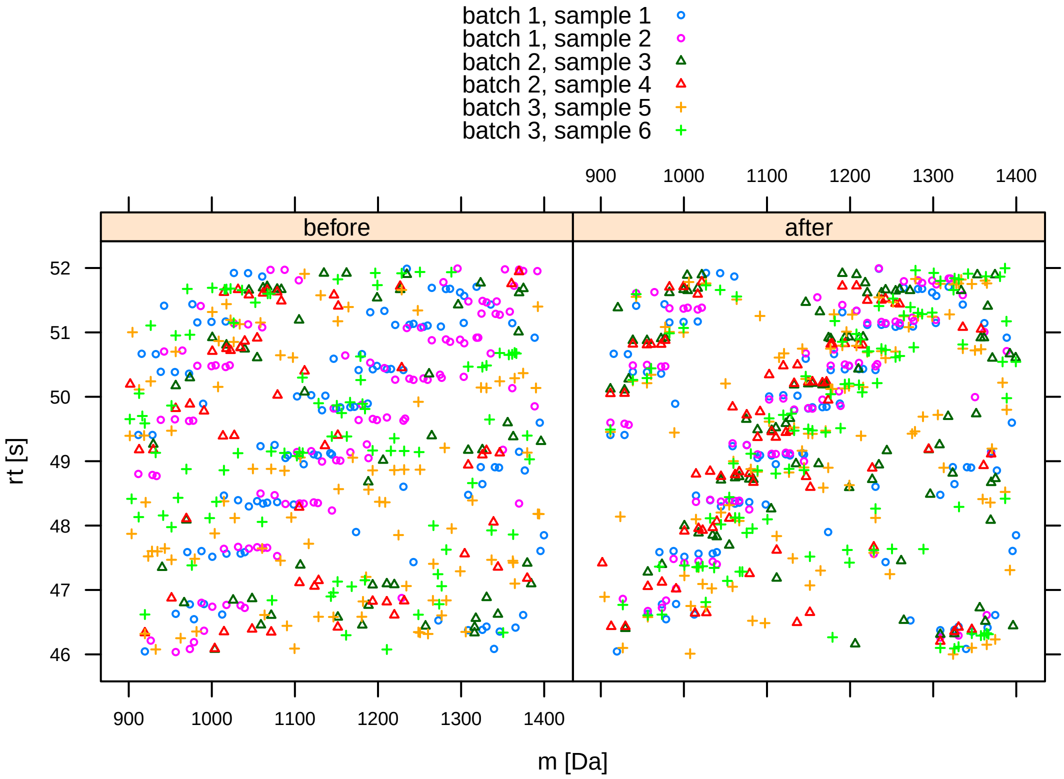


Figure 3.3: Peaks with charge −2 e from a fragment of 6 LC-MS spectra (2 spectra from each of the 3 batches). Different spectra are marked with different colors. Additionally shape corresponds to batches.

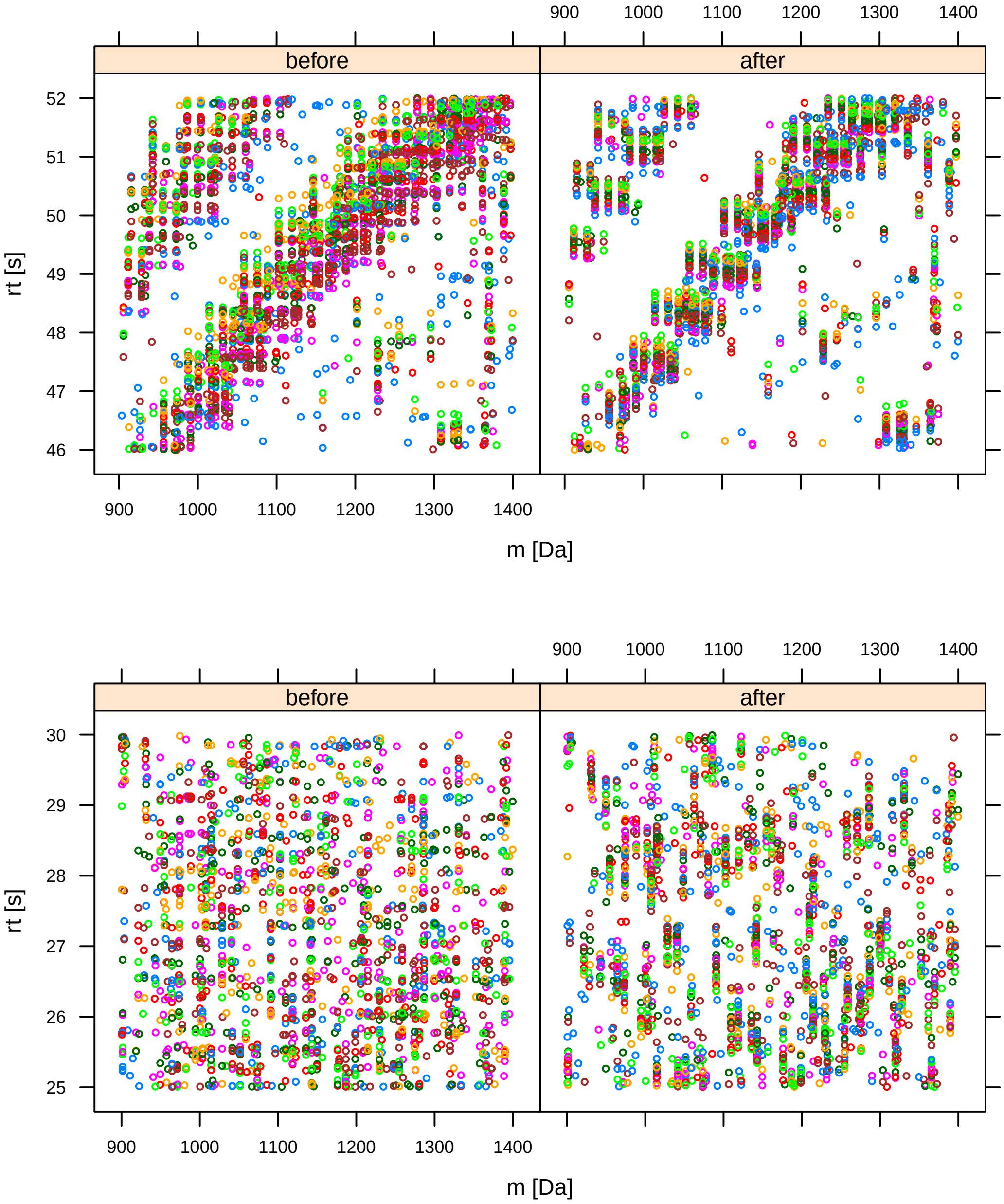


Figure 3.4: Peaks with charge −2 e from two fragments (top, bottom) of 29 LC-MS spectra from a single batch before and after the linear retention time correction. Different spectra are marked with colors (colors are recycled as only 7 colors are used). At the top, groups of peaks corresponding to isotopic envelopes of the same peptides in different samples can be distinguished after the alignment — a clear improvement from the state before the alignment. The improvement at the bottom picture is not as striking. This may be attributed to the fact that are almost no isotopic envelopes in this region. Notice however that vertical strings of peaks are much more vivid after the alignment, indicating that that the transformation moved corresponding peaks closer together.

3.2 Alignment via clustering

One of the possible approaches to the peak alignment problem is to apply a well known clustering algorithm. All monoistopic peaks from all samples can be treated as the input to such procedure, producing a grouping into subsets as the output. Each subset corresponds to one particle type (peptide) with a specific charge.

Here model based clustering via gaussian mixtures (Fraley and Raftery, 1998) will be explored. Due to efficiency reasons a two stage procedure will be proposed. The first stage is fast and produces a coarse clustering. The second stage is the model based clustering itself performed on each cluster from the first stage independently, producing more refined clusters.

The data actually being clustered are the mass-to-charge ratio and retention time pairs denoted by $(x_{p\,\mathrm{mz}}, x_{p\,\mathrm{rt}})_{p \in P}$. The intensities are not taken into account since usually there is no reason for a peptide to have the same intensities in two different samples.

This approach is already computationally demanding so we will not extend it further, despite the fact that some deficiencies can be pointed out. Firstly, one could complicate the model to include the fact that each peptide can have versions with different charges. Although no easy way is known to determine the relative intensities of differently charged versions a priori, those ratios should probably be the same in all samples. Leveraging this fact an enhanced procedure could make use of the intensity information. Another direction would be to constrain the number of peaks from each sample in a cluster to at most 1, however one has to remember that peak picking phase is not perfect, e.g. it can erroneously split one peak into two peaks. Finally the pick peaking and spectra alignment procedures could be integrated, but that would require enormous computational power.

In fact, due to efficiency reasons the model based clustering cannot be applied to the entire set of peaks directly. We propose the following two-stage procedure:

1. preliminary partitioning of the data set into non-overlapping subsets of moderate sizes,

2. application of the model based clustering to each subset separately.

In the first step of the clustering DBSCAN algorithm (Ester et al., 1996) is used, in the second step different models are fitted, description of which is provided in Section 3.2.1.

In order to improve the quality of the alignment and minimize the impact of the retention time deviations, the whole procedure is run repeatedly interleaved with the retention time correction procedure from the XCMS package (Smith et al., 2006) described in Section 3.2.3.

3.2.1 Model based clustering

The idea underlying model based clustering (Fraley and Raftery, 1998) is that each observation is an independent sample generated from a population of distributions. The observer does not know the origins of observations and tries to infer them, thus producing a clustering.

For each isotopic version of each peptide with a specific charge (denoted by r) we assume that peak locations are gaussian distributed with mean $(\mu_{r\,\mathrm{mz}}, \mu_{r\,\mathrm{rt}})$ and a diagonal covariance matrix Σ_r. The diagonality is motivated by the fact that errors in measurements along the mass-to-charge ratio axis and the retention time axis are independent since the first is due to the spectrometer itself and the second is due to the HPLC column.

A covariance matrix Σ can be written as λB, where $\lambda > 0$ is a scalar and B is a diagonal matrix having $\det B = 1$. This factorization is a convenient form to describe various parametrizations. The λ parameter corresponds to the cluster's volume and B to its shape. We investigate the following models:

$\Sigma_r = \lambda_r B_r$ – the most general diagonal model, where the volumes and shapes are allowed to vary between clusters,

$\Sigma_r = \lambda B$ – a model where clusters have identical shape and volume,

$\Sigma_r = \lambda_r B$ – a model where clusters have identical shape but their volumes may vary,

$\Sigma_r = \lambda B_r$ – a model where all the clusters have the same volume but their shapes may vary,

$\Sigma_r = \begin{pmatrix} c^2_{\text{mz}} & 0 \\ 0 & c^2_{\text{rt}} \end{pmatrix}$ – a model with fixed variances $c^2_{\text{mz}}, c^2_{\text{rt}}$,

$\Sigma_r = \begin{pmatrix} c^2_{\text{mz}} & 0 \\ 0 & \sigma^2_{\text{rt}} \end{pmatrix}$ – a model with fixed variance c^2_{mz} and unknown retention time variance σ^2_{rt} same for every cluster,

$\Sigma_r = \begin{pmatrix} c^2_{\text{mz}} & 0 \\ 0 & \sigma^2_{r\,\text{rt}} \end{pmatrix}$ – a model with fixed variance c^2_{mz} and unknown retention time variance $\sigma^2_{r\,\text{rt}}$ specific to each cluster.

The likelihood for a mixture of gaussian distributions can be expressed as:

$$f(x \mid \tau, \mu, \Sigma) = \prod_{p\in P} \sum_{r=1}^{R} \tau_r \frac{1}{2\pi \left|\Sigma_r\right|^{\frac{1}{2}}} \exp\left(-\frac{1}{2}(x_p - \mu_r)^{\mathrm{T}} \Sigma_r^{-1} (x_p - \mu_r)\right).$$

The MCLUST (Fraley and Raftery, 2002) package was used to estimate the parameters in the first four models.

Following (Fraley and Raftery, 2007), in the last three models we place an inverse gamma prior on σ^2_{rt} (or $\sigma^2_{r\,\text{rt}}$ independently for $r = 1, \ldots, R$) with hyperparameters $\frac{\nu_{\mathcal{P}}}{2}$ (shape) and $\frac{\xi^2_{\mathcal{P}\,\text{rt}}}{2}$ (scale) where $\nu_{\mathcal{P}} = 3$, $\xi^2_{\mathcal{P}\,\text{rt}} = \frac{\text{var}(x_{\cdot\,\text{rt}})}{R^2}$ (the empirical data variance along the retention time coordinate divided by the square of the number of components) and assume that μ_r come independently for $r = 1, \ldots, R$ from a Gaussian distribution with parameters $\mu_{\mathcal{P}}$ (mean) and $\frac{\Sigma_r}{\kappa_{\mathcal{P}}}$ (covariance matrix) where $\mu_{\mathcal{P}} = \frac{\sum_{p\in P} x_p}{|P|}$ (the empirical mean of the data), $\kappa_{\mathcal{P}} = 0.01$.

The weights $\tau_1, \ldots, \tau_R$ are also unknown. We use a uniform prior for them (i.e. a Dirichlet distribution with parameters $1, \ldots, 1$).

Our goal is to find the maximum a posteriori estimator for the τ, μ and Σ parameters.

Parameter estimation

Computing the parameters $\tau_1, \ldots, \tau_R, \mu_1, \ldots, \mu_R, \Sigma_1, \ldots, \Sigma_R$ that maximize the posterior density (MAP estimator) is rather problematic in case of the Gaussian mixture model. The Expectation-Maximization (EM) algorithm (Dempster et al., 1977; Minka, 1998) was designed to solve this kind of problems.

For conciseness in the next few paragraphs we will denote the peak data by x (i.e. $x_p = (\mathrm{mz}_p, \mathrm{rt}_p)$) and the model parameters by θ (i.e. $\theta_r = (\tau_r, \mu_r, \Sigma_r)$). Therefore the goal is to find:

$$\operatorname*{argmax}_{\theta} f(\theta \mid x) = \operatorname*{argmax}_{\theta} f(x \mid \theta) f(\theta),$$

where $f(\theta)$ is the prior density on θ. Alternatively we can of course search for:

$$\operatorname*{argmax}_{\theta} \left[\ln f(x \mid \theta) + \ln f(\theta) \right].$$

The EM algorithm requires pointing additional variables which will be denoted by z. In the case of gaussian mixtures it is convenient to define variables $z_p \in \{1, \ldots, R\}$ as the cluster (i.e. the peptide) peak $p \in \mathcal{P}$ belongs to.

We will use the equality $f(x \mid \theta) = \sum_z f(x, z \mid \theta)$ to introduce the z variable into the objective function, then introduce arbitrary weights $0 < w_z < 1$ such that $\sum_z w_z = 1$ and finally apply Jensen's inequality to the logarithm function in order to obtain a lower bound on the objective function:

$$\begin{aligned}
\ln f(x \mid \theta) + \ln f(\theta) &= \ln \sum_z f(x, z \mid \theta) + \ln f(\theta) \\
&= \ln \sum_z w_z \frac{f(x, z \mid \theta)}{w_z} + \ln f(\theta) \\
&\geq \sum_z w_z \ln \frac{f(x, z \mid \theta)}{w_z} + \ln f(\theta).
\end{aligned}$$

Note that the inequality above is valid for all parameters θ and weights w.

The EM algorithm is a procedure that starts with an arbitrary value of θ (or w) and iterates two steps:

- E-step – for fixed θ find w yielding the tightest bound (in fact this bound touches the objective function),
- M-step – for fixed w find θ maximizing the lower bound,

until convergence to a local maximum (it may depend on the starting point).

E-step

In order to perform the E-step we fix θ and optimize the lower bound over w. It is a constrained ($\sum_z w_z = 1$) maximization problem that can be solved[2] by introducing the Lagrange multiplier λ. For fixed z, differentiating the Lagrange function for the lower bound with respect to w_z yields:

$$\ln \frac{f(x,z \mid \theta)}{w_z} - w_z \frac{w_z}{f(x,z \mid \theta)} \frac{f(x,z \mid \theta)}{w_z^2} + \lambda = \ln \frac{f(x,z \mid \theta)}{w_z} + \lambda - 1.$$

Therefore the best lower bound is obtained for:

$$w_z = e^{\lambda - 1} f(x,z \mid \theta) = \frac{f(x,z \mid \theta)}{\sum_z f(x,z \mid \theta)} = \frac{f(x,z \mid \theta)}{f(x \mid \theta)} = f(z \mid x, \theta).$$

In the specific case of gaussian mixtures we have:

$$w_z = f(z \mid x, \theta) = \prod_{p \in P} f(z_p \mid x, \theta).$$

For fixed p and r we define (it will be needed in the M-step):

$$w_{pr} = \sum_{z\colon z_p = r} w_z = f(z_p \mid x, \theta)\Big|_{z_p = r} \tag{3.1}$$

$$\propto \tau_r \frac{1}{2\pi \left|\Sigma_r\right|^{\frac{1}{2}}} \exp\left(-\frac{1}{2} (x_p - \mu_r)^{\mathrm{T}} \Sigma_r^{-1} (x_p - \mu_r) \right), \tag{3.2}$$

which can be simply computed by evaluating the cluster densities at the data points.

M-step

For the purpose of lower bound optimization over θ for fixed w we maximize :

$$\sum_z w_z \ln f(x,z \mid \theta) + \ln f(\theta). \tag{3.3}$$

In the case of gaussian mixtures we postulate that points are drawn independently, i.e. $f(x,z \mid \theta) = f(x \mid z, \theta) f(z \mid \theta) = \prod_{p \in P} f(x_p \mid z_p, \mu_{z_p}, \Sigma_{z_p}) f(z_p \mid \tau) =$

[2] We need to keep in mind the $0 < w_z < 1$ inequalities, but the solution will turn out to satisfy them.

$\prod_{p \in P} f(x_p, z_p \mid \theta)$. Since we assume that (μ, Σ) and τ are independent a priori we have $f(\theta) = f(\tau) f(\mu, \Sigma)$ and the (3.3) expression can be rewritten as:

$$
\begin{aligned}
&\sum_z w_z \sum_{p \in P} \ln f(x_p, z_p \mid \theta) + \ln f(\theta) \\
=&\sum_z w_z \sum_{r=1}^{R} \sum_{p\colon z_p = r} \ln f(x_p, z_p \mid \theta) + \ln f(\theta) \\
=&\sum_{r=1}^{R} \sum_{p \in P} \sum_{z\colon z_p = r} w_z \ln f(x_p, z_p \mid \theta) + \ln f(\theta) \\
=&\sum_{r=1}^{R} \sum_{p \in P} w_{pr} \ln f(x_p, z_p \mid \theta)\Big|_{z_p = r} + \ln f(\theta) \\
=&\sum_{r=1}^{R} \left[\sum_{p \in P} w_{pr} \ln f(x_p \mid z_p, \mu_r, \Sigma_r)\Big|_{z_p = r} \right] + \ln f(\mu, \Sigma) \\
&+ \sum_{r=1}^{R} \sum_{p \in P} w_{pr} \ln f(z_p \mid \tau)\Big|_{z_p = r} + \ln f(\tau).
\end{aligned}
$$

Now maximization can be carried out independently for τ and (μ, Σ). The solution depends on the model and the priors used. In our case (see Section 3.2.1 for the description of the priors — in particular the κ, ξ and ν hyperparameters) for the last three models (not covered by the MCLUST package):

- since $\tau \sim \text{Dirichlet}\,(1, \ldots, 1)$ the corresponding subexpression to maximize is simply $\sum_{r=1}^{R} \sum_{p \in P} w_{pr} \ln \tau_r$ and the maximum is at:

$$\tau_r^* = \frac{\sum_{p \in P} w_{pr}}{\sum_{r=1}^{R} \sum_{p \in P} w_{pr}},$$

- maximization over the μ parameters yields:

$$\mu_r^* = \frac{\sum_{p \in P} w_{pr} x_p + \kappa_{\mathcal{P}} \mu_{\mathcal{P}}}{\sum_{p \in P} w_{pr} + \kappa_{\mathcal{P}}},$$

- maximization over the σ parameters yields (depending on the model):

$$\sigma^{2*}_{r\,\mathrm{rt}} = \frac{\sum_{p\in P} w_{pr}(x_{p\,\mathrm{rt}} - \mu^*_{r\,\mathrm{rt}})^2 + \kappa_{\mathcal{P}}(\mu^*_{r\,\mathrm{rt}} - \mu_{\mathcal{P}\,\mathrm{rt}})^2 + \xi^2_{\mathcal{P}\,\mathrm{rt}}}{\sum_{p\in P} w_{pr} + 1 + \nu_{\mathcal{P}} + 2}$$

or

$$\sigma^{2*}_{\mathrm{rt}} = \frac{\sum_{r=1}^{R}\sum_{p\in P} w_{pr}(x_{p\,\mathrm{rt}} - \mu^*_{r\,\mathrm{rt}})^2 + \kappa_{\mathcal{P}}\sum_{r=1}^{R}(\mu^*_{r\,\mathrm{rt}} - \mu_{\mathcal{P}\,\mathrm{rt}})^2 + \xi^2_{\mathcal{P}\,\mathrm{rt}}}{|P| + R + \nu_{\mathcal{P}} + 2}.$$

Ultimately the EM algorithm proposes a value θ^* as a guess for $\operatorname{argmax}_\theta f(\theta \mid x)$. The assignment of points to clusters is based on the probabilities $w^*_{pr} = f(z_p \mid x, \theta^*)\Big|_{z_p=r}$, i.e. point p is assigned to cluster $\operatorname{argmax}_{r=1,\ldots,R} w^*_{pr}$. Moreover, these values can be treated as certainty measure — if one is interested in high-quality clusters one can sieve out elements with low probabilities.

