MASS SPECTROMETRY DATA MINING FOR CANCER DETECTION

A Dissertation

Presented to

the Faculty of the Department of Mathematics

University of Houston

In Partial Fulfillment

of the Requirements for the Degree

Doctor of Philosophy

By

Ao Kong

December 2013

MASS SPECTROMETRY DATA MINING FOR CANCER DETECTION

Ao Kong

APPROVED:

Robert Azencott, Chairman
Dept. of Mathematics

Krešimir Josić
Dept. of Mathematics

Matthew Nicol
Dept. of Mathematics

David Hawke
Dept. of Pathology, MD Anderson Cancer Center

Dean, College of Natural Sciences and Mathematics

Acknowledgments

I would like to express my deepest gratitude to my advisor Dr. Robert Azencott for his expert guidance, patience, and encouragement throughout my study and research. His great insight and enormous experience lead me all the way to build this dissertation. I feel especially grateful for his availability to regularly meet with me once a week despite his busy schedule and his thoughtful advice to me and my family on our future plans. He has played the role of both a patient teacher and a kind father to bloom my path of career and life.

I am also very grateful to Dr. Chinmaya Gupta for his guidance and assistance to get me started on my first project. I have learned much from him. Thanks also go to Dr. Ennio Tasciotti and Chiara Bedin at the Methodist Hospital Research Institute. It has been an enjoyable time collaborating with them on that project.

I want to thank Dr. David Hawke, Dr. Philip Lorenzi and Dr. John Weinstein at MD Anderson Cancer Center for offering me an opportunity to work in their department during the summer 2013 and for guiding me throughout that summer project.

I would like to thank the committee members Dr. Krešimir Josić, Dr. David Hawke and Dr. Matthew Nicol for serving on my dissertation defense committee, and for their insightful suggestions for my dissertation.

I thank all my fellow graduate students, who made my stay at University of Houston incredible.

This dissertation is dedicated to my husband Xiaolei Qu and my parents Lingqi Kong and Meijuan Jin. I would not have been able to complete this dissertation without their unconditional love and continuous support.

MASS SPECTROMETRY DATA MINING FOR CANCER DETECTION

An Abstract of a Dissertation
Presented to
the Faculty of the Department of Mathematics
University of Houston

In Partial Fulfillment
of the Requirements for the Degree
Doctor of Philosophy

By
Ao Kong
December 2013

Abstract

Early detection of cancer is crucial for successful intervention strategies. Mass spectrometry-based high throughput proteomics is recognized as a major breakthrough in cancer detection. Many machine learning methods have been used to construct classifiers based on mass spectrometry data for discriminating between cancer stages, yet, the classifiers so constructed generally lack biological interpretability. To better assist clinical uses, a key step is to discover biomarker signature profiles, i.e. combinations of a small number of protein biomarkers strongly discriminating between cancer states.

This dissertation introduces two innovative algorithms to automatically search for a signature and to construct a high-performance signature-based classifier for cancer discrimination tasks based on mass spectrometry data, such as data acquired by MALDI or SELDI techniques. Our first algorithm assumes that homogeneous groups of mass spectra can be modeled by (unknown) Gibbs distributions to generate an optimal signature and an associated signature-based classifier by robust log-likelihood analysis; our second algorithm uses a stochastic optimization algorithm to search for two lists of biomarkers, and then constructs a signature-based classifier.

To support these two algorithms theoretically, this dissertation also studies the empirical probability distributions of mass spectrometry data and implements the actual fitting of Markov random fields to these high-dimensional distributions. We have validated our two signature discovery algorithms on several mass spectrometry datasets related to ovarian cancer and to colorectal cancer patients groups. For these cancer discrimination tasks, our algorithms have yielded better classification performances than existing machine learning algorithms and in addition,have generated

more interpretable explicit signatures.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

1.1 Background

In proteomics, mass spectrometry is a broadly used protein profiling technology to study the mixture of proteins/peptides present in biological tissues or fluids. Such study provides highly efficient potential tools for protein-based identification of cancer diseases, disease progression monitoring, and treatment response.

The achievements of mass spectrometry in analyzing proteins is due to the development of soft ionization techniques. There are three commonly used ionization methods: the electrospray ionisation (ESI), the matrix-assisted laser desorption ionization (MALDI), and surface-enhanced laser desorption and ionization (SELDI) [30]. The ions are separated based on their mass to charge (m/z) ratios in another part of the instrument, namely the mass analyzer, to generate information-rich mass spectra. Four primary types of mass analyzers are the ion trap, the time-of-flight (TOF), the quadrupole, and the Fourier transform ion cyclotron (FT-MS) analyzer [3]. Due to their simplicity, accuracy, high resolution, and sensitivity, MALDI and SELDI are popularly used in current research, usually coupled with TOF analyzer.