Number of clusters

The issue of choosing the number of clusters and the appropriate model parametrization can be reduced to the problem of model selection. Bayesian Information Criterion (BIC) (Haughton, 1988; Schwarz, 1978) for model M is defined as:

$$\mathrm{BIC} = -2\ln f_M(x \mid \theta^*_M) + \#_M \ln n$$

where:

- $\#_M$ is the number of parameters of the model M (it depends on the number of clusters),
- θ^*_M is the MAP estimate[3] of the parameters of the model M,
- n is the number of observations in the data.

In order to choose the number of clusters one fits many models, each with different number of clusters, and picks the one with the lowest BIC value.

[3] Usually it is the MLE estimate. Fraley and Raftery (2007) proposed to replace it with the MAP estimate.

3.2.2 DBSCAN algorithm

The DBSCAN algorithm (*Density-Based Spatial Clustering of Applications with Noise*) (Ester et al., 1996) was designed for finding clusters in spatial data, such as satellite images and protein structure data (Ester et al., 1996; Ng and Han, 1994). This algorithm applies a local clustering criterion — clusters are defined as dense regions in the data space which are separated by regions of low object density, called *noise*. Each cluster can have an arbitrary shape and the objects inside a cluster region may be arbitrarily distributed.

In order to describe what the DBSCAN algorithm produces, three definitions are introduced:

1. a point q is *directly density-reachable* from a point p if it is in p's ϵ-neighborhood, and if p is surrounded by sufficiently many points (at least *minPts*) such that one may consider p and q to be part of a cluster,

2. a point q is *density-reachable* from a point p if there is a sequence of points starting with p and ending with q such that the next point is directly density-reachable from the previous point,

3. two points p and q are *density-connected* if there is a point o such that o and p as well as o and q are density-reachable.

The notion of density-connected points is introduced since the relation of density-reachable is not symmetric (because q might not have sufficiently many neighbors to be considered a cluster element).

A subset of points is called a cluster if it satisfies two properties:

- all points within the subset are mutually density-connected,

- all points density-connected to any point of the subset are part of the subset as well.

Note that the resulting clustering depends on two parameters – ϵ and *minPts* through the definition of points being directly density-reachable.

3.2.3 Retention time correction

Retention time deviations are often too significant for the correct peak alignment be possible. Due to this fact Smith et al. (2006) propose an iterative procedure that comprises multiple steps of peak alignment alternated with the retention time correction. This procedure is implemented in the XCMS package (Smith et al., 2006).

In this method for retention time correction it is assumed that in the input peak alignment one can find a number of reasonable peak clusters called *well behaved groups*. Those groups have at most one observation from the majority of samples (the number defining the ”majority” is a parameter). It is very probable that such clusters are properly matched and can be used as standards for the correction. The well behaved groups are usually distributed evenly over the significant portions of the retention time. For each such cluster the median retention time is calculated. Apart from this, deviation from this median is also computed for each sample. Since it may happen that two peaks with similar retention times and different masses show different deviations, authors approximate the deviations using a local regression fitting method *loess* (Cleveland et al., 1992) which uses segmented low-order polynomial. Subsequently the fitted functions are used to correct the retention times of the original peaks.

There are many other approaches to the problem of retention time correction (see e.g. (Jaitly et al., 2006)). We have decided to use the procedure implemented in the XCMS package due to its availability and to make a fair comparison to the XCMS peak alignment method.

3.2.4 Feature selection and False Discovery Rates

A few concepts related to feature selection and classification problems will be introduced here briefly. They will help in validation of the peak clustering results. An idea will be explored that properly aligned mass spectra enable further reasoning (e.g. classification of mass spectra from healthy donors and diseased patients).

T-test

The most commonly used test for location problems is the t-test. It comes in several variants differing in the assumptions about the distributions being compared.

We will use the independent two sample unequal variance version. Assume that points $x_1, \ldots, x_n$ and points $y_1, \ldots, y_m$ are independent draws from two normal distributions. The null hypothesis is that the two distributions' means are equal and the statistic is:

$$\frac{\overline{x} - \overline{y}}{\sqrt{\frac{s_x^2}{n} + \frac{s_y^2}{m}}}, \tag{3.4}$$

where $\overline{x} = \frac{\sum_{i=1}^{n} x_i}{n}$ is the estimator of the mean, $s_x^2 = \frac{\sum_{i=1}^{n}(x_i - \overline{x})^2}{n-1}$ is the unbiased estimator of the variance. The value of this statistic can be used to rank features.

Random Forest feature selection

Random Forest (Breiman, 2001) is an example of a classification meta-algorithm that employs an ensemble of a sufficiently large number of possibly weak yet independent classifiers. In case of Random Forest the base classifier is a decision tree.

Decision tree is a binary tree with internal nodes labeled by a feature and a threshold. Leaves of decision tree are labeled by the class. A data point that is to be classified travels from the root to a leaf at each node heading left or right depending on the feature exceeding the threshold. In the Random Forest algorithm the class of a data point is determined by voting of a decision tree set.

Each decision tree in the Random Forest algorithm is built independently according to the same iterative procedure. The basic step of this procedure is leaf splitting (it will be described in a moment). Let N be the total number of points (i.e. from both classes), M be the number of features. Fix $m << M$. In order to grow a decision tree the following steps are conducted:

- sample N data points with replacement as a training set for the tree,
- start with a single node tree (the node corresponds to the whole training set),

- repeat until no leaf needs splitting:
    - pick a leaf that needs splitting,
    - select m features randomly,
    - split the node using the best feature (out of m selected) and threshold.

In order to split a leaf one considers points in the training set that reach this leaf. The leaf needs splitting only if more than one class has a representative in this point set. The splitting consists of picking the feature and the threshold that discern the classes best according to a criterion (e.g. Gini impurity, information gain).

For each decision tree the set of points not used to build the tree is called the *out-of-bag* set and can be used to assess feature importance. Out-of-bag points are put down the tree and percentage of votes for the correct class is computed. For each feature, the same procedure is repeated but with the values of this feature permuted randomly. The difference between the percentages in both cases (original and permuted) is used to rank the features.

False Discovery Rates

Feature selection is usually conducted by scoring attributes individually with a statistic. We use the t-test and Random Forest feature scoring for this purpose. Then a particular level of the score is chosen as a threshold and attributes exceeding this threshold are declared significant. However, the issue of selecting an appropriate threshold is problematic. It is hard to assess what level yields statistically significant selections, i.e. the ones that are unlikely to occur by chance.

The first thing that needs to be defined is the type I error we wish to control in the multiple hypotheses testing scheme. Here we use the False Discovery Rates (FDR) (Benjamini and Hochberg, 1995) which can be defined as:

$$\mathrm{FDR} = \mathrm{E}\left[\frac{V}{R}\right], \tag{3.5}$$

where:

- V is the number of rejected true null hypotheses,

- R is the total number of rejected null hypotheses.

Since there is a problem with this definition when $R = 0$ the FDR is declared as 0 in that case[4].

In order to estimate the FDR we use the procedure from (Storey and Tibshirani, 2003). Let t be the critical value for the statistic (i.e. if the statistic for a feature exceeds t we select that feature). Let T be the number of features with statistics exceeding t. The decision attribute values are randomly permuted R times and the same number (number of features with statistics exceeding t) is computed for each permutation. Let us denote those numbers by T_i where $i = 1, \ldots, R$. The FDR estimator is given by the formula:

$$\widehat{FDR}(t) = \frac{\sum_{i=1}^{R} T_i}{TR}.$$

The t threshold can subsequently be chosen for which the FDR value is sufficiently low.

3.2.5 Data set

Data was provided by the Mass Spectrometry Laboratory from the Institute of Biochemistry and Biophysics of Polish Academy of Sciences. The mass spectrometer used in the experiments was an ElectroSpray Ionization Fourier Transform Ion Cyclotron Resonance (ESI-FTICR) coupled with an HPLC retention column.

The data set comprised mass spectra acquired from plasma samples of colorectal cancer patients. Apart from the patient data, control samples were also collected from healthy donors and analyzed with the mass spectrometer. The colorectal cancer data set consisted of 40 spectra, 23 samples corresponding to patients and 17 to healthy donors. Blood samples were collected from patients and age-matched healthy controls. For plasma collection K3 E (Greiner Bio-One Cat. No. 455036) tubes supplemented with EDTA were used and after collection samples were centrifuged at 2800 g for 15 min at 4 °C. Obtained plasma was aliquoted in 200 µl portions, frozen in liquid nitrogen and stored at −70 °C for further use. For analysis, plasma aliquots were centrifuged through a 5 kDa

[4] Also positive FDR (pFDR) was proposed in (Storey, 2003) as $\mathrm{E}\left[\frac{V}{R} \mid R > 0\right]$.

or 30 kDa (or both) cutoff filtration membrane (Millipore Ultrafree-MC) in the presence of 20% acetonitrile as a chaotropic agent. Membrane was thoroughly washed with 25% acetonitrile prior to use. To the filtrate the internal standard was added. An HPLC purified peptide (200 pg in each experiment) obtained from tryptic digest of lysozyme (FESNFNTQATNR, molecular mass 1428.65 Da) was used as an internal standard.

The raw data in mzXML file format was preprocessed using the XCMS package (Smith et al., 2006) from the Bioconductor project (Gentleman et al., 2004). We used *mz2m* (Gambin et al., 2007), a program for mono-isotopic peak detection, to obtain a list of peak coordinates, i.e. mass-to-charge ratios and retention times of the most abundant molecules. In total, there were 155294 mono-isotopic peaks detected in 40 samples. The time range of detected peaks was 922.6 s to 4871.3 s (15.4 min to 81.2 min) and the mass-to-charge ratio range was −1499.33 Th to −300.127 Th (c.f. Fig. 3.5).

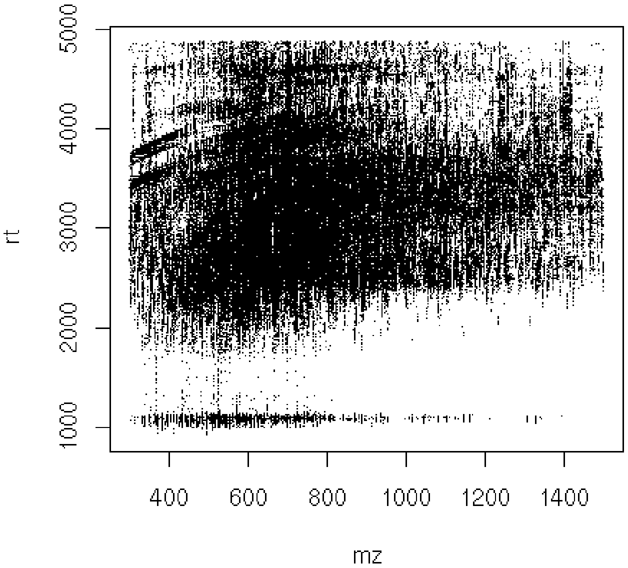


Figure 3.5: Colorectal cancer data, peaks from 40 samples presented as points in two-dimensional space. Retention time dimension units are seconds.

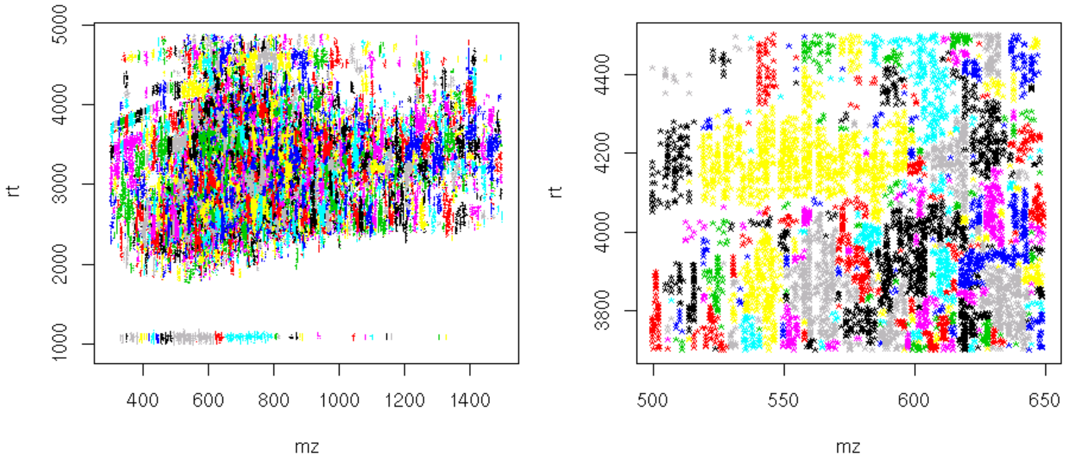


Figure 3.6: Colorectal cancer data clustered with the DBSCAN algorithm, $\epsilon_m = 5$, $\epsilon_{rt} = 30$, $minPts = 10$. Picture on the right presents fragment of the data in greater detail. The cluster colors are recycled.

3.2.6 Results

We tested our approach on the LC-MS dataset. The DBSCAN procedure was run with the following parameters:

- $\epsilon_{\text{mz}} = 5$, $\epsilon_{\text{rt}} = 30$ — describing neighborhood size, i.e. point p is point's q ϵ-neighbor if and only if $|x_{p\,\text{mz}} - x_{q\,\text{mz}}| < \epsilon_{\text{mz}}$ and $|x_{p\,\text{rt}} - x_{q\,\text{rt}}| < \epsilon_{\text{rt}}$,
- $minPts \in \{0, 10\}$ — the minimal number of points in the neighborhood needed to form a cluster,
- the upper limit for size of a preliminary cluster was 1000 elements.

In case of $minPts = 0$ no peaks are treated as noise, even the ones in the very sparse regions. For $minPts = 10$ we assume that some of the peaks might have noise origins. Of course another explanation for their origins can be that they in fact correspond to real peptides, but the retention time drift was so big, that they cannot be aligned to any peaks in this iteration. Hopefully, excluding them at this stage does not necessarily mean they they will never be properly aligned.

If retention time correction step (discussed in Section 3.2.3) is performed after the clustering step the drifts might become smaller.

There were 9026 preliminary groups obtained with parameter $minPts = 0$ and 8216 with $minPts = 10$ (c.f. Fig. 3.6). In the latter case 3076 points were marked as noise.

All the models were fitted within each of the preliminary clusters in the second stage of the algorithm. At one time the same model was assumed in all the preliminary clusters. Hence, model selection problem was solely to select the appropriate number of clusters, the one that minimizes the BIC value. For a preliminary cluster of size n ($1 \leq n \leq 1000$) clusterings of the number of clusters from interval $[n/40, n/10]$ were compared (40 is the number of samples).

In case of the models with fixed covariance matrix, the standard deviation on mass-to-charge ratios (c_{mz}) was set to 0.04, which was selected after consultations with experienced mass spectrometer operators. We tested fixed retention time deviations (c_{rt}) of 50, 100 and 200 seconds. Apart from the mentioned models, we also fitted both models where the retention time deviation was estimated by the algorithm.

The initial cluster assignments that are being improved with the EM algorithm were obtained with the model-based hierarchical algorithm. Implementation from the MCLUST (Fraley and Raftery, 2002) R package was used. The model assumed in the hierarchical algorithm was λI which stands for identical, spherical clusters. The original clusters are rather ellipsoidal than spherical and hence model λB, identical diagonal clusters, would be more appropriate but was not implemented. To overcome this problem, the dimensions, which are given in different units, had to be properly scaled. It was established with empirical tests that dividing the retention time values by 100 results in reasonable clusters.

3.2.7 Visual validation

We show an example of a preliminary cluster obtained from the DBSCAN algorithm and different clusterings resulting from different parametrizations (see Fig. 3.7). Peaks corresponding to the same peptide are expected to form elongated groups along the retention time axis with rather small variance along the mass-to-charge ratio axis. One can see that the less constrained models detected clusters that should not occur in nature — elongated along the mass-to-charge ratio axis. The remaining models had fixed mass-to-charge ratio deviation and hence the clusters look more as expected. The differences between the three models with fixed retention time deviations (50, 100 and 200 seconds) are small on the whole set and they do not differ much in case of this subset. The clusterings of the last two models also share some clusters with those three, however they are more similar to each other than to any other model.

3.2.8 Classification based validation

The quality of alignments was also evaluated by assessing the *False Discovery Rates* (FDR) for two feature selection methods. These methods were applied in order to pick the features that discern samples from two groups well. The idea underlying this procedure was that properly aligned mass spectra enable further reasoning while random-like aligned mass spectra should contain less biologically relevant information.

FDR comparison of the models

The goal of feature selection is to extract aligned peaks, that best discern classes of samples.

In all the experiments reported here there were 500 permutations performed for the t-test. In case of the Random Forest based feature selection there were 100 permutations performed. For each model there were 1000 trees grown, each time the m parameter was equal to the square of the total number of attributes.

Figure 3.8 shows FDR plots for each of the clusterings for the t-test and Random Forest based feature selection. For a given statistic score threshold (horizontal axis) one prefers methods with lower FDR.

The differences between models are small. Moreover, t-test and Random Forest model rankings are not consistent. The λB_k model is best in the t-test ranking which is surprising since according to the visual analysis (Fig. 3.7) it produces strange looking clusters. On the other hand the Random Forest ranking

parameter name	description	value set
`minfrac`	minimum fraction of samples necessary in at least one of the sample groups	0 (to avoid classification bias)
`minsamp`	minimum number of samples necessary in at least one of the sample groups	0
`bw`	bandwidth (standard deviation) of gaussian smoothing kernel applied to the peak density chromatogram	default value
`mzwid`	width of overlapping mass-to-charge ratio slices used for creating peak density chromatograms and grouping peaks across samples	$0.01, 0.02,$ $0.04, 0.08,$ $0.10, 0.20,$ 0.50
`max`	maximum number of groups identified in a single mass-to-charge ratio slice	default value

Table 3.1: Different parametrizations for the XCMS clustering method.

seems consistent with the visual analysis — the models with the mass-to-charge ratio deviation fixed outperform the rest.

FDR comparison with the XCMS package with retention time correction

We compared performance of alignments acquired with our method to the grouping proposed in the XCMS package (Smith et al., 2006). In the XCMS algorithm we used several parametrizations as listed in Table 3.1. For both methods we applied the same filtering criterion: clusters that did not contain at least 3 peaks were sieved out. To estimate the effect of retention time drift on the quality of clustering we have repeated all experiments with one iteration of the retention time correction.

We performed one iteration of the retention time correction with all the models. The same procedure was applied for the XCMS clustering for different

mzwid	**Before ret. time corr.**		**After ret. time corr.**	
	Number	Fraction	Number	Fraction
0.01	60187	0.388	57942	0.373
0.02	52510	0.338	50145	0.323
0.04	45614	0.294	42790	0.276
0.08	40412	0.26	38204	0.246
0.1	38632	0.249	36370	0.234
0.2	33756	0.217	31925	0.206
0.5	30232	0.195	29722	0.191

Table 3.3: Number of rejected peaks for the XCMS clustering results for different values of the mzwid parameter.

parametrizations (see Table 3.1). The FDRs for the t-test and Random Forest based feature selection after the retention time correction are illustrated in the lower part of Fig 3.9.

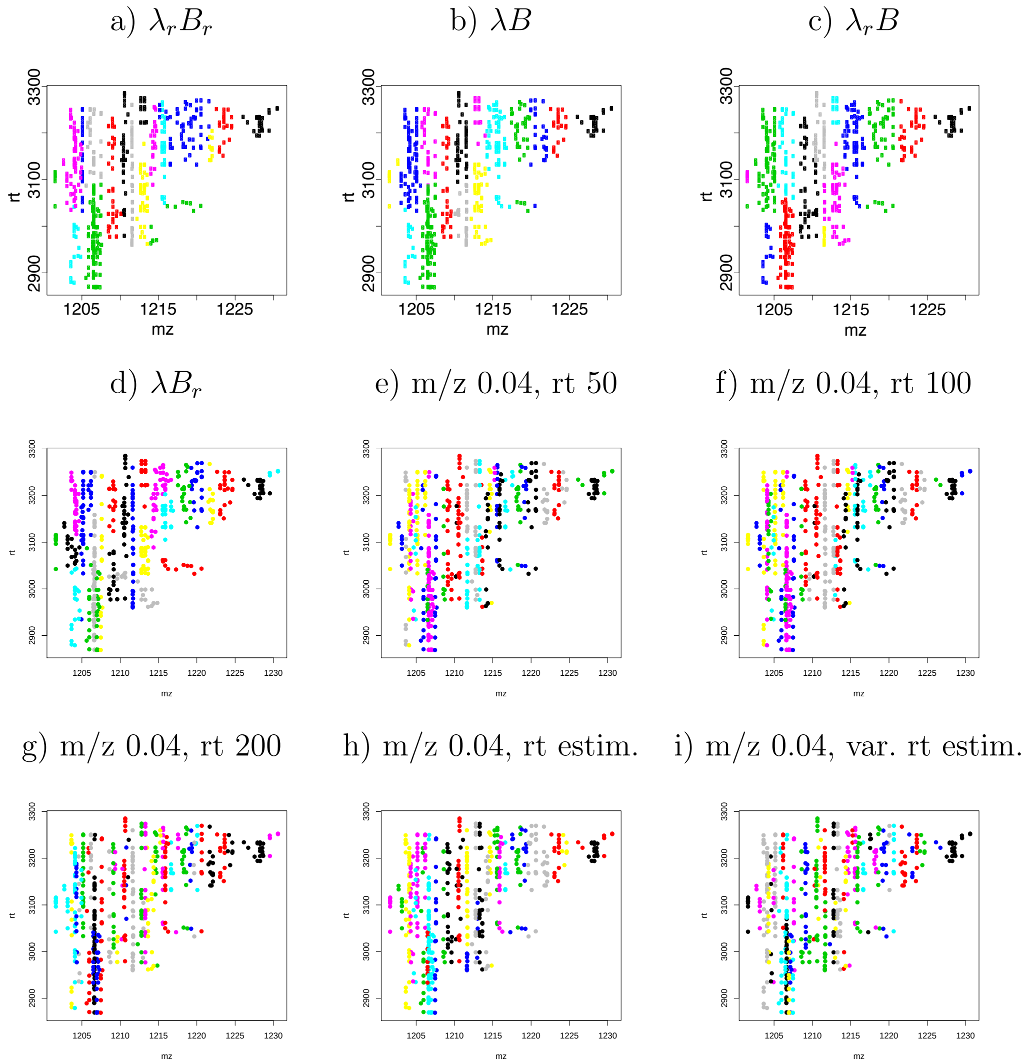
Model	**Before ret. time corr.**		**After ret. time corr.**	
	Number	Fraction	Number	Fraction
a	19858	0.128	14558	0.094
b	25805	0.166	21376	0.138
c	22261	0.143	18398	0.118
d	20950	0.135	16182	0.104
e	29376	0.189	23214	0.149
f	29366	0.189	21502	0.138
g	29326	0.189	20741	0.134
h	29909	0.193	23092	0.149
i	30029	0.193	22353	0.144

Table 3.2: Number of rejected peaks. Explanation of model names: a) $\lambda_k B_k$, b) λB, c) $\lambda_k B$, d) λB_k, e) – i) all the models have deviation in the mass-to-charge ratio dimension fixed to 0.04, retention time deviations are: e) 50, f) 100, g) 200, h) estimated from the data, the same for every cluster, i) estimated from the data for each cluster.

Once again the differences are small. The number of clusters produced by all the methods is roughly on the same level (data not shown) but significantly more peaks are filtered out in case of the XCMS than the other methods (see Tables 3.2 and 3.3).

Conclusions

Overall, models with both mass-to-charge ratio and retention time deviations fixed seem to be the best choice. They produce reasonably looking clusters and are ranked well with the Random Forest based feature selection FDRs. Moreover they outperform the XCMS package with less peaks being filtered out.



a) $\lambda_r B_r$

b) λB

c) $\lambda_r B$

d) λB_r

e) m/z 0.04, rt 50

f) m/z 0.04, rt 100

g) m/z 0.04, rt 200

h) m/z 0.04, rt estim.

i) m/z 0.04, var. rt estim.

Figure 3.7: Example of clustering acquired for one of the preliminary clusters from Figure 3.6 of size 999. a) – i) stand for fitted models: a) $\lambda_r B_r$, b) λB, c) $\lambda_r B$, d) λB_r, e) – i) models have deviation in the m/z dimension fixed to 0.04, retention time deviations are: e) 50, f) 100, g) 200, h) estimated from the data, the same for every cluster, i) estimated from the data for each cluster.

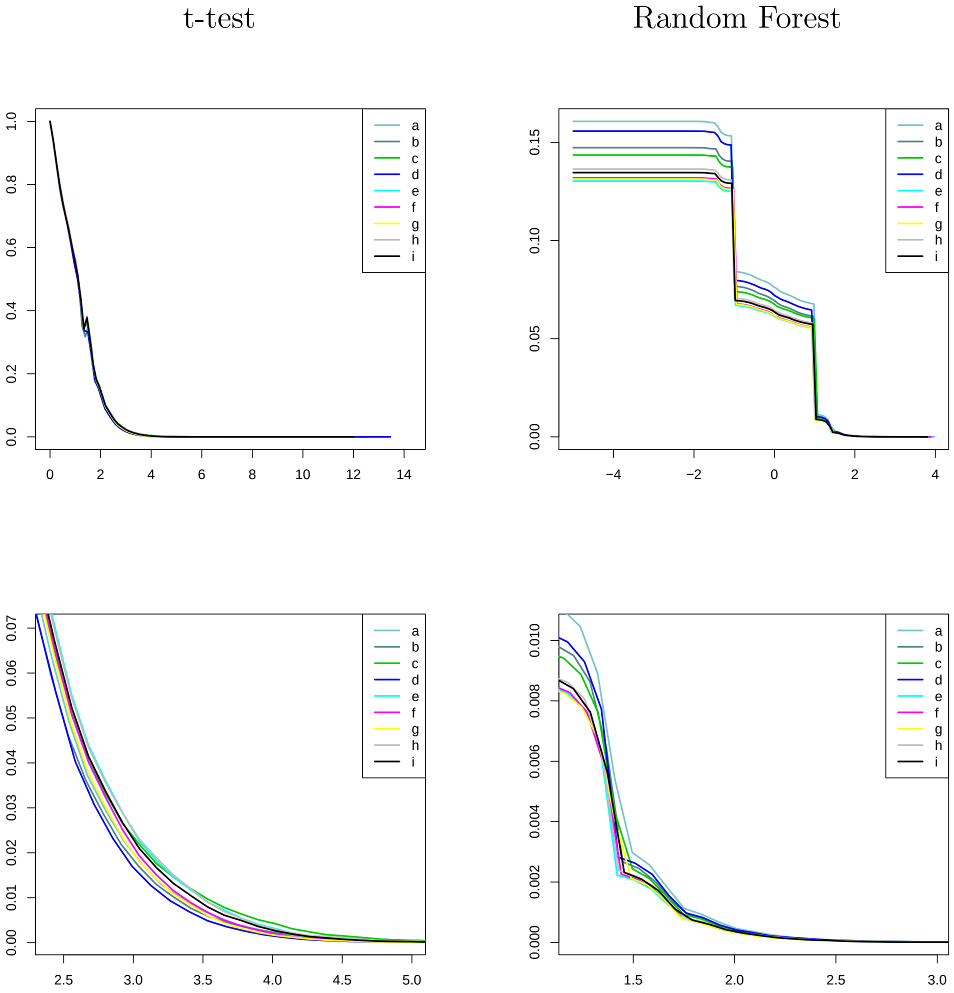


Figure 3.8: The FDR statistic computed for the t-test and the Random Forest based feature selection. The horizontal axis shows different values of the score threshold, the vertical axis shows the FDR values. a) – i) stand for fitted models: a) $\lambda_k B_k$, b) λB, c) $\lambda_k B$, d) λB_k, e) – i) all the models have deviation in the mass-to-charge ratio dimension fixed to 0.04, retention time deviations are: e) 50, f) 100, g) 200, h) estimated from the data, the same for every cluster, i) estimated from the data for each cluster.

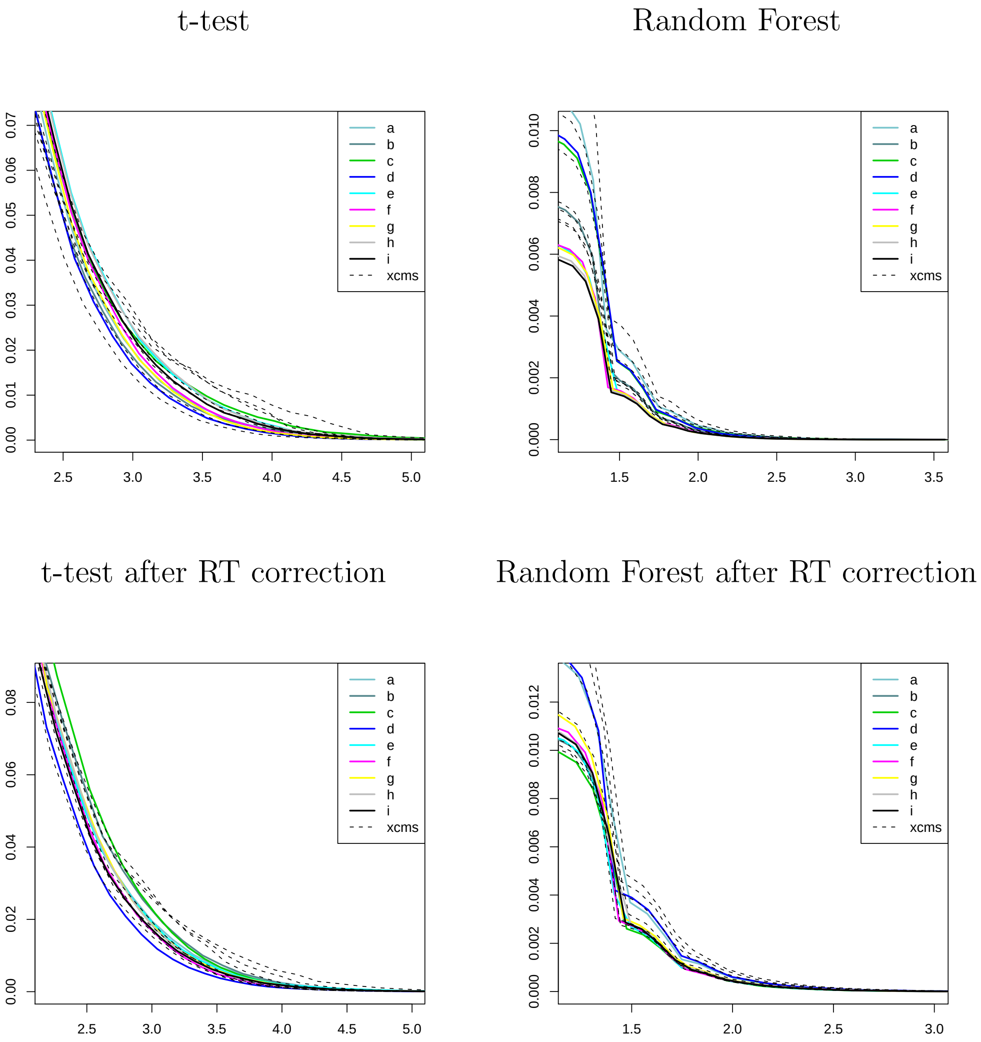


Figure 3.9: The FDR statistic computed for the t-test and the Random Forest based feature selection. Lower plots show the FDRs for alignments after the retention time correction. Upper plots show the FDRs before the retention time correction for comparison. The horizontal axis shows different values of the score threshold, the vertical axis shows the FDR values. Explanation of model names: a) $\lambda_k B_k$, b) λB, c)$\lambda_k B$, d) λB_k, e)–i) all the models have fixed deviation of the mass-to-charge ratio dimension to 0.04, retention time deviations are: e) 50, f) 100, g) 200, h) estimated, the same for all clusters, i) estimated, varying between clusters. The outcome of the XCMS clustering (with different parametrizations) is also plotted.

Chapter 4

Proteolytic activity modelling

With basic spectra processing problems solved one may begin thinking about biologically meaningful questions. One area of interest might be *peptidase* activity. Peptidases (or *proteases*, *proteinases*) are enzymes that break peptide bonds linking amino acids in a peptide chain. The discovery of the causative genetic underpinnings of cancer has been a focus of biomedical research for decades, but the multigenic nature of cancer has hindered progress in understanding the underlying mechanisms that lead to a specific disease phenotype. The contribution of proteolysis in processes of tumor invasion and metastasis has been recognized many years ago and the overall consensus is that protease biology represents a fertile ground for advances that will be clinically useful (Matrisian and Sledge, 2003). Using peptide degradation pattern for the diagnostic purposes seems biologically sound as the amount of peptides in the circulation changes dynamically according to the physiological or pathological state of an individual. Moreover, it was reported that the degradation enzymes affect the dynamics of signaling pathways (Reznik and Fricker, 2001). The objective of this chapter is to develop mathematical tools and computational methods to describe proteolytic activity.

Modern biotechnology offers efficient techniques for large-scale measurements of molecular activity characterizing numerous cellular processes. Besides experimental techniques for high-throughput analysis, various formal methods and algorithmic approaches were proposed for molecular modeling (Turner et al., 2004): directed graphs (particles are at the nodes, reactions are edges), Bayesian networks (the nodes correspond to random variables describing e.g. gene expression levels), boolean networks (the objects under the study can be in either the active or inactive state), ordinary differential equations (Guldberg-Waage law) and partial differential equations (taking the cell space structure into account). The stochastic dynamics of a mixture of molecular species interacting through different biochemical reactions can be accurately modelled by the chemical master equation (CME) (Pahle, 2009), i.e. the system of differential equations describing the evolution of a stochastic process.

All those formalisms are quite general (can describe many kinds of reactions), but their computer implementations are not efficient enough for systems with large numbers of molecules. This obstacle is crucial for stochastic modeling, where the standard approach to solving the CME is via computationally expensive simulations (Pahle, 2009). The analytic solutions of the CME were obtained only in very limited cases of a monomolecular reaction systems (Jahnke and Huisinga, 2007) (all reactions of the form $A \to B$), or a slightly more complex system allowing binary reactions (Gelenbe, 2008) (binary reactions have the form $A + B \to C$).

While modeling proteolytic activity one has to consider substrings of a peptide as a result of cutting. Thus a single protein can give rise to tens of thousands of molecule kinds (if every substring needs to be taken into account). Also the presence of splitting reactions prevents from obtaining analytic solutions. The authors of (Moles et al., 2003) discuss parameter fitting methods in differential equations models and emphasize the computational difficulty of the problem.

Therefore, even though there exists a large body of research concerning modeling enzymatic reaction systems with differential equations, see e.g. (Ciliberto et al., 2007), the stochastic framework is rarely considered. A rare exception is the model proposed recently in (Goldobin and Zaikin, 2009) for the problem of protein degradation in macromolecular complex called proteasome.

Organization of the chapter

Section 4.1 is based on (Kluge et al., 2009). To our knowledge it was the first formal approach to modeling *exopeptidase* (peptidases that cut only one amino acid from an end of a peptide chain) activity from liquid chromatography mass spectrometry samples. A statistical model of peptidome degradation is designed and a Metropolis-Hastings algorithm for Bayesian inference of model parameters is proposed. The model is successfully validated on a real LC-MS dataset. Our findings support the hypotheses about disease-specific exopeptidase activity, which can lead to new diagnostic approach in clinical proteomics.

Section 4.2 is based on (Gambin and Kluge, 2010). It generalizes the model from Section 4.1 by considering cuts at arbitrary sites of peptide chains (i.e. in addition to exopeptidases also *endopeptidases* are handled). Moreover, proteolytic activity is modeled in time by studying the evolution of the underlying stochastic process before reaching equilibrium (we no longer need to assume a constant flow of long peptide sequences into the system). The model uses peptidase cleavage pattern data from the MEROPS database (Rawlings and Barrett, 2000) and is tested on a simulated dataset.

4.1 Stationary model for exopeptidase activity

With the development of proteomic analytic technologies, especially mass spectrometry (MS), great hopes for early diagnostics of cancer were expressed (Petricoin et al., 2002). However, the initial optimism has encountered strong criticism. The criticism was addressed not against the idea of using protein profiles as a diagnostic tool but against poor quality of data obtained from SELDI type detectors and non-reproducibility of experimental conditions (Diamandis, 2003, 2004).

Moreover, despite years of intensive MS analysis, only a small number of proteins have been validated as cancer biomarkers. Also the MS samples where characterized as highly unstable, mainly because of ex-vivo proteolytic processing (Marshall et al., 2003; Verrills, 2006). Changes in protein profiles can be generated simply by the amount of time between sample draw and analysis. Surprisingly this obstacle gives rise to a completely new approach enthusiastically described as “spinning biological trash into diagnostic gold” (Liotta and Petricoin, 2006).

In (Diamandis, 2006) the advantages and limitations of clinical peptidomics were summarized. The authors proposed to characterize the proteolytic activity, as it could lead to better patient discrimination. Therefore our research objective was to build a mathematical model of exopeptidase activity and to check whether the model exhibits differences between samples from healthy donors and diseased patients.

In a typical LC-MS experiment a complex mixture of peptides is separated using liquid chromatography coupled on-line with electrospray mass spectrometer. After appropriate preprocessing (see Chapters 2 and 3) each detected peptide is characterized by two coordinates – its molecular mass-to-charge ratio and retention time value.

Much work has already been invested into detection of molecular mass biomarkers for various pathologies and diagnostic procedures have been suggested (Adam et al., 2002; Geurts et al., 2005; Jacobs and Menon, 2004; Li et al., 2002; Lilien et al., 2003; Tibshirani et al., 2004; Wu et al., 2003; Yu et al., 2005). Unfortunately, it is extremely hard to obtain stable MS results reproducible over time and across different laboratories (Hu et al., 2005). Often the differences in sample collection or sample handling protocol affect the proteome to a degree that can dominate biological changes. Also the ex-vivo peptide degradation process was regarded as a serious obstacle in MS analysis.