MALDI and SELDI generate high-dimensional mass spectra from specimens. Biological information contained in the mass spectra is investigated by expert empirical analysis or complex statistical analysis for medical diagnosis and prognosis. Recent development recognizes proteomic biomarkers, a non-invasive screening method, as a breakthrough to extract such useful information from mass spectra [64].

Managing and analyzing high-dimensional data is currently a challenging task for biomarker detection, since most mass spectra can have several tens of thousands data points [44]. Moreover, data acquisition is blurred by variations and errors introduced by data acquisition split into successive distinct sessions, causing difficulties in data pre-processing. Purely empirical analysis by biologists can not satisfy such requirements. Mass spectra analysis is relying more and more on statistical and machine learning technologies.

Proteomic profiles analysis aims to lower mass spectra data dimensionality, to detect discriminating proteomic biomarkers, and to classify automatically new mass spectra via differentially expressed biomarkers. These three basic steps are usually referred to as the pre-processing, biomarker detection, and group classification. The pre-processing phase is usually performed by commercial software such as Ciphergen ProteinChip, Markerview, and PROcess, or in-house software such as LMS, LIMPIC, SpecAlign [92], MassSpecWavelet [94], and Cromwell Package [24]. Biomarker detection normally depends on in-house algorithms. Sample classification, as the final goal of mass spectra analysis, has been actively and promisingly performed by utilizing various machine learning technique in recently years. However, there is no gold standard in computational proteomics as these three stages are usually performed through different proprietary algorithms tailored to particular applications [75].

In the sample classification step, popular machine learning algorithms such as support vector machines, artificial neural networks, or k-nearest neighbors usually generate "black-box" classifiers, which are difficult to interpret or extend biologically. To develop clinically usable mass spectra analysis tools, a key step is to discover "signature profiles", i.e. combinations of a small number of protein biomarkers strongly discriminating between cancer states [96] [100] [3] [33]. However, to reach this goal, the patterns of mass spectrometry datasets need to be understood in depth.

In this study, we have innovatively investigated discrimination between homogeneous groups of mass spectrometry datasets in the context of Markov Random Field (MRF) models and developed several novel algorithms for automatic discovery of biologically interpretable "signature profiles" to solve multiple discrimination tasks. We have tested our signature discovery algorithms on a new MALDI-TOF dataset for colorectal cancer and two well-known ovarian cancer SELDI-TOF datasets. We have generated explicit signatures with high discriminating power between the various cancer patients groups involved in these data. We have compared performances between our optimized signature-based classifiers and several benchmark machine learning techniques.

1.2 Literature Review

1.2.1 Pre-processing

Pre-processing software normally include more or less the same substeps, namely data denoising (or smoothing), peak normalization, baseline removal, peak detection and peak alignment, implemented by various well known algorithmic techniques.

Denoising (or smoothing) aims to reduce the high-frequency noise caused by sources such as electrical interference, random ion motions, statistical fluctuation in the detector gain, or chemical impurities [25]. The most commonly used denoising approaches are smoothing, wavelet transform, and deconvolution filters [80]. Typical smoothing filters are the Gaussian filter [88] [102], the moving average filter [55] [52] [49] [40] [31], and the Savitzky-Golay filter [49] [40], which smooth out the noise by averaging intensities over a moving window. Wavelet transform (WT) employs widely used signal process techniques to investigate mass spectra, which includes continuous wavelet transform(CWT) [27] [45], discrete wavelet transform (DWT) [71] [11] [101], and undecimated discrete wavelet transform(UDWT) [24] [39]. In [56], it is assumed that the spectra can be modeled as a sum of peaks with some parametric form and implemented a deconvolution filter to study the true peaks and its additive noise.

It is not feasible to compare peak intensities between different spectra directly by magnitudes due to the differences in the protein amounts in the sample of protein. Normalization rescales the intensities of mass spectra and, therefore, to facilitate mass spectra comparisons. The commonly used normalization methods within each spectrum, involve normalization by rescaling with respect to total ion current (TIC) [62] [74] [91] [59] [90] [7] [42] [50] [43] [63] or with respect to a specific control peak [35] [87] [22] [65] [82]. Normalization across samples has also been studied and adjusts intensities for all spectra with respect to a global scaling value [66] [81] [67] [78].