Recently a novel way of diagnosing cancer was suggested in (Villanueva et al., 2006a,b). The authors postulate, that the diagnostic peptides originate after ex-vivo exoproteolytic processing of high abundance protein fragments. Paradoxically, these findings indicate that inhibition of proteolysis in ex-vivo samples could limit biomarker discovery. See also (Koomen et al., 2005) for the information on the peptidome degradation process analyzed with the use of mass spectrometry technology.

Using peptide degradation pattern for the diagnostic purposes seems biologically sound as the amount of peptides in the circulation changes dynamically according to the physiological or pathological state of an individual. Moreover, it was reported that the degradation enzymes (especially exopeptidases) affect the dynamics of signaling pathways (Reznik and Fricker, 2001).

Even though there exists a large body of research concerning modeling enzymatic reaction systems with differential equations, see e.g. (Ciliberto et al., 2007), to the best of our knowledge this work is the first attempt to build a model specifically with exopeptidase activity in mind.

Results summary

We propose a comprehensive statistical and computational framework for analysis of peptide degradation patterns in LC-MS samples. In our approach the exopeptidase activity is modeled as a continuous time Markov process. The stationary distribution of this process is proved to be a product of Poisson laws. A Metropolis-Hastings (Hastings, 1970) sampler is implemented to estimate the parameters of the model. These correspond to the rates of cleavage for different amino acids. The model is tested on simulated data and validated on a colorectal cancer dataset. Parameter estimates for diseased patients and healthy donors differ significantly and allow for accurate classification. Moreover, the estimated differences in activity of proteolytic enzymes in cancer and healthy samples correlates with experimentally verified activity of metallopeptidases in colorectal cancer development (Leeman et al., 2003; Masaki et al., 2001). The scheme of data processing and analysis workflow is depicted in Fig. 4.1.

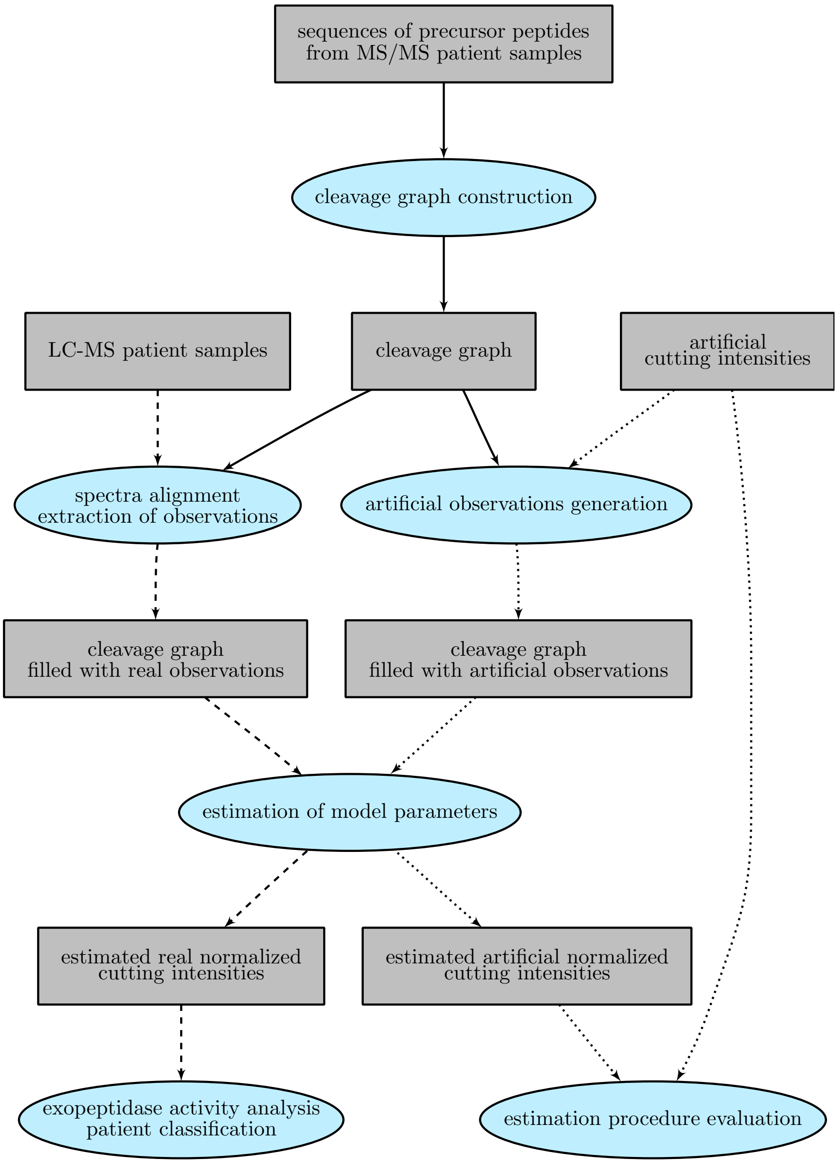

sequences of precursor peptides from MS/MS patient samples
cleavage graph construction
LC-MS patient samples
cleavage graph
artificial cutting intensities
spectra alignment extraction of observations
artificial observations generation
cleavage graph filled with real observations
cleavage graph filled with artificial observations
estimation of model parameters
estimated real normalized cutting intensities
estimated artificial normalized cutting intensities
exopeptidase activity analysis patient classification
estimation procedure evaluation

Figure 4.1: Data processing and analysis workflow.

Availability

The source code (R with C) of our estimation procedure is freely available at `http://bioputer.mimuw.edu.pl/papers/exopep`. The site also contains additional figures and peptide sequences generating the cleavage graph.

4.1.1 Model description

Our model has two main components: the first one describes the cleavage (peptide degradation) process itself, while the second accounts for imperfections at the data acquisition stage.

Model for the cleavage process

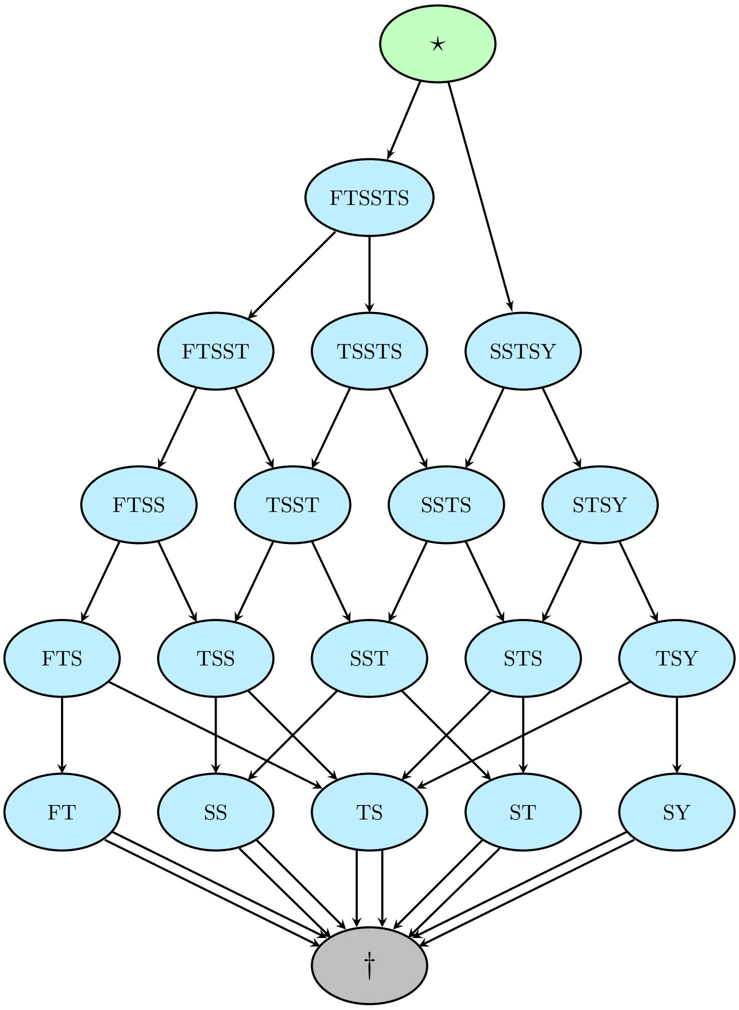


Figure 4.2: The cleavage graph for 2 precursor peptides FTSSTS and SSTSY with source and sink nodes added.

Peptide sequences whose proteolysis we wish to model give rise to a graph $(\mathcal{V}, \mathcal{E})$, which we will call the *cleavage graph*. Nodes $\mathcal{V}$ of this graph correspond to all peptide subsequences of length at least 2. A directed edge from node i to j is placed if subsequence j can be obtained from subsequence i by cutting off a single amino acid from the N-terminus or the C-terminus. Each edge is labeled with the amino acid being cut off and the terminus it is being cut off from, thus the set $\mathcal{R}$ of possible labels has 20×2 elements. The label for edge $i \to j$ is denoted by $r(i,j)$. We assume that the labeling and structure of the cleavage graph is known. An exemplary cleavage graph is presented in Fig. 4.2.

It is helpful to think of the peptide subsequences as particles placed at nodes of the cleavage graph and moving along its edges. Then the probabilistic dynamics of the cleavage process is described by the following intensities of transition:

- particles are created at node i with intensity $a_{\star i}$,
- every particle placed at i can move to j with intensity $a_{r(i,j)}$ independently of all other particles, provided that there exists an edge $i \to j$,
- every particle placed at i can be annihilated with intensity $a_{i\dagger}$ independently of all other particles.

We refer to the $(a_r)_{r\in\mathcal{R}}$ parameters as the *cutting intensities.*

More formally, let random variable $X_i(t)$ denote the number of particles at node $i \in \mathcal{V}$ at time t and write $X(t) = (X_i(t))_{i\in\mathcal{V}}$. We regard $(X(t), t \geq 0)$ as a homogeneous Markov process in the space of configurations $x = (x_i)_{i\in\mathcal{V}}$, $x_i \in \{0, 1, \dots\}$. We use the standard notation for restricted configurations, writing e.g. $x_{-i} = (x_k)_{k\in\mathcal{V}\colon k\neq i}$. The process has the following intensities of transition ($x \neq x'$):

$$Q(x,x') = \begin{cases} a_{\star i} & \text{if } x'_i = x_i + 1,\ x'_{-i} = x_{-i} \text{ for some } i, \\ a_{r(i,j)}x_i & \text{if } x'_j = x_j + 1,\ x'_i = x_i - 1, \\ & \quad \text{and } x'_{-i-j} = x_{-i-j} \text{ for some } i \to j, \\ a_{i\dagger}x_i & \text{if } x'_i = x_i - 1,\ x'_{-i} = x_{-i} \text{ for some } i. \end{cases}$$

We assume that the process reached the equilibrium state. At each node, we are interested in the distribution of the number of particles. Perhaps surprisingly, we can prove that those numbers are independent and each one follows a Poisson distribution.

Proposition 1 (Equilibrium distribution) *The process $(X(t))$ has the equilibrium (stationary) distribution π given by:*

$$\pi(x) = \prod_{i \in \mathcal{V}} \mathrm{e}^{\lambda_i} \frac{\lambda_i^{x_i}}{x_i!},$$

where the configuration of intensities $(\lambda_i)_{i \in \mathcal{V}}$ is the unique solution to the following system of "balance" equations:

$$\sum_{k \to i} \lambda_k a_{r(k,i)} + a_{\star i} = \lambda_i \left(\sum_{i \to j} a_{r(i,j)} + a_{i\dagger} \right) \quad \textit{for every } i \in \mathcal{V}.$$

Note that it is easy to solve the system of "balance" equations recursively starting from the nodes without parents. The proposition can be proved by simply checking the global balance condition (i.e. that for every configuration x the equality $\sum_{x' \neq x} \pi(x) Q(x, x') = \sum_{x' \neq x} \pi(x') Q(x', x)$ holds).

The above description of the cleavage process is valid for any directed acyclic graph. Since we are concerned with exopeptidase activity modeling, we impose some restrictions. Let $\mathcal{V}_{\text{in}}$ be the set of nodes that have no parents. We set $a_{\star i}$ to 0 for $i \in \mathcal{V} \setminus \mathcal{V}_{\text{in}}$ and $a_{i\dagger}$ to 0 if node i has children. If i has no children then $\alpha_{i\dagger}$ is expressed as a sum of two elements from $\{a_r \mid r \in \mathcal{R}\}$, corresponding to the amino acids on both ends of subsequence i.

In order to go further with the description of the model, we need to change the parameterization a little bit. Write $b_{\star i} = s_1 a_{\star i}$ for $i \in \mathcal{V}$, where $s_1 = \sum_{i \in \mathcal{V}} a_{\star i}$ forcing $\sum_{i \in \mathcal{V}} b_{\star i} = 1$ and similarly $b_r = s_2 a_r$ for $r \in \mathcal{R}$, where $s_2 = \sum_{r \in \mathcal{R}} a_r$ forcing $\sum_{r \in \mathcal{R}} b_r = 1$. Now we can express λ_i as $s \mu_i$, $s = \frac{s_2}{s_1}$ for $i \in \mathcal{V}$ where μ_i depend only on $(b_r)_{r \in \mathcal{R}}$ and $(b_{\star k})_{k \in \mathcal{V}_{\text{in}}}$. We place a Gamma prior with parameters $(S_{\text{shape}}, S_{\text{rate}})$ on s and a Dirichlet prior with parameters $(B_r)_{r \in \mathcal{R}}$ on $(b_r)_{r \in \mathcal{R}}$ and $(B_{\star i})_{i \in \mathcal{V}_{\text{in}}}$ on $(b_{\star i})_{i \in \mathcal{V}_{\text{in}}}$. Since we are interested in relative intensities only, our goal is to estimate $(b_r)_{r \in \mathcal{R}}$, which we will call the *normalized cutting intensities.*

Model for data acquisition

Ideally, after the data preprocessing step one would get an exact reading on the numbers of particles corresponding to every possible subsequence present in the cleavage graph. In reality we must deal with many kinds of experimental errors.

First of all, many readings are missing. We can see which readings are missing and which are not. A vector of binary variables $(\epsilon_i)_{i \in \mathcal{V}}$ indicates the non-missing readings.

Some of the non-missing readings may be incorrect, meaning that they are taken from the wrong peaks from the LC-MS spectra, and have little to do with the peptides mentioned in the cleavage graph. This information is hidden and modeled by the δ variables coming from a Bernoulli process with success probability q.

Moreover, assuming that each correct reading is a sample from a Poisson distribution would imply low relative errors for readings from high peaks. This is clearly not realistic in case of the LC-MS data. Therefore, we assume that correct readings y_i for i such that $\delta_i = 1$ come from independent log-normal distributions with parameters $\ln x_i$ and τ (see Eqn. (4.1)), where x is the hidden realization of the cleavage process. Incorrect readings y_i for i such that $\delta_i = 0$ come independently from a background distribution with density bg. This density is estimated from the data (all mono-isotopic peak intensities in an LC-MS sample).

Note that from now on we define δ_i, x_i and y_i only for $i \in \mathcal{V}$ such that $\epsilon_i = 1$. When we write $i\colon \delta_i = 1$ we mean only those indices i, for which δ_i is defined. When we write x we mean $(x_i)_{i\colon \epsilon_i=1}$, etc.

Posterior distribution

The dependence structure of the variables in the hierarchical Bayesian model is shown in Fig. 4.3. The posterior distribution can be written as:

$$\begin{aligned} f(s, b_\star, b, \delta, x \mid y) &\propto f(y \mid s, b_\star, b, \delta, x) f(s, b_\star, b, \delta, x) \\ &= f(y \mid \delta, x) f(\delta) f(x \mid s, b_\star, b) f(s) f(b_\star) f(b), \end{aligned}$$

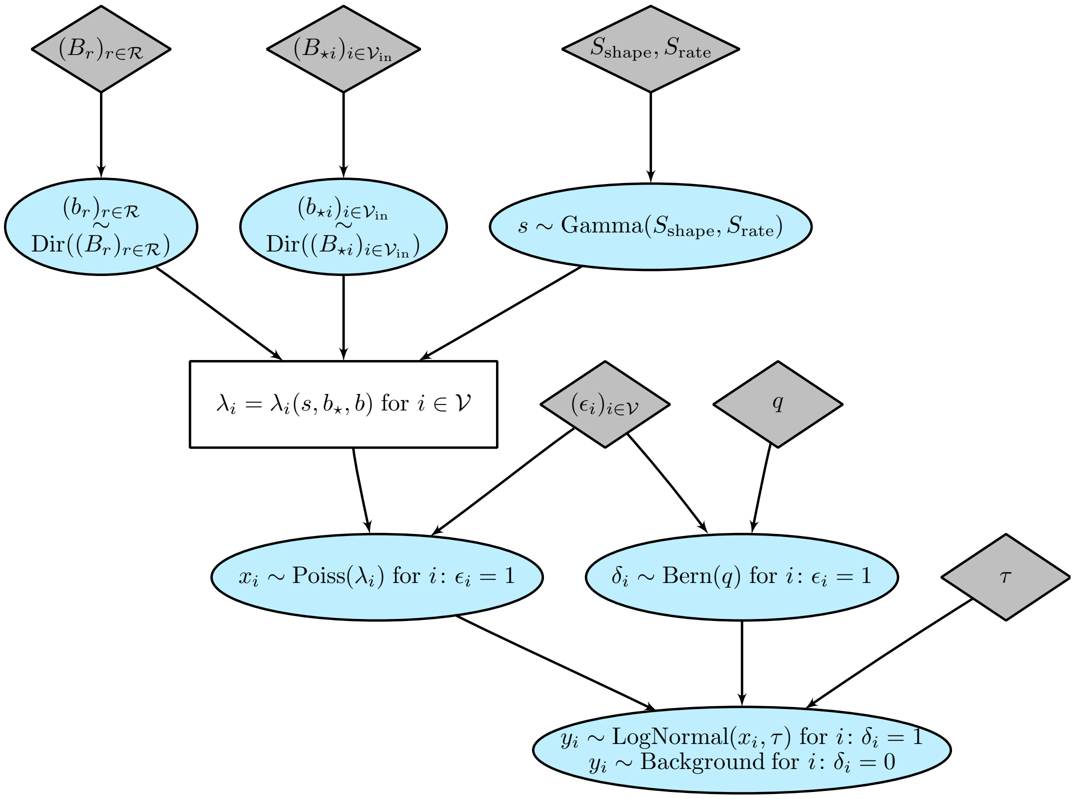

$(B_r)_{r\in\mathcal{R}}$
$(B_{\star i})_{i\in\mathcal{V}_{\text{in}}}$
$S_{\text{shape}}, S_{\text{rate}}$
$(b_r)_{r\in\mathcal{R}} \sim \text{Dir}((B_r)_{r\in\mathcal{R}})$
$(b_{\star i})_{i\in\mathcal{V}_{\text{in}}} \sim \text{Dir}((B_{\star i})_{i\in\mathcal{V}_{\text{in}}})$
$s \sim \text{Gamma}(S_{\text{shape}}, S_{\text{rate}})$
$\lambda_i = \lambda_i(s, b_\star, b)$ for $i \in \mathcal{V}$
$(\epsilon_i)_{i\in\mathcal{V}}$
q
$x_i \sim \text{Poiss}(\lambda_i)$ for $i\colon \epsilon_i = 1$
$\delta_i \sim \text{Bern}(q)$ for $i\colon \epsilon_i = 1$
τ
$y_i \sim \text{LogNormal}(x_i, \tau)$ for $i\colon \delta_i = 1$
$y_i \sim$ Background for $i\colon \delta_i = 0$

Figure 4.3: The hierarchical Bayesian model of cleavage activity and data acquisition.

where:

$$
\begin{aligned}
f(y \mid \delta, x) &= \prod_{i\colon \delta_i=0} \text{bg}(y_i) \prod_{i\colon \delta_i=1} \frac{1}{y_i \tau \sqrt{2\pi}} \mathrm{e}^{-\frac{(\ln y_i - \ln x_i)^2}{2\tau^2}}, \qquad (4.1)\\
f(\delta) &= q^{|\{i \mid \delta_i=1\}|}(1-q)^{|\{i \mid \delta_i=0\}|},\\
f(x \mid s, b_\star, b) &= s^{\sum_i x_i} \prod_i \frac{\mu_i^{x_i}}{x_i!} \mathrm{e}^{-s\mu_i},\\
f(s) &= s^{S_{\text{shape}}-1} \frac{S_{\text{rate}}^{S_{\text{shape}}} \mathrm{e}^{-S_{\text{rate}} s}}{\Gamma(S_{\text{shape}})},\\
f(b) &= \frac{\Gamma\left(\sum_{r\in\mathcal{R}} B_r\right)}{\prod_{r\in\mathcal{R}} \Gamma(B_r)} \prod_{r\in\mathcal{R}} b_r^{B_r-1},\\
f(b_\star) &= \frac{\Gamma\left(\sum_{i\in\mathcal{V}_{\text{in}}} B_{\star i}\right)}{\prod_{i\in\mathcal{V}_{\text{in}}} \Gamma(B_{\star i})} \prod_{i\in\mathcal{V}_{\text{in}}} b_{\star i}^{B_{\star i}-1}.
\end{aligned}
$$

Integrating out s yields:

$$
\begin{aligned}
f(x \mid b_\star, b) &= \int_0^\infty f(x \mid s, b_\star, b) f(s) \,\mathrm{d}s \\
&= \int_0^\infty s^{\sum_i x_i} \left(\prod_i \frac{\mu_i^{x_i}}{x_i!} \exp(-s\mu_i) \right) s^{S_{\text{shape}}-1} \frac{S_{\text{rate}}^{S_{\text{shape}}} \exp(-S_{\text{rate}} s)}{\Gamma(S_{\text{shape}})} \,\mathrm{d}s \\
&= \left(\prod_i \frac{\mu_i^{x_i}}{x_i!} \right) \frac{S_{\text{rate}}^{S_{\text{shape}}}}{\Gamma(S_{\text{shape}})} \frac{\Gamma\left(S_{\text{shape}} + \sum_i x_i\right)}{\left(S_{\text{rate}} + \sum_i \mu_i\right)^{S_{\text{shape}} + \sum_i x_i}} \\
&\qquad \int_0^\infty s^{S_{\text{shape}} + \sum_i x_i - 1} \frac{\left(S_{\text{rate}} + \sum_i \mu_i\right)^{S_{\text{shape}} + \sum_i x_i} \exp\left(-s\left(S_{\text{rate}} + \sum_i \mu_i\right)\right)}{\Gamma\left(S_{\text{shape}} + \sum_i x_i\right)} \,\mathrm{d}s \\
&= \left(\prod_i \frac{\mu_i^{x_i}}{x_i!} \right) \frac{S_{\text{rate}}^{S_{\text{shape}}}}{\Gamma(S_{\text{shape}})} \frac{\Gamma\left(S_{\text{shape}} + \sum_i x_i\right)}{\left(S_{\text{rate}} + \sum_i \mu_i\right)^{S_{\text{shape}} + \sum_i x_i}},
\end{aligned}
$$

since the expression under the last integral is the density of the gamma distribution. By summing out δ we obtain:

$$
\begin{aligned}
f(y \mid x) &= \sum_{\delta \in \{0,1\}^{|\{i \mid \epsilon_i = 1\}|}} f(y \mid \delta, x) f(\delta) \\
&= \sum_{\delta \in \{0,1\}^{|\{i \mid \epsilon_i = 1\}|}} \left(\prod_{i\colon \delta_i = 0} \mathrm{bg}(y_i) \right) \left(\prod_{i\colon \delta_i = 1} \frac{1}{y_i \tau \sqrt{2\pi}} \exp\left(-\frac{(\ln y_i - \ln x_i)^2}{2\tau^2} \right) \right) \\
&\qquad (1-q)^{|\{i \mid \delta_i = 0\}|} q^{|\{i \mid \delta_i = 1\}|} \\
&= \prod_{i\colon \epsilon_i = 1} \left((1-q)\mathrm{bg}(y_i) + q \frac{1}{y_i \tau \sqrt{2\pi}} \exp\left(-\frac{(\ln y_i - \ln x_i)^2}{2\tau^2} \right) \right).
\end{aligned}
$$

Finally we can write:

$$
f(b_\star, b, x \mid y) \propto f(y \mid x) f(x \mid b_\star, b) f(b) f(b_\star).
$$

4.1.2 Estimation procedure

We wish to estimate the $(b_r)_{r \in \mathcal{R}}$ parameters. The closed form of the expression $f(b_\star, b, x \mid y)$ was derived in the previous section. Since we are unable to integrate out $b_\star$ and x, we use the Metropolis–Hastings (Hastings, 1970) algorithm with the standard acceptance rule to sample $(b_\star, b, x)$ from the posterior distribution.

Transition proposal is generated by selecting with equal probability one of the three following rules:

1. changing $b_\star$:
   - generate $i, j \in \mathcal{V}_{\text{in}}$, $i \neq j$ uniformly,
   - generate $(b'_{\star i}, b'_{\star j}) \sim (b_{\star i} + b_{\star j})\text{Dir}(c\frac{b_{\star i}}{b_{\star i}+b_{\star j}} + 1, c\frac{b_{\star j}}{b_{\star i}+b_{\star j}} + 1)$, where c is a parameter of the procedure,
   - set $b'_{\star k}$ to $b_{\star k}$ for $k \notin \{i, j\}$,
   - propose transition $(b_\star, b, x) \mapsto (b'_\star, b, x)$,
2. changing b (analogously to changing $b_\star$),
3. changing x:
   - generate i such that $\epsilon_i = 1$ uniformly,
   - generate $x'_i \sim \text{LogNormal}(\ln x_i, d)$, where d is a parameter of the procedure,
   - set x'_k to x_k for $k \neq i$,
   - propose transition $(b_\star, b, x) \mapsto (b_\star, b, x')$.

4.1.3 Model testing

Compositional data analysis

The normalized cutting intensities lie on a simplex. The theory for analysis of such data (termed *compositional data analysis*) is summarized in (Aitchison and Egozcue, 2005). In short, it consists of interpreting the simplex as a Euclidean linear vector space and then applying standard analysis techniques. We use the following concepts:

- the *centered log ratio* transform of a point $(z_i)_{i=1,\dots,n}$ on a simplex is defined as:
  $$\text{clr}(z) = \left(\ln \frac{z_i}{\text{g}(z)}\right)_{i=1,\dots,n},$$
  where g denotes the geometric mean,

- the *Aitchison distance* is the Euclidean distance between clr-transformed points,
- the analogue of the expected value of a variable $(Z_i)_{i=1,\dots,n}$ on a simplex is the *Aitchison mean* defined as:

  $$\mathcal{C}\left((\exp \mathrm{E}[\ln Z_i])_{i=1,\dots,n}\right),$$

  where $\mathcal{C}$ denotes rescaling of the components so that their sum is 1,
- the analogue of principal component analysis is well defined (it amounts to performing PCA on clr-transformed data).

Testing on synthetic datasets

Three datasets (readings for the nodes of the cleavage graph) were generated according to the model with parameters $B_{\star i} = 2$ for $i \in \mathcal{V}_{\text{in}}$, $B_r = 2$ for $r \in \mathcal{R}$, $s = 10^6$, $\tau = 0.2$. Based on each of these datasets another dataset was derived by selecting nodes with correct readings with $q = 0.7$. Readings at all other nodes were resampled as being incorrect, each with λ parameter selected uniformly from $(\lambda_i)_{i\in\mathcal{V}}$. Thus we have two version of each of the three datasets – with and without incorrect readings.

The Metropolis–Hastings algorithm was run with parameters $c = 80$ (for changing $b_\star$ and b) and $d = 0.05$ (for changing x) for 3×10^6 iterations to recover the $(b_r)_{r\in\mathcal{R}}$ parameters.

Initial $b_\star$ and b were sampled from the Dirichlet priors. Initial x_i parameters were sampled from the log-normal distributions with parameters $\ln y_i, \tau$.

During the algorithm run the same Dirichlet priors and τ as during data generation were used. The S_{shape}, S_{rate} parameters were set to 0 and the q parameter was set appropriately to 1 or 0.7. On each dataset eight independent algorithm runs were conducted (therefore in total there were $3 \times 2 \times 8$ algorithm runs), each time with randomly selected 90% of the readings hidden as missing data. This was motivated by the fact that on real data only about 10% of the nodes of the cleavage graph had readings.

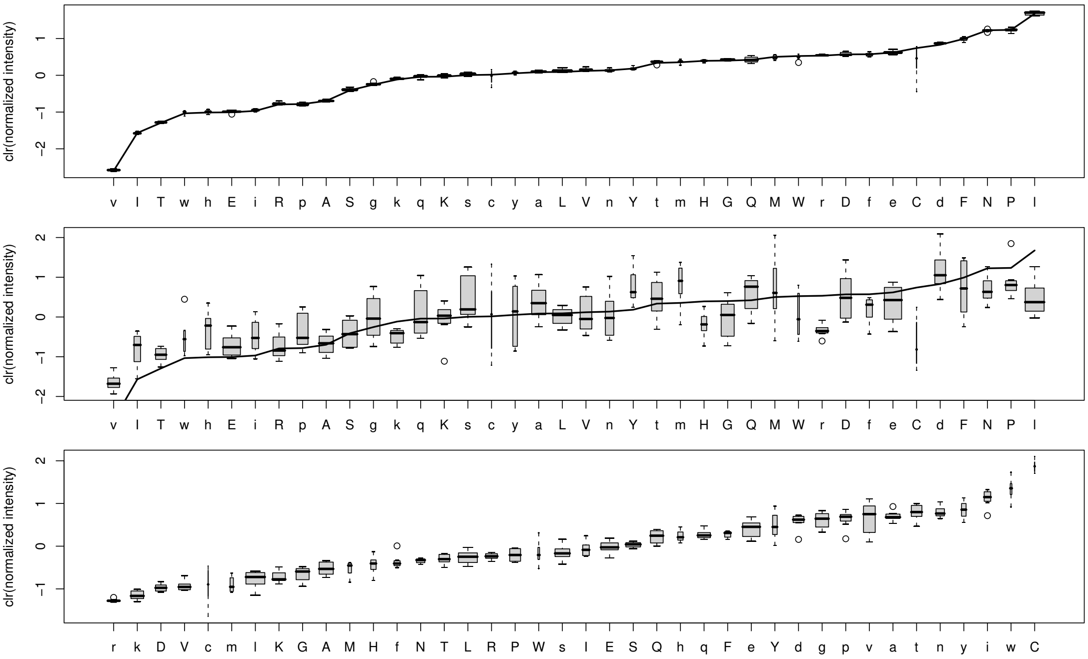


Figure 4.4: Clr-transformed normalized amino acid cutting intensities estimated from an artificial dataset without (top) and with (middle) incorrect readings and from a real dataset (bottom) based on 8 runs of the algorithm. The horizontal axis is sorted by the true intensities marked with the thick black line (top, middle) or by the Aitchison mean (bottom). Uppercase letters denote cutting from the N-terminus, while lowercase – from the C-terminus. The width of the bars is proportional to the number of appearances of the corresponding amino acid in the maximal vertices of the cleavage graph (i.e. the precursor peptides).

Results for one dataset with and without incorrect readings are presented in Fig. 4.4. Results for other datasets look similarly and are available as supplementary materials. Clearly the estimates for data without incorrect readings are very accurate. Since the estimates for data with incorrect readings are visually less appealing, we computed the Aitchison distance (see Section 4.1.3) between the averaged (over 8 runs using the Aitchison mean) estimated intensities and the true intensities. The results were 3.12 (dataset in Fig. 4.4), 2.64 and 3.23. To give those numbers some meaning, if we take independently two points from the Dirichlet prior with parameters $B_r = 2$ for $r \in \mathcal{R}$, then with probability greater

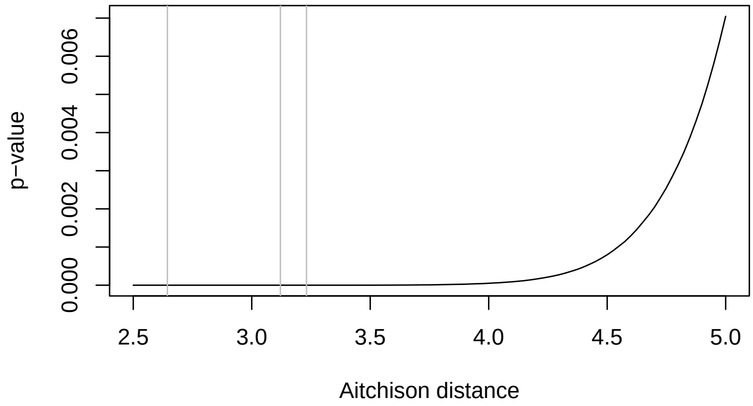


Figure 4.5: Cumulative distribution function of the Aitchison distance between two points taken independently from the Dirichlet prior with parameters $B_r = 2$ for $r \in \mathcal{R}$. Three vertical lines mark the distances between true and estimated intensities from synthetic datasets.

than 0.999 the Aitchison distance is greater than 4.5 (see Fig. 4.5).

4.1.4 Validation on LC-MS data

Colorectal cancer dataset

LC-MS and MS-MS data was provided by the Mass Spectrometry Laboratory of the Institute of Biochemistry and Biophysics of the Polish Academy of Sciences. The mass spectrometer used in the experiments was an ElectroSpray Ionization Fourier Transform Ion Cyclotron Resonance (ESI-FTICR) coupled with an HPLC retention column.

The dataset comprised mass spectra acquired from serum samples for colorectal cancer patients. Apart from the patient data, control samples were also collected from healthy donors and analyzed with the mass spectrometer. The colorectal cancer dataset consisted of 29 spectra, 15 samples corresponding to diseased patients and 14 to healthy donors.

Cleavage graph construction

For the construction of the cleavage graph we start with the information about successfully sequenced peptides from a LC-MS/MS experiment. This information covers a little over 1000 peptides and is comprised of peptide mass, mass-to-charge ratio, retention time, charge, amino acid sequence, protein of origin, and optionally the information about the oxidation for each peptide. For simplicity, the oxidated molecules are omitted. The set of vertices of the cleavage graph is defined using all other maximal sequences (i.e. the precursor peptides; they can be found in the supplementary materials) and their subsequences. It has 39544 elements, including 243 maximal vertices. This graph is fixed in all our experiments (both on artificial and real data).

Data preprocessing

For each spectrum we used the *mz2m* program (see Section 2.3) to obtain a list of mono-isotopic peak coordinates (m/z values and retention times) together with their charges and intensities.

For each cleavage graph vertex encoded by amino acid sequence we need to find a corresponding signal in the spectrum. One can easily compute mass of the sequence and consider mass-to-charge ratios for charge $z \in \{1, \ldots, 8\}$ (greater charges do not occur in the data). There is a problem however with the retention time. As we mentioned, the retention time is readily available for some sequences. We use this information to train the Random Forest regression algorithm (Breiman, 2001) to predict the retention time from the amino acid composition.

The lists of mono-isotopic peaks from spectra together with the list describing the nodes of the graph were aligned by applying to each list a linear transformation along the retention time axis as described in Section 3.1.

Assuming we know the retention time and several possible mass-to-charge ratios for a given sequence, we find peaks nearest to those locations on the LC-MS spectrum. The Euclidean metric is used with the retention time scaled by 10^{-2}. Signals which are further than 0.05 or with charge mismatch are discarded.

Intensities of the rest are summed and returned as the observed value at a suitable node of the cleavage graph.

We are aware that there are many factors influencing the intensity measurements (Mallick et al., 2007) (for instance isoelectric point). We leave integrating this knowledge into the model for the future, especially as we already have a component that accounts for inexact readings (generating y from x, cf. Fig. 4.3).

Testing on real dataset

In order to illustrate the applicability of the model to real data we analyzed the colorectal cancer dataset. We show that our model can be used to discriminate between diseased patients and healthy donors. In each of the 29 samples about 90% readings in the nodes of the cleavage graph were missing. Since there were only about 0.7% nodes with readings from all samples, it is not straightforward to bypass the model parameters estimation and perform classification directly on the data (one would have to deal somehow with the missing data). Therefore we leave comparison with other classification methods as a topic for further research, but we stress that our model can provide insights into the peptide degradation process.

The estimation algorithm was run 8 times on each sample with the q parameter set to 0.7 and all other parameters as described in the previous section. Figure 4.4 shows that the results are quite consistent (additional figures can be found in the supplementary materials). Obtained intensities were averaged over these 8 runs using the Aitchison mean.

Figure 4.6 shows the data projected on the first three principal components (accounting for almost 75% of total variance). The first and third components are heavily influenced by Cysteine cutting intensity (see loadings on Fig. 4.6) and do not discriminate samples well. The second component is the only one significant in this respect (Bonferroni corrected p-value from Kolmogorov-Smirnov test below 0.01). We tried to confirm whether it carries the information about the altered pattern of exopeptidase activity.

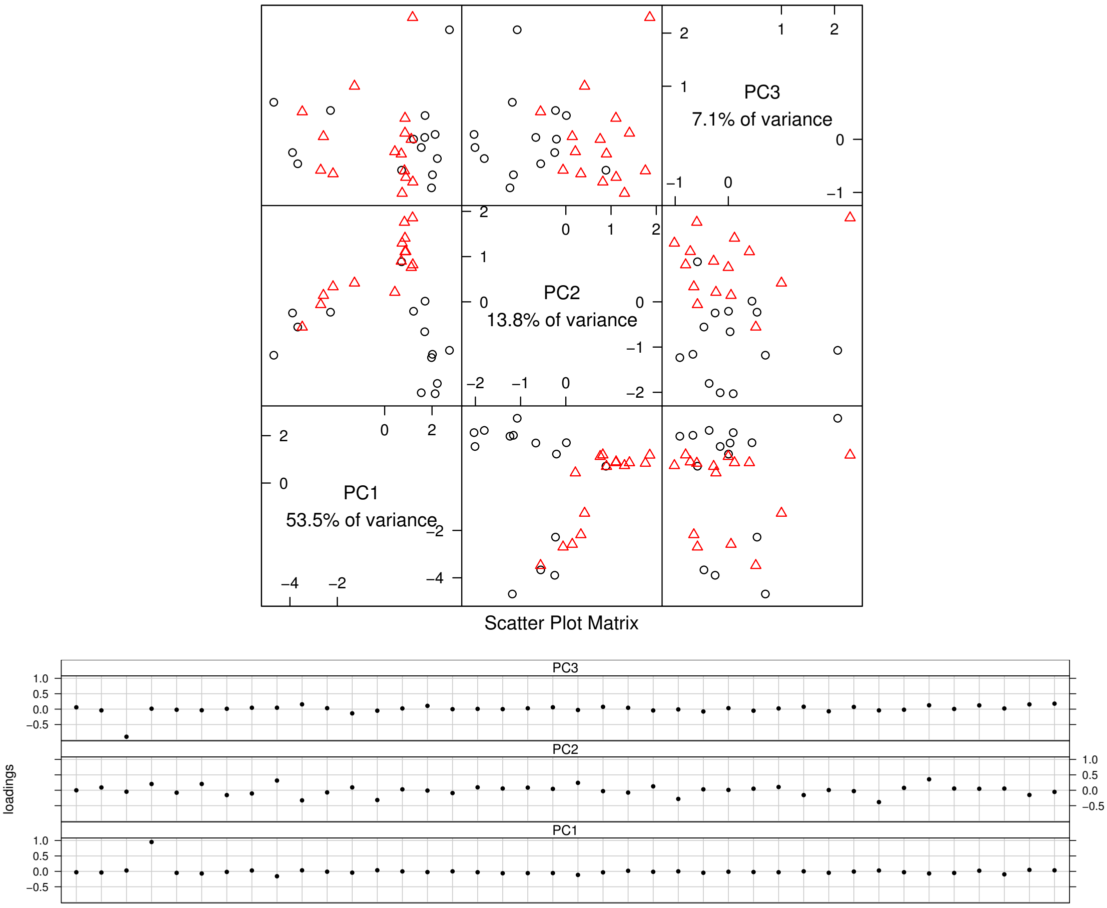


Figure 4.6: Scatter plot of the colorectal cancer dataset for the first three principal components and the corresponding loadings. Healthy donors are marked with black circles. Diseased patients are marked with red triangles. Uppercase letters denote cutting from the N-terminus, while lowercase – from the C-terminus.