Mass spectra also exhibit decreasing baselines, due to the matrix material added to the sample of interest. Methods to remove the baseline can be divided to two categories: Heuristic or model-based [80]. Heuristic approaches estimate the baseline by averaging or minimizing intensities over a local sliding window [7] [52] [28], applying piecewise regression [23] [87] [60] [34] [52] [96] [11] [63] or computing the convex hull of intensities in a region [55]. Model-based approaches design and fit a model for the pattern of the baseline [21] [56] [27] [45].

The peak detection and peak alignment subtasks are sometimes referred to as a whole feature extraction step, which results in a significant reduction in the number of original peaks. It locates the true peaks that carry proteomic information. Since peak intensities are correlated and practically a peak covers a region instead of being located at a precise abscissa, binning is the most frequently used technique to reduce dimensionality [7] [84] [22] [63] [52] [74] [97]. After this, peak detection is normally based on a signal-to-noise ratio (SNR) [24] [28] [57] [40] [52] [39], local maximum [55] [95], slopes of peaks [57], shape ratio [52] [11], or other model-based criteria [45] [49]. Peak alignment among several mass spectra regroups neighboring peaks into isolated single peaks, then, regroup peaks into a global list of prominent peaks to correct misalignment and to cross reference peaks across different mass spectra [92]

[2] [42] [43] [51] [81] [87]. A few other studies project the original space into another low-dimensional space with principle component analysis (PCA) and extract features from this new space, such as [97] [53].

The operation of pre-processing software is essentially user-dependable because subjective parameters exist at almost all substeps and in addition, the order of all substeps is subject to user's decision [25]. Any combination of different parameters and of preprocessing substeps may significantly alter the extracted features, which is a drawback for reproducibility.

1.2.2 Biomarker Detection

Biomarker detection, sometimes called feature selection, focuses on identifying a small list of peaks for discrimination purpose. The three most frequently used feature selection methods are filter, wrappers or embedded methods [77]. Filter methods, such as t-test [54] [93] [63], F-test [14], and peak probabilities comparison [84], rank and select peaks based on some computed statistics. Wrapper methods (e.g., [51] [74]), in the contest of preparation for hypothesis testing, evaluate various subsets of features by training and testing a classification model and chooses the best one. Embedded methods, just like wrapper methods, embed feature election in the training phase of classifiers. Embedded methods have been increasingly exploited in studies based on decision trees [31], random forests [32] [93], Support Vector Machines (SVM) [38] [68] [100], and Neural Networks [9] [74].

1.2.3 Group Classification

For the classification phase, bioengineering papers tend to employ machine learning algorithms to automatically discover reliable classifiers from pre-processed samples. Machine learning, including unsupervised and supervised learning methods, is a branch of artificial intelligence concerned with the design and evaluation of algorithms that enable the statistical model to self-modify its own parameters in view of previous results. Unsupervised learning aims to partition data sets into homogeneous subgroups based on some criteria that define how ”similar” two samples are. A few studies based on clustering [67] [70] or self-organizing maps [65] [22] [61] are of this type. Supervised learning, which analyzes pre-classified mass spectra data sets to generate automated classifiers, has been extensively applied for cancer discrimination: artificial neural networks (ANN) [9] [35] [67] [68] [46], decision trees [2] [73] [86] [10] [35] [60] [101] [68] [31] [79], boosted decision tree [72], k-nearest neighbors [102] [35] [87] [68] [79], random forests [37] [79], linear and quadratic discriminant analysis [93] [7] [53] [71] [87] [93] [79], and support vector machines (SVM) [93] [87] [68] [38] [51][97] [100] [90] [79].

1.3 Outline of Dissertation

The dissertation is divided into twelve chapters including the introduction.

Chapter 2 introduces the mass spectrometry technology. It gives a brief description of two commonly used mass spectrometry technique, MALDI-TOF and

SELDI-TOF, and the generated mass spectrometry datasets.

Because raw mass spectra are blurred by experimental variations and errors, a typical mass spectrometry analysis starts with a pre-processing step, Chapter 3 presents a stepwise pre-processing algorithm, which includes several key steps: normalization, smoothing, baseline removal and peak detection. It is quite important to isolate not only the strong peaks, but also the peaks that are highly reliable. We design a peak scoring method to evaluate the reliability of detected peaks, which will be used to compare the reliability of discovered signatures. To further facilitate mass spectra comparison and classification, we condense all detected peaks into a list of well separated reference peaks, within which we will select adequate signatures.

It has fairly often been suggested that better classification performance can be achieved by combining several biomarkers, yet the underlying links among peaks are quite often investigated rather superficially. Chapter 4 innovatively studies co-occurence patterns among peaks within a Markov Random Field framework and proposes to fit Gibbs models to the observed empirical distributions of simultaneous peak occurrences. Generalized from published likelihood-based techniques, three parameter estimation methods for Gibbs model fitting are developed.