To determine the hypothetical enzymes involved, we scanned MEROPS database (Rawlings and Barrett, 2000) for peptidases cleaving the specific bond (e.g. Threonine, Valine and Histidine from C-terminus, Phenylalanine and Aspartic acid from N-terminus, see Fig. 4.6). Resulting list contains many metallopeptidases, which are experimentally verified as crucial for colorectal cancer development (Leeman et al., 2003; Masaki et al., 2001).

We also investigated whether patient classification based on clr-transformed normalized cutting intensities can be performed. Using the SVM classifier (Schölkopf and Smola, 2002) with linear kernel, the .632+ bootstrap (Efron and Tibshirani, 1997) error estimate based on 1000 bootstrap replicates was 12.4%. We repeated the whole procedure 1000 times with class labels permuted randomly. The average .632+ bootstrap error estimate was 54.5% with standard deviation 12.5%, suggesting that cutting intensities indeed contain information about patient state (and perhaps that the .632+ estimator is too pessimistic).

4.1.5 Conclusions

Up to our knowledge, this work represents the first attempt to model the protein degradation process from LC-MS data. We described a mathematical framework allowing for adequate statistical modeling. The model was extensively tested on suitably chosen artificial datasets, as well as on real LC-MS samples.

The outcome of computational experiments is very promising. The estimation procedure yielded robust results even when dealing with errors and missing values in the input data. Moreover, the accurate classification results for colorectal cancer patients suggest diagnostic potential of the model.

On the other hand, we are aware of the problems with reproducibility of the LC-MS experiments. Two more datasets were at our disposal. They were collected at different times and processed with different HPLC columns. After preliminary analyzes we decided however to base the presentation of our model only on one dataset, because the results were hard to compare between different datasets. We believe that better LC-MS spectra alignment procedures would decrease the variability between the datasets.

Recently, new diagnostic test has been proposed to compare proteolytic activities within individual proteome of two groups of biological samples (Villanueva et al., 2008). It tracks degradation of artificial substrates under strictly controlled conditions. We plan to adopt our model to this setting and infer the hypotheses on the activity of postulated but as yet unidentified exopeptidases.

Finally, some concerns may be raised regarding the cleavage process stationarity assumption, especially as it is hard to strictly control the time between sample collection and MS analysis. In Section 4.2 we will modify our model to remove this assumption.

4.2 Time dependent model for peptidase activity

We develop a comprehensive mathematical model describing the activity of peptide cutting enzymes (peptidases). Our model enables parameter inference from mass spectrometry data. The dynamics of peptide degradation is described by means of biochemical reactions network. It is widely accepted that stochasticity is an inherent property of such systems, therefore we model the network of proteolytic reactions as a Markov process, whose evolution is described by the chemical master equation (CME).

In Section 4.1 we have proposed a mathematical model for exopeptidase activity by means of the CME. The general idea was similar to the one described here — to track the numbers of peptide particles as they are being cut into smaller sequences. To keep things simple we postulated a constant flow of long peptide sequences into the system and looked only at the stationary state of the cleavage process. We restricted ourselves to enzymes operating near the ends of peptides (exopeptidases) which allowed us to obtain an analytic solution. In particular, we characterized the stationary solution of the CME as a product of Poisson distributions. The model was tested in simulations and gives good predictions on colorectal cancer dataset, but it suffers from two significant limitations. Firstly, it considers only peptidases cutting one amino acids from the end of the peptide. Secondly, it assumes stationarity of the proteolysis process which does not make sense when one considers time series data.

In this section we address these two problems. Firstly, we integrate our model with peptidase database MEROPS (Rawlings and Barrett, 2000) and model endopeptidase activity as well. Secondly, we model proteolytic activity in time by studying the evolution of the underlying stochastic process before reaching equilibrium (we no longer need to assume a constant flow of long peptide sequences into the system).

Although allowing the endopeptidase cuts (i.e. splitting reaction of the form $A \rightarrow B + C$) complicates the model, we managed to characterize the time-evolution of peptide population (i.e. expected configuration of a Markov process). We derive the system of differential equations which describes the dynamics of means from the CME. The solution of the system is obtained by matrix exponentiation. For this problem we propose an original combinatorial approach which is interesting for its own sake and can be applied to a wide class of biochemical reactions systems.

The model parameters corresponding to the activity of specific enzymes are fitted to minimize the discrepancy between the expected amount of peptides calculated from the model and the readouts from mass spectra. The outcome of computational experiments performed on simulated datasets is very promising. The estimation is efficient even in the presence of erroneous or missing readouts and the model is capable of inferring the enzyme concentration levels.

In Section 4.1 we used a Markov Chain Monte Carlo method to estimate enzyme cutting intensities. In particular, the Metropolis-Hastings algorithm was applied to sample parameters from the posterior distribution. In this approach we decided to apply a generic quasi-Newton method for parameter estimation in order to avoid writing a custom sampler and reduce the complexity of error modeling.

Our main results can be summarized as follows:

- proposing a rigorous model for proteolytic activity inferred from mass spectrometry data,
- giving an explicit representations of means for proteolytic reactions network,
- describing a new matrix exponentiation algorithm,
- suggesting a method for the estimation of model parameters.

In Section 4.2.1 we introduce the mathematical model of serum proteolysis process and present a method to calculate expected values of peptide amounts in time. Section 4.2.2 deals with matrix exponentiation. The approach to model parameters estimation is described in Section 4.2.3. The method for incorporating the biological information about proteolytic events into the model is presented in Section 4.2.4. Finally, Section 4.2.5 contains results of experiments and discussion of further research.

4.2.1 Cleavage process

Peptide sequences whose proteolysis we wish to model give rise to a bipartite multidigraph, which we call the *cleavage graph*. The first set of nodes of this graph corresponds to all subsequences of the peptides considered. We call them *peptide nodes*, and denote by $\mathcal{V}$. The second set of nodes, called *event nodes*, corresponds to all possible proteolytic events. By proteolytic event we mean the cleavage of a specific substrate at a specific site made by a specific peptidase. Hence each event node is labelled by a peptidase, and has one ingoing edge (leading from the substrate of proteolysis) and two outgoing edges (leading to peptide prefix and suffix obtained by cutting the substrate at a single site).

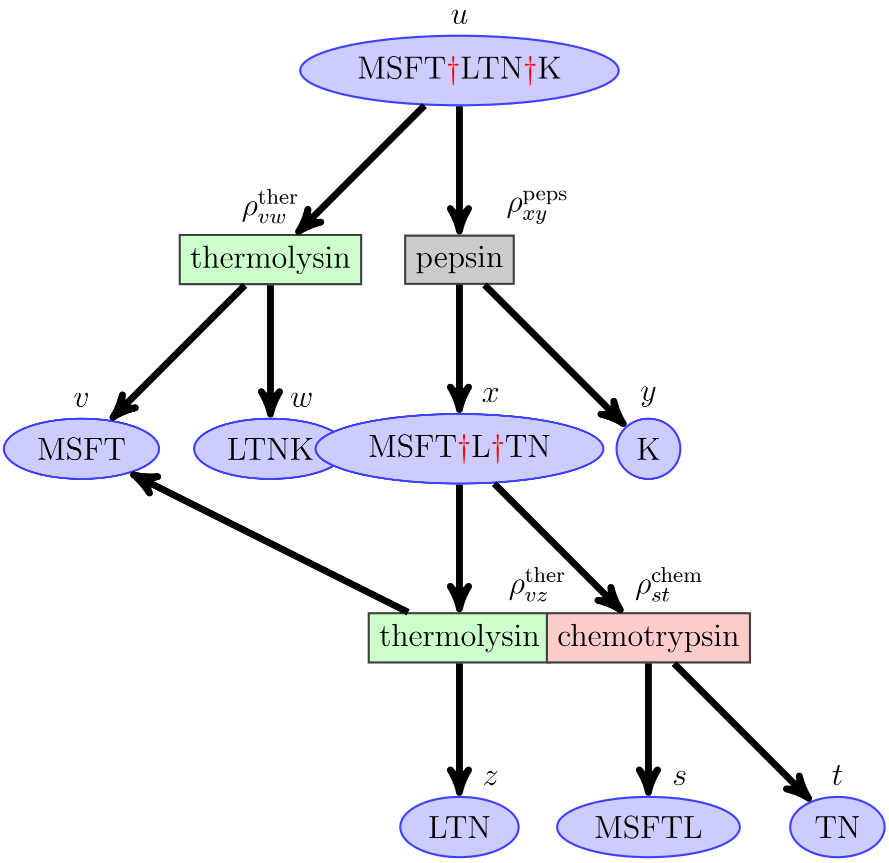

u
MSFT†LTN†K
ρ_{vw}^{ther}
ρ_{xy}^{peps}
thermolysin
pepsin
v
w
x
y
MSFT
LTNK
MSFT†L†TN
K
ρ_{vz}^{ther}
ρ_{st}^{chem}
thermolysin
chemotrypsin
z
s
t
LTN
MSFTL
TN

Figure 4.7: The cleavage graph for the precursor peptide MSFTLTNK (*alcohol dehydrogenase* fragment). Proteolytic events are taken from the MEROPS database (Rawlings and Barrett, 2000).

It is helpful to think of the peptide subsequences as particles placed at peptide nodes of the cleavage graph. The particles are moving along the edges of the graph according to the Petri net operational semantics, i.e. the transition (event node) consumes one substrate particle, and produces two particles (prefix and suffix of the substrate). As every transition needs exactly one token to occur, our system belongs to the subclass of *communication-free nets* (Esparza, 1997).

We assume that the labeling and structure of the cleavage graph is known. In Section 4.2.4 we show how to assign specific enzymes to event nodes. An exemplary cleavage graph is presented in Figure 4.7.

In the exemplary graph four proteolytic events which engage three peptidases are depicted. For $u, v, w \in \mathcal{V}$ we use the notation $u = v \dagger w$ when peptides v and w can be obtained directly by cutting u (v is a non-empty strict prefix and w is a non-empty strict suffix of u). The operation $\dagger$ can be viewed as string concatenation. To identify a cleavage site we write simply $v \dagger w$.

Denote by $\mathcal{P}$ the set of all peptidases whose activity is modeled. Coefficients ρ^p_{vw} (for peptidase $p \in \mathcal{P}$ and cleavage $v \dagger w$) put over the event nodes in Figure 4.7 correspond to the affinity between the peptidase cleavage pattern and the cleavage site (we call them *affinity coefficients*). They are defined for every possible cleavage $v \dagger w$ and calculated at the graph construction stage (see Section 4.2.4 for details).

Our ultimate goal is to estimate peptidase cutting intensities vector $c = (c_p)_{p \in \mathcal{P}}$. We assume that the propensity of a given peptidase p to perform the cleavage $v \dagger w$ is proportional to its overall intensity c_p weighted with the appropriate affinity coefficient (i. e. ρ^p_{vw}). The cleavage intensity for a given site $v \dagger w$ is a weighted sum of intensities of all peptidases considered in our model.

To define the probabilistic dynamics of the cleavage process more formally, let random variables $X_i(t)$ denote the number of particles at peptide node $i \in \mathcal{V}$ at time t and write $X(t) = (X_i(t))_{i \in \mathcal{V}}$. We regard $(X(t), t \geq 0)$ as a homogeneous Markov process in the space of configurations $x = (x_i)_{i \in \mathcal{V}}$, $x_i \in \{0, 1, \dots\}$. Denote by ρ_{vw} the vector of all peptidase affinity coefficients for the cleavage $v \dagger w$ (for convenience if $v \dagger w \notin \mathcal{V}$ then define $\rho_{vw} = 0$). Define ϵ_z for $z \in \mathcal{V}$ as a vector of dimension $|\mathcal{V}|$ with only one non-zero coordinate corresponding to the vertex z.

The process has the following intensities of transition from state x to state x' where $x \neq x'$:

$$Q_{xx'} = \begin{cases} c^{\mathrm{T}} \rho_{vw} x_u & \text{if } x' = x - \epsilon_u + \epsilon_v + \epsilon_w \text{ and } u = v \dagger w \ , \\ 0 & \text{otherwise.} \end{cases}$$

We are interested in the distribution of this process at finite time points.

The model of exopeptidase activity presented in Section 4.1 allowed for the full characterization of the underlying Markov process. In the present setting the splitting reactions corresponding to endopeptidase proteolytic events make the system behavior more complex. Especially, there are no analytic results for Markov processes modeling such systems. In our approach we focus on time evolution of the expected numbers of particles.

Consider the probability distribution characterizing the time evolution of a Markov process $(X(t))_{t>0}$:

$$P(x,t) = \mathcal{P}(X(t) = x).$$

The distribution P is the solution of the chemical master equation:

$$\begin{aligned} \frac{\partial}{\partial t} P(x,t) &= \sum_{y \neq x} \left(Q_{yx} P(y,t) - Q_{xy} P(x,t) \right) \\ &= \sum_{u = v \dagger w} c^{\mathrm{T}} \rho_{vw} \left[(x_u + 1) P(x + \epsilon_u - \epsilon_v - \epsilon_w, t) - x_u P(x,t) \right] \\ &= \sum_{u = v \dagger w} c^{\mathrm{T}} \rho_{vw} [x'_u P(x',t) - x_u P(x,t)], \end{aligned}$$

where $x' = x + \epsilon_u - \epsilon_v - \epsilon_w$, i.e. x' denotes a configuration before the cleavage $v \dagger w$.

Denote by $\mathrm{E}_q\,(t)$ the expected number of instances of peptide q at time t. We have from the chemical master equation above:

$$
\begin{aligned}
\mathrm{E}_q(t) &= \sum_x x_q P(x,t), \\
\frac{d}{dt}\mathrm{E}_q(t) &= \sum_x x_q \frac{\partial}{\partial t} P(x,t) \\
&= \sum_x x_q \sum_{u=v\dagger w} c^{\mathrm{T}}\rho_{vw}\left[(x_u+1)P(x+\epsilon_u-\epsilon_v-\epsilon_w,t) - x_u P(x,t)\right] \\
&= \sum_{u=v\dagger w} c^{\mathrm{T}}\rho_{vw}\left[\sum_x x_q(x_u+1)P(x+\epsilon_u-\epsilon_v-\epsilon_w,t) - \sum_x x_q x_u P(x,t)\right] \\
&= \sum_{u=v\dagger w} c^{\mathrm{T}}\rho_{vw}\left[\sum_x (x-\epsilon_u+\epsilon_v+\epsilon_w)_q x_u P(x,t) - \sum_x x_q x_u P(x,t)\right] \\
&= \sum_{u=v\dagger w} c^{\mathrm{T}}\rho_{vw}\sum_x (-\epsilon_u+\epsilon_v+\epsilon_w)_q x_u P(x,t).
\end{aligned}
$$

Now observe that for all $q \notin \{u, v, w\}$ the summands are zero (as $(-\epsilon_u+\epsilon_v+\epsilon_w)_q = 0$) and consider three cases: $v = q$, $w = q$ and $u = q$ (the first two may overlap if $v = w = q$). The following holds:

$$
\frac{d}{t}\mathrm{E}_q(t) = \sum_{u=q\dagger w} c^{\mathrm{T}}\rho_{qw}\,\mathrm{E}_u(t) + \sum_{u=v\dagger q} c^{\mathrm{T}}\rho_{vq}\,\mathrm{E}_u(t) - \sum_{q=v\dagger w} c^{\mathrm{T}}\rho_{vw}\,\mathrm{E}_q(t). \tag{4.2}
$$

The first two summands correspond to the creation of particle q from u by performing two kinds of cleavages: $q \dagger w$ or $v \dagger q$, i.e. the word q can form a suffix or a prefix of u. The third summand corresponds to the consumption of the particle q. It happens when q is cleaved at some site.

We introduce the notation $q \to v$ if v can be directly obtained by cutting q, i.e. v is a non-empty strict prefix or a non-empty strict suffix of q. Note that $q \to v$ means that there exists the cleavage site $q = v \dagger w$ or $q = z \dagger v$ or both.

Denote by λ_{uq} the intensity of creating q from u by a single cleavage of the form $u = q \dagger w$ or $u = v \dagger q$, i.e. $\lambda_{uq} = c^{\mathrm{T}}(\rho_{qw} + \rho_{vq})$. Let $\lambda_{qq} = -\sum_{q=v\dagger w} c^{\mathrm{T}}\rho_{vw}$, i.e. minus the intensity of consuming q in all cleavages involving this peptide. Note that the following equality holds:

$$
\lambda_{qq} = -\frac{1}{2}\left[\sum_{q=v\dagger w}\lambda_{qv} + \sum_{q=z\dagger v}\lambda_{qv} - \sum_{\substack{q=v\dagger w\\ q=z\dagger v}}\lambda_{qv}\right] = -\frac{1}{2}\sum_{q\to v}\lambda_{qv}.
$$

Now the equations (4.2) have the following form:

$$\left[\frac{d}{dt}\,\mathrm{E}_q\left(t\right)=\sum_{u\to q}\lambda_{uq}\,\mathrm{E}_u\left(t\right)+\lambda_{qq}\,\mathrm{E}_q\left(t\right)\right]_{q\in\mathcal{V}}. \tag{4.3}$$

The solution of the system of linear constant coefficient ordinary differential equations like (4.3) is given by:

$$\mathrm{E}\left(t\right)=\mathrm{E}\left(0\right)^{\mathrm{T}}\exp(\Lambda t), \tag{4.4}$$

where $\mathrm{E}\left(t\right)=\left(\mathrm{E}_v\left(t\right)\right)_{v\in\mathcal{V}}$, $\mathrm{E}\left(0\right)=\left(\mathrm{E}_v\left(0\right)\right)_{v\in\mathcal{V}}$ and matrix $\Lambda=\left(\lambda_{vw}\right)_{v,w\in\mathcal{V}}$. The matrix exponentiation in Equation (4.4) can be computed by dozens of methods (Moler and Loan, 2003) that originate from mathematical analysis, matrix theory or approximation theory. Here we propose a new method, which can be a tempting alternative to existing ones[1], as long as the coefficient matrix is triangular.

4.2.2 Matrix exponentiation

Define relation $<$ on all subsequences as a transitive closure of the relation $\to$, i.e. $v<u \iff v$ is a non-empty substring of u. We also write $v\le u$ when $v<u$ or $v=u$.

Notice that the system (4.3) can be solved by processing equations in the topological order (in terms of the $\le$ partial order, starting from the maximal elements). One can postulate (or check) that the solution can be written as:

$$\mathrm{E}_u\left(t\right)=\sum_{v\ge u}b_{uv}\exp(\lambda_{vv}t),\quad\text{where}\quad b_{uv}=\sum_{w\ge v}a_{uvw}\,\mathrm{E}_w\left(0\right). \tag{4.5}$$

This way $\mathrm{E}_u\left(t\right)=\sum_{w\ge v\ge u}a_{uvw}\exp(\lambda_{vv}t)\,\mathrm{E}_w\left(0\right)$. Define $a_{uvw}=0$ when $w\ge v\ge u$ does not hold. Now we can write those equations in vectorized form:

$$\mathrm{E}_u\left(t\right)=\mathrm{E}\left(0\right)^{\mathrm{T}}\sum_{v\ge u}a_{uv}\exp(\lambda_{vv}t),$$

[1] In our implementation a function from one of the R software package (Team, 2009) libraries is used.

where $a_{uv} = (a_{uvw})_{w\in\mathcal{V}}$. Notice that $\sum_{v\geq u} a_{uv} \exp(\lambda_{vv}t)$ corresponds to the u-th column of the matrix $\exp(\Lambda t)$.

The a_{uvw} for $w \geq v \geq u$ coefficients are real numbers that can be calculated from the system of equations (4.3):

$$\begin{aligned}
\frac{d}{dt}\,\mathrm{E}_u\,(t) &= \lambda_{uu}\,\mathrm{E}_u\,(t) + \sum_{w\to u} \lambda_{wu}\,\mathrm{E}_w\,(t) \\
&= \lambda_{uu}\,\mathrm{E}_u\,(t) + \sum_{w\to u} \lambda_{wu} \sum_{v\geq w} b_{wv} \exp(\lambda_{vv}t) \\
&= \lambda_{uu}\,\mathrm{E}_u\,(t) + \sum_{v\geq w\to u} \lambda_{wu} b_{wv} \exp(\lambda_{vv}t).
\end{aligned}$$

Solving the differential equation yields:

$$\begin{aligned}
\mathrm{E}_u\,(t) = &\left[\mathrm{E}_u\,(0) - \sum_{v>u} \frac{1}{\lambda_{vv}-\lambda_{uu}} \sum_{v\geq w\to u} \lambda_{wu} b_{wv}\right] \exp(\lambda_{uu}t) \\
&+ \sum_{v>u} \frac{1}{\lambda_{vv}-\lambda_{uu}} \sum_{v\geq w\to u} \lambda_{wu} b_{wv} \exp(\lambda_{vv}t),
\end{aligned}$$

which gives a recursive formula for the b coefficients (see Equation (4.5)):

$$\begin{aligned}
b_{uv} &= \frac{1}{\lambda_{vv}-\lambda_{uu}} \sum_{v\geq w\to u} \lambda_{wu} b_{wv} \quad \text{for } v > u, \\
b_{uu} &= \left[\mathrm{E}_u\,(0) - \sum_{v>u} b_{uv}\right],
\end{aligned}$$

and, in consequence, for the a coefficients (again see Equation (4.5)):

$$a_{uvw} = \frac{1}{\lambda_{vv}-\lambda_{uu}} \sum_{v\geq z\to u} \lambda_{zu} a_{zvw} \quad \text{for } w > v > u, \tag{4.6}$$

$$a_{uuw} = -\sum_{w\geq v>u} a_{uvw} \quad \text{for } w > u, \tag{4.7}$$

$$a_{uuu} = 1. \tag{4.8}$$

The recursive equations (4.6), (4.7) and (4.8) can be translated into a dynamic programming algorithm. They let us solve the system of differential equations (4.3) and simultaneously define the $\exp(\Lambda t)$ matrix. Since they exploit the structure of the $\leq$ partial order, they might be preferred to the more general methods. Good implementation would use efficient data structures for partial order traversal to compute the sums above.

4.2.3 Estimation procedure

In this section we describe our approach to estimation of the peptidase cutting intensities (i.e. the vector $c = (c_p)_{p \in \mathcal{P}}$, where $\mathcal{P}$ is the set of peptidases considered in the model). Assume that we have a series of mass spectra analyzed at time points $t_1, \ldots, t_k$ at our disposal. For every sequence $v \in \mathcal{V}$ recall that $x_v(t_i)$ denotes the amount of the v peptide observed in mass spectra at time point t_i. Denote by $\mathcal{V}_i^{\text{obs}}$ the set of sequences observed in the spectra at time point t_i.

From equation (4.4) we calculate $\mathrm{E}_v(t)$ as a function of parameters c and $\mathrm{E}(0) = (\mathrm{E}_v(0))_{v \in \mathcal{V}}$.

Our goal is to find values c^* and $\mathrm{E}(0)^*$ that minimize the discrepancy between the expected amount of peptides calculated in the model and the amount of peptides observed in the mass spectra, i.e.

$$(c^*, \mathrm{E}(0)^*) = \arg\min_{c, \mathrm{E}(0)} \Phi(c, \mathrm{E}(0))$$

with constraints:

$$\begin{array}{l} \mathrm{E}_v(0) > 0 \quad \text{for all} \quad v \in \mathcal{V}, \\ c_p > 0 \quad \text{for all} \quad p \in \mathcal{P}, \end{array}$$

where the *objective function* $\Phi(c, \mathrm{E}(0))$ is defined as follows:

$$\Phi(c, \mathrm{E}(0)) = \sum_i \sum_{v \in \mathcal{V}_i^{\text{obs}}} \left[\mathrm{E}_v(t_i) - x_v(t_i)\right]^2.$$

We solve the above constrained minimization problem using the BFGS method implemented in R – a state-of-the-art, freely available statistical software package (Team, 2009). BFGS is a popular quasi-Newton method named for its discoverers Broyden, Fletcher, Goldfarb and Shanno (Nocedal and Wright, 1999) To handle the inequality constraints we use the limited-memory modification of the BFGS proposed in (Lu et al., 1994).

4.2.4 MEROPS – a peptide cleavage database

To assign the appropriate affinity coefficients ρ_{vw} to all event nodes $v \dagger w$ of the cleavage graph we have to fix the set of peptidases included in the model (denoted by $\mathcal{P}$). This can be e.g. the set of all human proteolytic enzymes stored in databases like MEROPS (Igarashi et al., 2007; Rawlings and Barrett, 2000) or some smaller set of enzymes when we have some knowledge about the digestion of investigated peptides mixture.

Consider single event node $v \dagger w$. For each peptidase $p \in \mathcal{P}$ we have to determine the affinity coefficient ρ^{p}_{vw}. To this aim we look in the MEROPS database for the knowledge about all reported proteolytic events for this peptidase. Every proteolytic event stored in the MEROPS database is characterized by the following information: the name of the peptidase, the name of the substrate cleaved and the amino acid composition of the neighborhood of the cleavage site. It is widely assumed that the propensity of a given peptidase to cleave in the given locus depends only on four amino acids to the left and four amino acids to the right of the cleavage site.

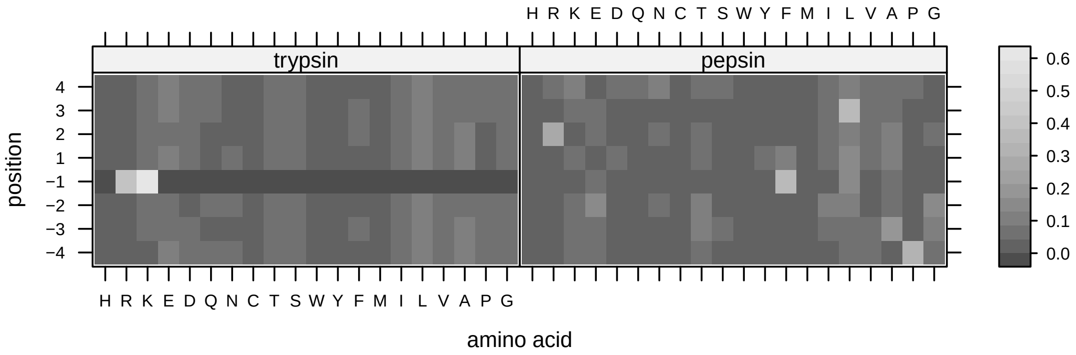


Figure 4.8: Graphical representation of the transposed affinity matrices for trypsin and pepsin. Each row corresponds to a position relative to the cleavage site and shows the probabilities of seeing each amino acid (estimated from the MEROPS database).

The information from the MEROPS database about the proteolytic events involving p is summarized in a 20×8 matrix. We call this matrix an *affinity matrix* and denote it by F_p. The rows of F_p correspond to amino acids (letters from the alphabet Σ) and columns to the positions in the cleaved pattern (see Figure 4.8).

We define $F_p(i, j)$ as the fraction[2] of events such that amino acid $i \in \Sigma$ is at position $j \in \{-4, -3, -2, -1, 1, 2, 3, 4\}$ relative to the cleavage site.

Some cleavages are performed near the end of the peptide sequence. Denote the fraction of database events having exactly $k \in \{1, 2, 3, 4\}$ amino acids to the left of the cleavage site as ϕ_{pk}^{left} (the part to the right of the cleavage site may be arbitrary), i.e. let "$*$" denote any amino acid and "$-$" the lack of amino acids in a cleavage pattern and let:

- ϕ_{p1}^{left} be the fraction of $- - - * \dagger$ sites,
- ϕ_{p2}^{left} be the fraction of $- - * * \dagger$ sites,
- ϕ_{p3}^{left} be the fraction of $- * * * \dagger$ sites,
- ϕ_{p4}^{left} be the fraction of $* * * * \dagger$ sites.

Analogously define the ϕ_p^{right} vector.

To calculate the coefficient ρ_{vw}^p for peptidase p and cleavage $v \dagger w$ let $k_{\text{left}} = \min\{|v|, 4\}$, where $|v|$ is the length of v, and analogously $k_{\text{right}} = \min\{|w|, 4\}$. Let v be a string of amino acids $v[k_{\text{left}}], \ldots, v[1]$ and w a string of amino acids $w[1], \ldots, w[k_{\text{right}}]$. We estimate ρ_{vw}^p as:

$$\rho_{vw}^p = \phi_{pk_{\text{left}}}^{\text{left}} \prod_{j=1}^{k_{\text{left}}} F_p(v[j], -j) \prod_{j=1}^{k_{\text{right}}} F_p(w[j], j)\, \phi_{pk_{\text{right}}}^{\text{right}}.$$

Using the formula above we make a simplifying assumption that amino acids in the successive positions of cleavage pattern are independent of each other. This is along the lines of commonly used representation of patterns in biological sequences called PSSM (position-specific scoring matrix).

[2] It is quite probable that the set of proteolytic events used to built affinity matrix F does not include a particular example actually existent in nature, hence it is usual to add a *pseudocount*, corresponding to a uniform Bayesian prior.

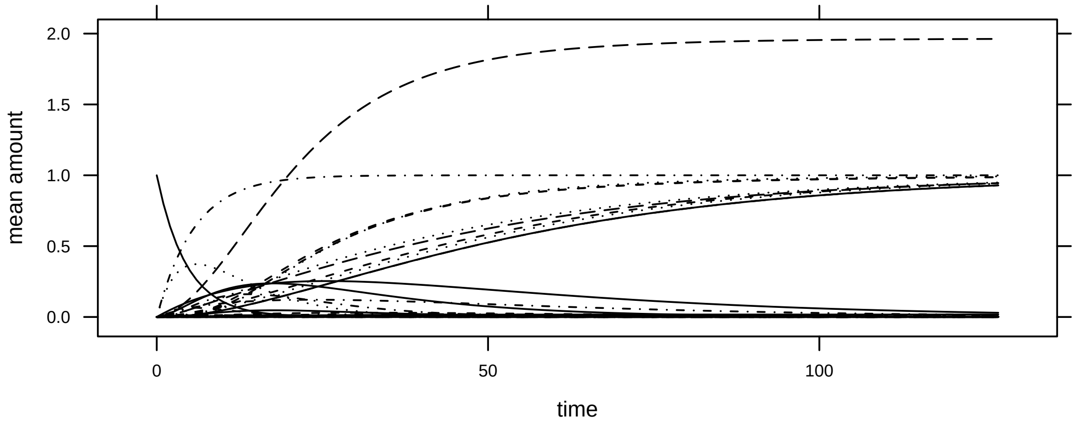


Figure 4.9: Time evolution of a population of peptides started from 1 unit of VAHRFKDLGEEN particles. As time passes all subsequences of length at least 2 get cut and all that remain is 2 units of a single amino acid sequence E (since there are two E amino acids in VAHRFKDLGEEN) and 1 unit of every other single amino acid sequence.

4.2.5 Results and conclusions

Our model has been extensively tested on simulated datasets. In this section we present the results obtained for modeling the activity of two proteolytic enzymes: trypsin and pepsin. Trypsin is a peptidase that specifically cleaves at the carboxylic side of lysine and arginine residues. The distribution of Lys and Arg residues in proteins make this enzyme useful for fragmenting long and heavy chains before a mass spectrometry analysis. Pepsin is most efficient in cleaving peptide bonds between hydrophobic and preferably aromatic amino acids such as phenylalanine, tryptophan, and tyrosine. It was the first animal enzyme to be discovered. Recently it is used in an on-line enzymatic digestion MS technique for rapid monitoring of chemical exposures after a terrorist or military attack with chemical agents (Carol-Visser et al., 2008).

The *human serum albumin* protein fragment VAHRFKDLGEEN has been digested in silico by trypsin and pepsin according to our model. Figure 4.9 illustrates the time evolution of the population of peptides from time 0 to 127. The starting expected amount of the VAHRFKDLGEEN peptide was set to 1 and all its subsequences to 0. The coefficients ρ_{vw} were calculated from affinity matrices

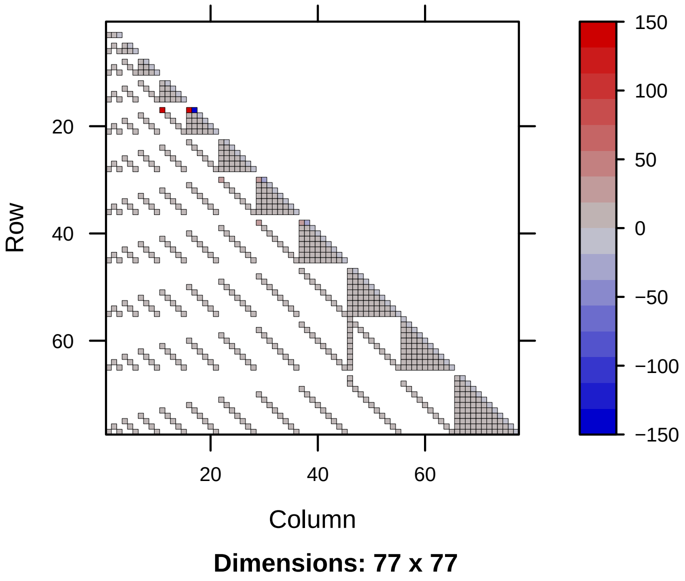


Figure 4.10: The graphical representation of the matrix $\Lambda = (\lambda_{vw})_{v,w \in \mathcal{V}}$ for peptide VAHRFKDLGEEN.

for trypsin and pepsin (see Figure 4.8). The cutting intensities c_{tryps} and c_{peps} were both set to 5×10^7. The structure of the cleavage graph is encoded in the matrix Λ presented in Figure 4.10. The cleavage $u = v \dagger w$ gives raise to non-zero parameters λ_{uv} and λ_{uw}. The matrix Λ has a triangular form, because the cutting operation always shortens the involved sequence.

The expected amounts of peptides were taken for time points $0, 1, 3, 7, 15, 31, 63$ and 127 and perturbed data was simulated by adding gaussian noise with standard deviation 0 (no perturbation), 0.1, 0.2 or 0.3. This step reflects the measurement errors in MS technology. Next a fraction (0%, 20% or 40%) of readings was randomly selected for removal from the datasets to mimic missing readings (i.e. corresponding to not being able to find a peak in the LC-MS spectrum for a given peptide). Thus in total there were 4 × 3 combinations of the perturbation level and missing readings number. For each combination 8 datasets were generated.

For each of the 96 datasets the L-BFGS-B method implemented in the function optim in R (Team, 2009) was run 8 times in order to recover the true cutting intensities $(c_{\text{tryps}}, c_{\text{peps}})$ and the true expected amounts of peptides at time 0 (denoted by $\mathrm{E}\,(0)$). The objective was to minimize the Φ function defined in Section 4.2.3. Best results (in terms of the Φ function) out of 8 runs were taken. The accuracy of the estimation procedure depends on the introduced noise level, as depicted in Figure 4.11.

The outcome of the modeling is very promising: the parameter estimation is robust to the noise in the data and it can handle datasets with missing values. The validation of our model on real data is planned. To this end we intend to tune the model using time series of good quality tandem mass spectrometry experiments for a very simple system (e.g. a single protein digested by one enzyme). After successful model tuning on easy experimental data we would like to cope with complex peptide mixtures, like human serum samples. It is worth noting that the peptidase activity model described in this paper has the potential to diagnose pathological states, particularly to predict cancer spread, as during the metastasis many proteolytic enzymes are engaged in the extracellular matrix digestion.

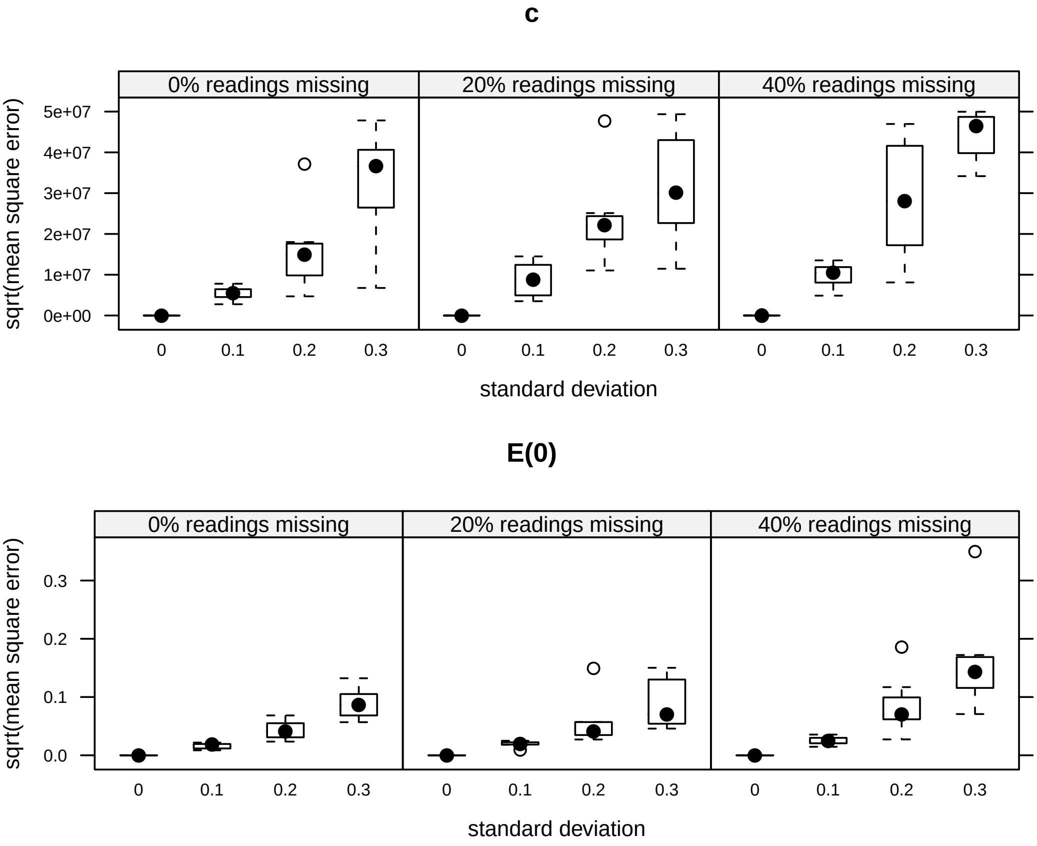


Figure 4.11: The influence of errors on the accuracy of the estimation procedure. Each boxplot is based on 8 datasets. The true c is a vector $(5 \times 10^7, 5 \times 10^7)$. The true $\mathrm{E}\,(0)$ is a vector of 76 zeroes and 1 one.

Chapter 5

Conclusions

In this thesis computational methods have been proposed for problems related to LC-MS data processing.

For the problem of isotopic envelopes detection and interpretation an efficient procedure based on the sweeping method has been developed. Its novelty is in looking at a 2-dimensional spectrum as a whole. The procedure has been implemented and shows sensitivity sufficient for medical applications such as searching for biomarkers. The software has been used by the MS laboratory of the Institute of Biophysics and Biochemistry of the Polish Academy of Sciences.

Two procedures have been presented that perform LC-MS spectra alignment. The problem is non-trivial due to large size of the datasets. The first method estimates retention time shifting and scaling with a Metropolis-Hastings algorithm, while the second one uses the Expectation-Maximization algorithm for gaussian mixture model based clustering inside preliminary clusters obtained from a DB-SCAN algorithm run. Both strategies are efficient enough to be used with real LC-MS datasets.