To prove the feasibility of the methodology proposed in Chapter 4, Chapter 5 then applies these three parameter estimation methods to a group of mass spectrometry samples in order to parametrize an underlying Gibbs distribution.

Chapter 6 derives, for a typical discrimination task between two groups G^+ and G^-, an optimal classifier based on the fitting of two distinct Gibbs models to G^+ and

G^{-}. The optimal classifier turns out to be a linear function on an extended sample space.

The main goal of our study is to discover "signature profiles"based on identification of discriminating patterns between mass spectrometry samples from different patient groups. Chapter 7 sets up the framework of signature discovery algorithm by giving the definition of signatures and sketching a general procedure for signature based classification. Chapter 8 and 9 present, in detail, two signature discovery algorithms.

Although the optimal classifier between two Gibbs models is known and easy to compute, a practical problem is that it is hard to collect sufficient samples to accurately estimate Gibbs models. This motivates us, in Chapter 8, to design a robust log-likelihood (RLL) method to seek such an optimal classifier without preliminary fitting of Gibbs models. Chapter 9 then proposes another innovative algorithm that discovers signature profiles by Maximing Detecting Power (MDP). We implement this optimization by Simulated Annealing. We then validate separately each of the algorithms on simulated datasets and compute their convergence rate in the framework of large deviation theory.

In Chapter 10, we introduce two real mass spectrometry datasets, which are related to colorectal cancer and ovarian cancer. Our two signature discovery algorithms are tested on these datasets to discover signatures for cancer stage discrimination. Group classification performance is computed for our signature-based classifiers.

To handle the high dimensionality of mass spectrometry data, and their inherent variability, ”machine learning” algorithms have been a popular approach to facilitate automatic classification between mass spectra. In chapter 11, to emphasize the advantage of our signature discovery algorithms, we compare their performances with those of several popular machine learning algorithms on the real mass spectrometry datasets.

Chapter 12 summarizes the achievements of our study and the advantage of our signature based discrimination algorithms. It also points out our future research goals in this area.

Chapter 2

Mass Spectra Acquisition and Cancer Discrimination

This chapter gives a brief description of the two most commonly used mass spectrometry technologies (MALDI-TOF and SELDI-TOF) and the generated mass spectrometry datasets. Since cancer detection is our final goal for analyzing mass spectra, we also introduce generically the cancer discrimination problem we will be dealing with throughout this dissertation.

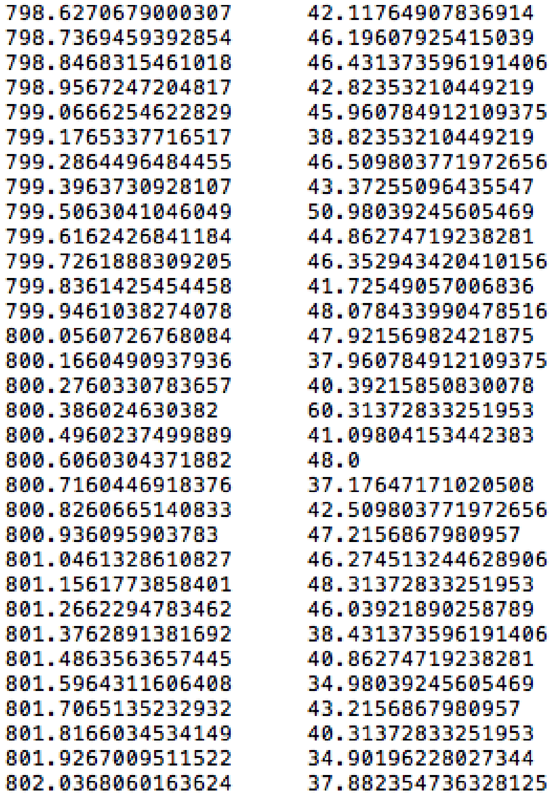
2.1 Mass Spectrometry Technology

A mass spectrometer platform includes: a soft ionization system (i), a mass analyzer (ii), an ion detector (iii), and a system to output and store spectra (iv).

Biological samples are first imported directly into (i) or are purified to simple proteins by biochemical fractionation or affinity selection, which are inserted into (i) sequentially. Samples whether degraded or not are then mixed with an energy-absorbing matrix material that allows them to crystallize before being placed on a steel plate. SELDI, a modification of MALDI, adds special chemistry on the surface of the plate to capture specific proteins. The plate is placed into a vacuum chamber (ii) where a laser hits the plate leading to ionization of proteins into particles. These ionized particles are then accelerated by an electric field and fly into a time-of-flight tube, where the time for the particles to fly through the tube is a function of the molecular weight and charge of the protein. (iii) at the end of the tube, a sensor records the number of particles with their time of flights (TOF) over a short time interval. Each TOF is transformed into a particle mass-to-charge (m/z) and is stored with its corresponding particle "intensity" as a data point. A mass spectra gathers up to tens of thousands of such (m/z, intensity) pairs for a single biological sample. A spectrometer usually processes hundreds of biological samples simultaneously, therefore produces a high-dimensional dataset. This massive dataset is the mass spectra acquisition output and is stored by (iv), which is usually in a text file.

2.2 Mass Spectra Datasets

One mass spectrometry data folder may have hundreds of .txt files, each .txt file contains one mass spectrum from one biological sample. Two lists of numbers are saved in one such .txt file: the left column lists the m/z values and the right column lists the corresponding intensity values, as shown in Figure 2.1.



```
798.6270679000307      42.11764907836914
798.7369459392854      46.19607925415039
798.8468315461018      46.431373596191406
798.9567247204817      42.82353210449219
799.0666254622829      45.960784912109375
799.1765337716517      38.82353210449219
799.2864496484455      46.509803771972656
799.3963730928107      43.37255096435547
799.5063041046049      50.98039245605469
799.6162426841184      44.86274719238281
799.7261888309205      46.352943420410156
799.8361425454458      41.72549057006836
799.9461038274078      48.078433990478516
800.0560726768084      47.92156982421875
800.1660490937936      37.960784912109375
800.2760330783657      40.39215850830078
800.386024630382       60.31372833251953
800.4960237499889      41.09804153442383
800.6060304371882      48.0
800.7160446918376      37.17647171020508
800.8260665140833      42.509803771972656
800.936095903783       47.2156867980957
801.0461328610827      46.274513244628906
801.1561773858401      48.31372833251953
801.2662294783462      46.03921890258789
801.3762891381692      38.431373596191406
801.4863563657445      40.86274719238281
801.5964311606408      34.98039245605469
801.7065135232932      43.2156867980957
801.8166034534149      40.31372833251953
801.9267009511522      34.90196228027344
802.0368060163624      37.882354736328125
```

Figure 2.1: An example of a .txt mass spectrum file

Figure 2.2 displays, as an example, one mass spectrum obtained from human protein sample by MALDI-TOF technique. This spectrum has 36,930 distinct (m/z, intensity) points, which are connected by piecewise lines in our figure.

Two spectra from two subjects may have largely different intensity scales, even if they are from the same group. Figure 2.3 compares two spectra from two patients with the same cancer, one of which has a largest peak with intensity larger than 600 and the other one has all peak intensities inferior to 200. Therefore, to make spectra

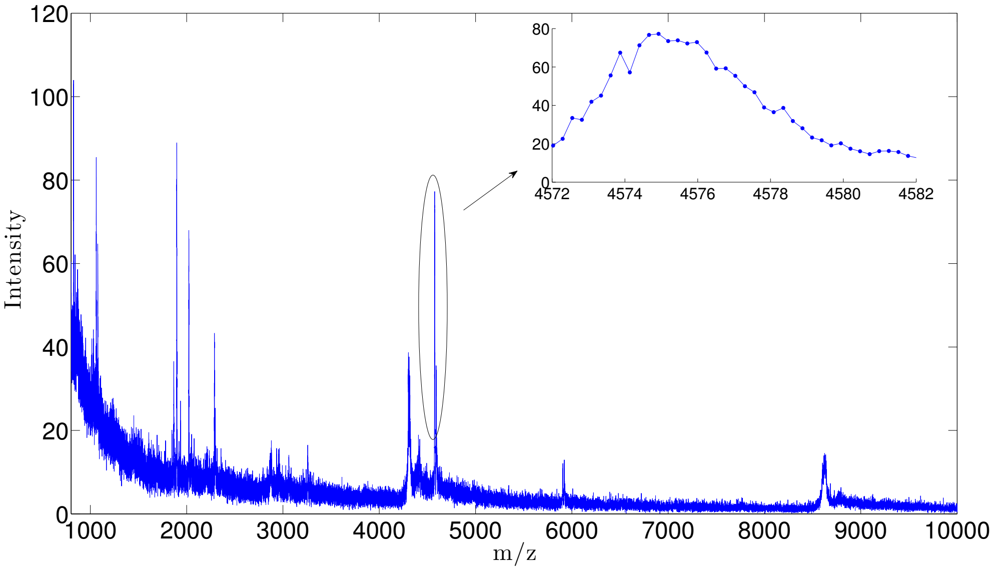


Figure 2.2: A raw mass spectrum

comparable, normalization is needed in pre-processing (see Section 1.2.1).