A framework based on the chemical master equation for inferring proteolytic activity from LC-MS data has been introduced. To my knowledge it is the first attempt to explicitly model the process of proteolysis from the LC-MS readings of the quantities of the peptides being cut. Two versions of the framework have been proposed. The first one assumes stationarity of the proteolysis process and handles endopeptidases only. The second extends the first one by describing the process in time (stationarity assumption is no longer needed), handling endopeptidases and integrating knowledge from the MEROPS peptidase database.

A big open challenge is the integration of different stages of MS processing and incorporation of knowledge from external sources (e.g. databases). One could imagine for example a framework where the isotopic envelopes detection is coupled with spectra alignment, so that low quality isotopic envelopes are enhanced by observing their similarity to envelopes in other spectra. As a result the quality of alignment can also be improved and both stages benefit from each other. Such mutual dependence can be captured by a bayesian model and is in fact the basis of estimation procedures like Gibbs sampling and Expectation-Maximization. With increasing computational power at our disposal we will be able to build bigger models having components for capturing different aspects of data resulting in increased quality of predictions.

Bibliography

B. L. Adam, Y. Qu, J. W. Davis, M. D. Ward, M. A. Clements, L. H. Cazares, O. J. Semmes, P. F. Schellhammer, Y. Yasui, Z. Feng, and G. L. J. Wright. Serum protein fingerprinting coupled with a pattern-matching algorithm distinguishes prostate cancer from benign prostate hyperplasia and healthy men. *Cancer Research*, 62:3609–3614, 2002.

J. Aitchison and J. J. Egozcue. Compositional data analysis: Where are we and where should we be heading? *Mathematical Geology*, 37(7):829–850, 2005.

Y. Benjamini and Y. Hochberg. Controlling the false discovery rate: a practical and powerful approach to multiple testing. *Journal of the Royal Statistical Society*, 57:289–300, 1995.

L. Breiman. Random forests. *Machine Learning*, 45(1):5–32, 2001.

D. Bylunda, R. Danielssona, G. Malmquistb, and K. E. Markides. Chromatographic alignment by warping and dynamic programming as a pre-processing tool for parafac modelling of liquid chromatographymass spectrometry data. *Journal of Chromatography A*, 961(2):237–244, 2002.

J. Carol-Visser, M. van der Schans, A. Fidder, A. G. Hulst, B. L. van Baar, H. Irth, and D. Noort. Development of an automated on-line pepsin digestion-liquid chromatography-tandem mass spectrometry configuration for the rapid analysis of protein adducts of chemical warfare agents. *Journal of Chromatography B*, 870(1):91–97, 2008.

A. Ciliberto, F. Capuani, and J. J. Tyson. Modeling networks of coupled enzymatic reactions using the total quasi-steady state approximation. *PLoS Computational Biology*, 3(3):e45, 2007.

W. S. Cleveland, E. Grosse, and W. M. Shyu. Local regression models. In J. M. Chambers and T. J. Hastie, editors, *Statistical Models in S*, pages 309–376, Pacific Grove: Wadsworth, 1992.

F. Delaglio, S. Grzesiek, G. W. Vuister, G. Zhu, J. Pfeifer, and A. Bax. NMRpipe: A multidimensional spectral processing system based on Unix pipes. *Journal of Biomolecular NMR*, 6:277–293, 1995.

A. P. Dempster, N. Laird, and D. Rubin. Maximum likelihood from incomplete data via the EM algorithm. *Journal of Royal Statististical Society*, Series B (39):1–38, 1977.

E. P. Diamandis. Proteomic patterns in biological fluids: Do they represent the future of cancer diagnostics? *Clinical Chemistry*, 49:1272–1275, 2003.

E. P. Diamandis. Mass spectrometry as a diagnostic and a cancer biomarker discovery tool: Opportunities and potential limitations. *Molecular & Cellular Proteomics*, 3:367–378, 2004.

E. P. Diamandis. Peptidomics for cancer diagnosis: Present and future. *Journal of Proteome Research*, 5(9):2079–2082, 2006.

B. Efron and R. Tibshirani. Improvements on cross-validation: The .632+ bootstrap method. *Journal of the American Statistical Association*, 92(438):548–560, 1997.

J. Esparza. Petri nets, commutative context-free grammars, and basic parallel processes. *Fundamenta Informaticae*, 31(1):13–25, 1997.

M. Ester, H.-P. Kriegel, J. Sander, and X. Xu. A density-based algorithm for discovering clusters in large spatial databases with noise. In E. Simoudis, J. Han, and U. Fayyad, editors, *Second International Conference on Knowledge Discovery and Data Mining*, pages 226–231. AAAI Press, 1996.

J. B. Fenn, M. Mann, C. K. Meng, S. F. Wong, and C. M. Whitehouse. Electrospray ionization for mass spectrometry of large biomolecules. *Science*, 246 (4926):64–71, 1989.

D. Figeys, I. L. Steward, and T. Thomson. ^{18}O labeling: a tool for proteomics. *Rapid Communnications in Mass Spectrometry*, 15:2456–2465, 2001.

C. Fraley and A. E. Raftery. How many clusters? which clustering method? answers via model-based cluster analysis. *The Computer Journal*, 41(8):578–588, 1998.

C. Fraley and A. E. Raftery. MCLUST: Software for model-based clustering, density estimation and discriminant. Technical Report 415R, University of Washington, Department of Statistics, 2002.

C. Fraley and A. E. Raftery. Bayesian regularization for normal mixture estimation and model-based clustering. *Journal of Classification*, 24:155–181, 2007.

A. Gambin and B. Kluge. Modeling proteolysis from mass spectrometry proteomic data. *Fundamenta Informaticae*, 103:89–104, 2010.

A. Gambin, M. Łuksza, B. Kluge, J. Ostrowski, and J. Karczmarski. Efficient model-based clustering for lc-ms data. *Workshop on Algorithms in Bioinformatics, LNBI*, 4175:32–43, 2006.

A. Gambin, J. Dutkowski, J. Karczmarski, B. Kluge, K. Kowalczyk, J. Ostrowski, J. Poznański, J. Tiuryn, M. Bakun, and M. Dadlez. Automated reduction and interpretation of multidimensional mass spectra for analysis of complex peptide mixtures. *International Journal of Mass Spectrometry*, 260:20–30, 2007.

E. Gelenbe. Network of interacting synthetic molecules in steady state. *Proceedings of the Royal Society A: Mathematical, Physical and Engineering Science*, 464(2096):2219–2228, 2008.

R. C. Gentleman, V. J. Carey, D. M. Bates, et al. Bioconductor: Open software development for computational biology and bioinformatics. *Genome Biology*, 5:R80, 2004.

P. Geurts, M. Fillet, D. de Seny, M. A. Meuwis, M. Malaise, M. P. Merville, and L. Wehenkel. Proteomic mass spectra classification using decision tree based ensemble methods. *Bioinformatics*, 21(14):3138–3145, 2005.

T. Goddard and D. Kneller. Sparky 3, 2006. University of California, San Francisco.

D. S. Goldobin and A. Zaikin. Towards quantitative prediction of proteasomal digestion patterns of pr oteins. *Journal of Statistical Mechanics: Theory and Experiment*, 2009(01):P01009, 2009.

W. K. Hastings. Monte carlo sampling methods using markov chains and their applications. *Biometrika*, 57(1):97–109, 1970.

D. M. A. Haughton. On the choice of a model to fit data from an exponential family. *The Annals of Statistics*, 16:342–355, 1988.

D. M. Horn, R. A. Zubarev, and F. W. McLafferty. Automated reduction and interpretation of high resolution electrospray mass spectra of large molecules. *Journal of the American Society for Mass Spectrometry*, 11(4):320–332, 2000.

J. Hu, K. R. Coombes, J. S. Morris, and K. A. Baggerly. The importance of experimental design in proteomic mass spectrometry experiments: Some cautionary tales. *Briefings in Functional Genomics and Proteomics*, 3(4):322–331, 2005.

Y. Igarashi, A. Eroshkin, S. Gramatikova, G. Gramatikoff, Y. Zhang, J. Smith, A. Osterman, and A. Godzik. Cutdb: a proteolytic event database. *Nucleic Acids Research*, 2007.

I. J. Jacobs and U. Menon. Progress and challenges in screening for early detection of ovarian cancer. *Molecular & Cellular Proteomics*, 3:355–366, 2004.

T. Jahnke and W. Huisinga. Solving the chemical master equation for monomolecular reaction systems analytically. *Journal of Mathematical Biology*, 54(1):1–26, 2007.

N. Jaitly, M. E. Monroe, V. A. Petyuk, T. R. W. Clauss, J. N. Adkins, and R. D. Smith. Robust algorithm for alignment of liquid chromatography-mass spectrometry analyses in an accurate mass and time tag data analysis pipeline. *Analytical Chemistry*, 78(21):7397–7409, 2006.

G. M. Janini, T. P. Conrads, T. D. Veenstra, H. J. Issaq, and K. C. Chan. Multidimensional separation of peptides for effective proteomic analysis. *Journal of Chromatography B*, 817(1):35–47, 2005.

J. Keeler. *Understanding NMR Spectroscopy.* John Wiley & Sons, 2005.

B. Kluge, A. Gambin, and W. Niemiro. Modeling exopeptidase activity from lc-ms data. *Journal of Computational Biology*, 16(2):395–406, 2009.

J. M. Koomen, D. Li, L.-c. Xiao, T. C. Liu, K. R. Coombes, J. Abbruzzese, and R. Kobayashi. Direct tandem mass spectrometry reveals limitations in protein profiling experiments for plasma biomarker discovery. *Journal of Proteome Research*, 4(3):972–981, 2005.

E. Lange, C. Gröpl, O. Schulz-Trieglaff, A. Leinenbach, C. Huber, and K. Reinert. A geometric approach for the alignment of liquid chromatography-mass spectrometry data. *Bioinformatics*, 23(13):i273–i281, 2007.

E. Lange, R. Tautenhahn, S. Neumann, and C. Gröpl. Critical assessment of alignment procedures for lc-ms proteomics and metabolomics measurements. *BMC Bioinformatics*, 9:375, 2008.

M. F. Leeman, S. Curran, and G. I. Murray. New insights into the roles of matrix metalloproteinases in colorectal cancer development and progression. *Journal of Pathology*, 201:528–534, 2003.

J. Li, Z. Zhang, J. Rosenzweig, Y. Y. Wang, and D. W. Chan. Proteomics and bioinformatics approaches for identification of serum biomarkers to detect breast cancer. *Clinical Chemistry*, 48:1296–1304, 2002.

X. J. Li, E. C. Kemp, H. Zhang, and R. Aebersold. A software suite for the generation and comparison of peptide arrays from sets of data collected by liquid chromatography-mass spectrometry. *Molecular & Cellular Proteomics*, 4(9):1328–1340, 2005.

R. H. Lilien, H. Farid, and B. R. Donald. Probabilistic disease classification of expression-dependent proteomic data from mass spectrometry of human serum. *Journal of Computational Biology*, 10(6):925–946, 2003.

L. A. Liotta and E. F. Petricoin. Serum peptidome for cancer detection: spinning biologic trash into diagnostic gold. *Journal of Clinical Investigation*, 116(1): 26–30, 2006.

P. Lu, J. Nocedal, C. Zhu, and R. H. Byrd. A limited-memory algorithm for bound constrained optimization. *SIAM Journal on Scientific Computing*, 16: 1190–1208, 1994.

M. Łuksza, B. Kluge, J. Ostrowski, J. Karczmarski, and A. Gambin. Two-stage model-based clustering for liquid chromatography mass spectrometry data analysis. *Statistical Applications in Genetics and Molecular Biology*, 8 (1):Article 15, 2009.

P. Mallick, M. Schirle, S. S. Chen, M. R. Flory, H. Lee, D. Martin, J. Ranish, B. Raught, R. Schmitt, T. Werner, B. Kuster, and R. Aebersold. Computational prediction of proteotypic peptides for quantitative proteomics. *Nature Biotechnology*, 25(1):125–131, 2007.

R. E. March. Quadrupole ion trap mass spectrometry: a view at the turn of the century. *International Journal of Mass Spectrometry*, 200(1–3):285–312, 2000.

A. G. Marshall, C. L. Hendrickson, and G. S. Jackson. Fourier transform ion cyclotron resonance mass spectrometry: A primer. *Mass Spectrometry Reviews*, 17(1):1–35, 1998.

J. Marshall, P. Kupchak, W. Zhu, J. Yantha, T. Vrees, S. Furesz, K. Jacks, C. Smith, I. Kireeva, R. Zhang, M. Takahashi, E. Stanton, and G. Jackowski.

Processing of serum proteins underlies the mass spectral fingerprinting of myocardial infarction. *Journal of Proteome Research*, 2(4):361–372, 2003.

T. Masaki, H. Matsuoka, M. Sugiyama, N. Abe, A. Goto, A. Sakamoto, and Y. Atomi. Matrilysin (mmp-7) as a significant determinant of malignant potential of early invasive colorectal carcinomas. *British Journal of Cancer*, 84: 1317–1321, 2001.

L. Matrisian and S. Sledge, G.W. Mohlaet. Extracellular proteolysis and cancer: Meeting summary and future directions. *Cancer Research*, 63(19):6105–6109, 2003.

T. Minka. Expectation-maximization as lower bound maximization, 1998. URL `http://research.microsoft.com/en-us/um/people/minka/papers/em.html`.

C. Moler and C. V. Loan. Nineteen dubious ways to compute the exponential of a matrix, twenty-five years later. *SIAM Review*, 45(1):3–49, 2003.

C. G. Moles, P. Mendes, and J. R. Banga. Parameter estimation in biochemical pathways: a comparison of global optimization methods. *Genome Research*, 13:2467–2474, 2003.

R. T. Ng and J. Han. Efficient and effective clustering methods for spatial data mining. In J. Bocca, M. Jarke, and C. Zaniolo, editors, *20th International Conference on Very Large Data Bases, September 12–15, 1994, Santiago, Chile proceedings*, pages 144–155, Los Altos, CA 94022, USA, 1994. Morgan Kaufmann Publishers.

J. Nocedal and S. J. Wright. *Numerical optimization*. Springer, 1999.

J. Pahle. Biochemical simulations: stochastic, approximate stochastic and hybrid approaches. *Briefings in Bioinformatics*, 10(1):53–64, 2009.

E. F. Petricoin, A. M. Ardekani, B. A. Hitt, P. J. Levine, V. A. Fusaro, S. M. Steinberg, G. B. Mills, C. Simone, D. A. Fishman, E. C. Kohn, and L. A. Liotta. Use of proteomic patterns in serum to identify ovarian cancer. *Lancet*, 359:572–577, 2002.

J. T. Prince and E. M. Marcotte. Chromatographic alignment of esi-lc-ms proteomics data sets by ordered bijective interpolated warping. *Analytical Chemistry*, 78(17):6140–6152, 2006.

N. D. Rawlings and A. J. Barrett. Merops: the peptidase database. *Nucleic Acids Research*, 28(1):323–325, 2000.

S. E. Reznik and L. D. Fricker. Carboxypeptidases from a to z: implications in embryonic development and wnt binding. *Cellular and Molecular Life Sciences*, 58(12–13):1790–1804, 2001.

RNCOS. Global bioinformatics market outlook. Research Report, 2010. URL http://www.rncos.com/Report/IM554.htm.

B. Schölkopf and A. J. Smola. *Learning with Kernels: Support Vector Machines, Regularization, Optimization, and Beyond*. MIT Press, 2002.

G. Schwarz. Estimating the dimension of a model. *The Annals of Statistics*, 6: 461–464, 1978.

M. W. Senko, S. C. Beu, and F. W. McLafferty. Determination of monoisotopic masses and ion populations for large biomolecules from resolved isotopic distributions. *Journal of the American Society for Mass Spectrometry*, 6(4):229–233, 1995a.

M. W. Senko, S. C. Beu, and F. W. McLafferty. Automated assignment of charge states from resolved isotopic peaks for multiply charged ions. *Journal of the American Society for Mass Spectrometry*, 6(1):52–56, 1995b.

A. A. Shvartsburg, E. F. Strittmatter, R. Smith, K. Tang, and F. Li. Two-dimensional gas-phase separation coupled to mass spectrometry for analysis of complex mixtures. *Analytical Chemistry*, 77(9):6381–6388, 2005.

C. A. Smith, E. J. Want, G. O'Maille, R. Abagyan, and G. Siuzdak. XCMS: Processing mass spectrometry data for metabolite profiling using nonlinear peak alignment, matching, and identification. *Analytical Chemistry*, 78(3): 779–787, 2006.

D. Storey. The positive false discovery rate: A bayesian interpretation and the q-value. *Annals of Statistics*, 31(6):2013–2035, 2003.

J. Storey and R. Tibshirani. Statistical significance for genomewide studies. *Proceedings of the National Academy of Sciences*, 100(16):9440–9445, 2003.

R. D. C. Team. *R: A Language and Environment for Statistical Computing*. R Foundation for Statistical Computing, Vienna, Austria, 2009. URL http://www.R-project.org.

R. Tibshirani, T. Hastie, B. Narasimhan, S. Soltys, G. Shi, A. Koong, and Q. T. Le. Sample classification from protein mass spectrometry, by 'peak probability contrasts'. *Bioinformatics*, 20:3034–3044, 2004.

T. E. Turner, S. Schnell, and K. Burrage. Stochastic approaches for modelling in vivo reactions. *Computational Biology and Chemistry*, 28:165–178, 2004.

S. N. Twigger, B. D. Halligan, and R. Y. Slyper. Zoomquant: An application for the quantitation of stable isotope labeled peptides. *Journal of The American Society for Mass Spectrometry*, 16:302–306, 2005.

D. Valkenborg and T. Burzykowski. A markov-chain model for the analysis of high-resolution enzymatically ^{18}O-labeled mass spectra. *Statistical Applications in Genetics and Molecular Biology*, 10(1):Article 1, 2011.

N. M. Verrills. Clinical proteomics: present and future prospects. *Clinical Biochemist Reviews*, 27(2):99–116, 2006.

J. Villanueva, A. Martorella, K. Lawlor, J. Philip, M. Fleisher, R. Robbins, and T. P. Serum peptidome patterns that distinguish metastatic thyroid carcinoma from cancer-free controls are unbiased by gender and age. *Mol. Cell. Proteomics*, 5:1840–1852, 2006a.

J. Villanueva, D. R. Shaffer, J. Philip, C. A. Chaparro, H. Erdjument-Bromage, A. B. Olshen, M. Fleisher, H. Lilja, E. Brogi, J. Boyd, M. Sanchez-Carbayo, E. C. Holland, C. Cordon-Cardo, H. I. Scher, and P. Tempst. Differential exoprotease activities confer tumor-specific serum peptidome patterns. *Journal of Clinical Investigation*, 116(1):271–284, 2006b.

J. Villanueva, A. Nazarian, K. Lawlor, S. Y. San, R. J. Robbins, and P. Tempst. A sequence-specific exopeptidase activity test (sseat) for 'functional' biomarker discovery. *Molecular & Cellular Proteomics*, 7(3):509–518, 2008.

B. Wu, T. Abbott, D. Fishman, W. McMurray, G. Mor, K. Stone, D. Ward, K. Williams, and H. Zhao. Comparison of statistical methods for classification of ovarian cancer using mass spectrometry data. *Bioinformatics*, 19(13):1636–1643, 2003.

K. Y. Yeung, C. Fraley, A. Murua, A. E. Raftery, and W. L. Ruzzo. Model-based clustering and data transformations for gene expression data. *Bioinformatics*, 17(10):977–987, 2001.

J. S. Yu, S. Ongarello, R. Fiedler, X. W. Chen, G. Toffolo, C. Cobelli, and Z. Trajanoski. Ovarian cancer identification based on dimensionality reduction for high-throughput mass spectrometry data. *Bioinformatics*, 21(10):2200–2209, 2005.

Z. Zhang and A. G. Marshall. A universal algorithm for fast and automated charge state deconvolution of electrospray mass-to-charge ratio spectra. *Journal of the American Society for Mass Spectrometry*, 9(3):225–233, 1998.