Mass spectra have high-frequency noise caused by electrical interference or random ion motions. In order to reveal the true patterns of mass spectra, denoising (or smoothing) is needed in pre-processing (see Section 1.2.1). For example, in Figure 2.4, the peak pattern of the original spectrum (blue) is really hard to identify because of its high frequency noise. The red curve has smoothed it to facilitate peak detection.

Spectra display numerous oscillations, modelized as peaks above a slowly varying baseline, which can also largely affect the intensity scale of spectra. The baseline effect is due to the cluster of matrix material on peptide particles, which can be very abundant in the lower mass range and decreases dramatically with the increasing of mass range. Figure 2.5 gives an example of the baseline computed for the spectrum

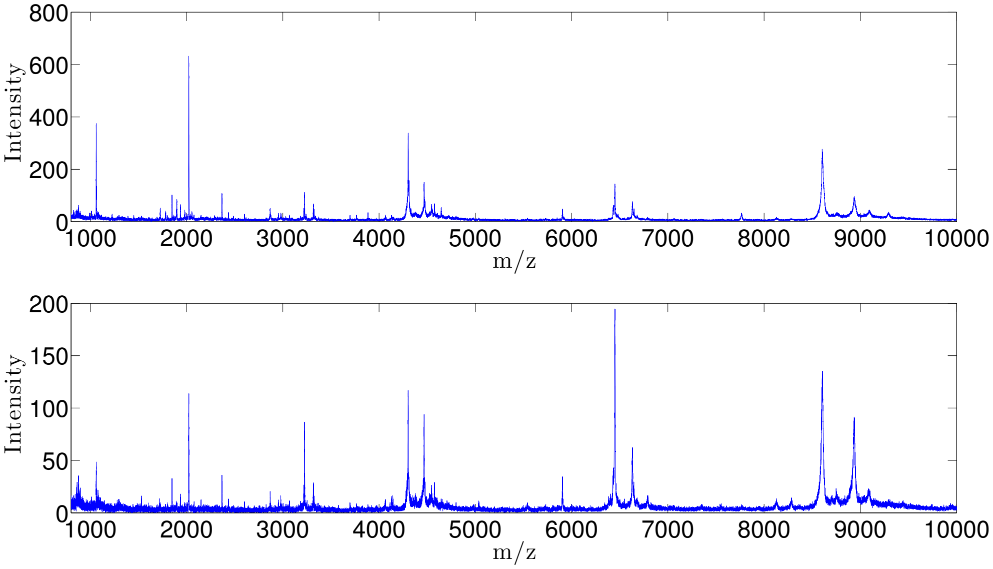


Figure 2.3: Comparison of intensity scales between two spectra

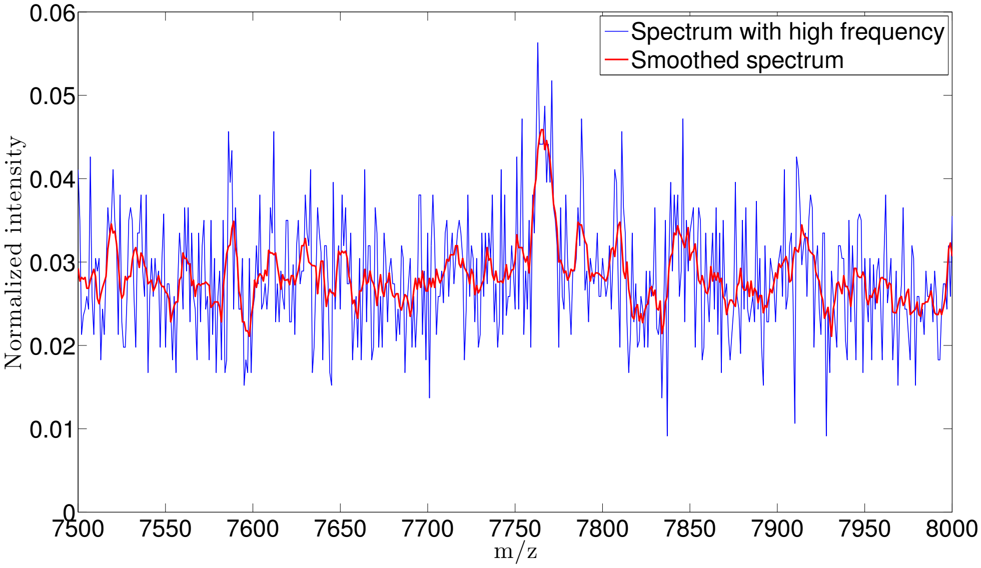


Figure 2.4: High-frequency noise of mass spectrum

in Figure 2.2. Baseline removal is one important step in pre-processing as well (see Section 1.2.1).

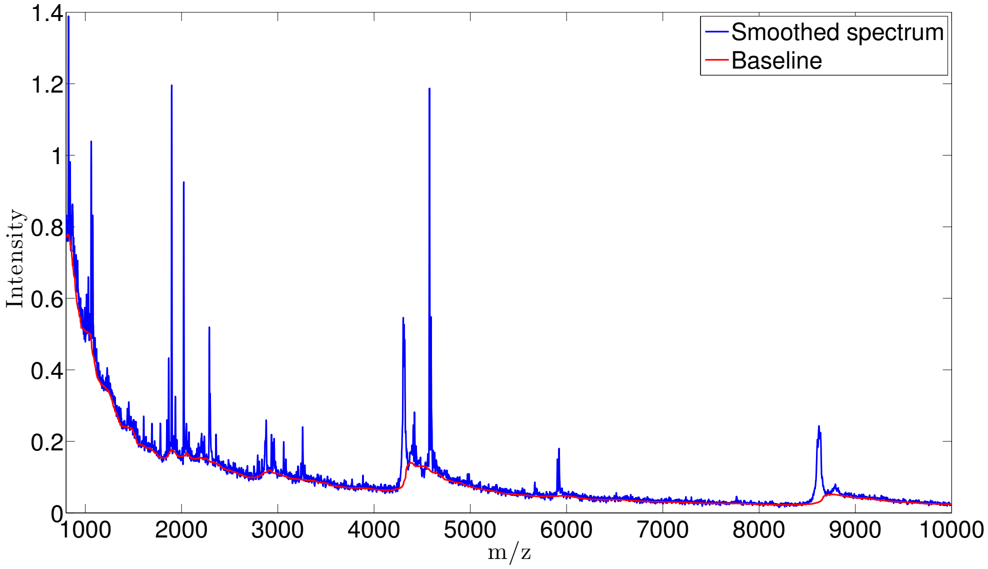


Figure 2.5: Baseline of mass spectrum

Horizontal perturbations of mass spectra abscissas are related to the accuracy of a mass spectrometer machine, which affects the m/z value of each point. Namely, at the time a mass spectrometer detects a peptide particle, the m/z value recorded has potential deviation from the true m/z value of this particle. The deviation is quantified by the mass spectrometer accuracy ρ, which is defined as

$$\rho = \max_{x_0} \frac{|x - x_0|}{x_0},$$

where x is the reported m/z value and x_0 is the true m/z value. ρ is usually provided by the manufacture of mass spectrometer as a fixed parameter, which is normally between 0.1% and 0.3%. Therefore, there is a roughly error window $x \pm \rho x$ around an observed m/z value x where the true m/z value lies. Figure 2.6 displays, as an example, a error window of a point with a m/z value 4370.2. Because of such variation, a same peptide particle can be detected with two different m/z ratios in two mass spectra. This makes the direct comparison between mass spectra impossible without peak realignment. Peak alignment is often implemented by commercial software (see Section 1.2.1).

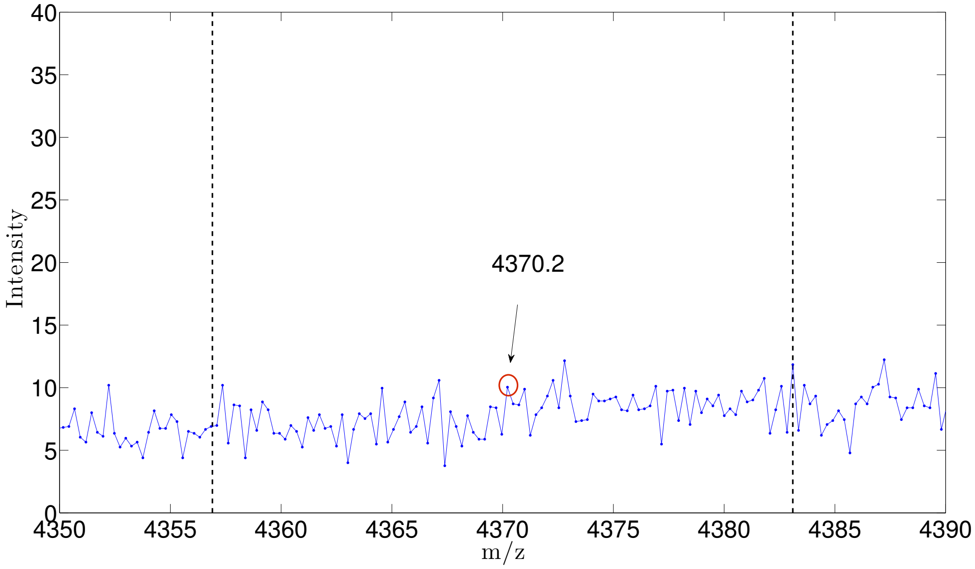


Figure 2.6: Error window for a m/z value

The acquisition of mass spectra is blurred by variations introduced in successions of distinct experimental phases. Two mass spectra from one biological sample, called replicates, can be different as shown in Figure 2.7, even though they are produced by repeating the same experimental process. However, we can still observe that there are obvious common peak patterns between the two replicates. A mass spectrometry analysis approach that utilizes two or more replicates per subject can achieve a more reliable result (see Section 7.4).

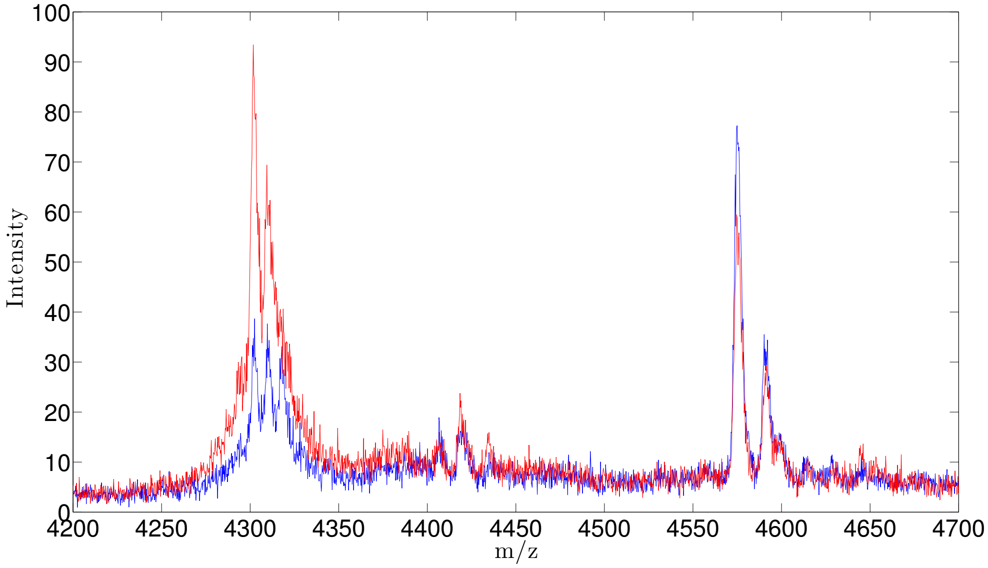


Figure 2.7: Two mass spectrum replicates

2.3 Cancer Discrimination Tasks

Cancer detection in our study is not only about discrimination between cancerous patients and healthy patients, but also involves discrimination among different cancer stages.

Suppose we have gathered a group of biological samples from n_1 patients with a certain cancer and a control group of biological samples from n_2 healthy patients. (The two groups of individuals may also be patients with early stage of cancer versus late stage of cancer in the case of discrimination between different cancer stages.) After acquiring one mass spectrum from each of patient protein samples through the same process using either MALDI-TOF or SELDI-TOF technique, we have two groups of mass spectra G^+ and G^-. Generally speaking, a cancer detection problem is to discover common peak patterns of mass spectra in one group and to select distinct peak patterns characterizing the two groups to construct an accurate and robust classifier to differentiate G^+ from G^-.

Chapter 3

Mass Spectra Pre-processing

Mass spectra can be notoriously complex, not only because of its large scale, but also because several types of experimental noise and errors can complicate its expressions. Data pre-processing, which ”cleans up” raw mass spectra, can increase specificity and sensitivity of protein recognition [36] as well as tease out redundant information to save classification cost. Pre-processing principles are well known, but implementation details vary considerably, and are often not accessible in commercial software. For better context control, we have developed our own sequence of pipelined pre-processing steps. Peaks detected after the pre-processing procedure can not be used directly to compare spectra because of vertical and horizontal variations they have. To take account of the abscissa dependent noise affecting peaks intensities, our algorithm designs a method to evaluate the reliability of the detected peaks; to reduce the effect of horizontal variations, we condense all peaks to a list of consensus reference peak peaks.

Note: This section is based on previous work of Dr. Robert Azencott and Dr. Chinmaya Gupta.

3.1 Stepwise Pre-processing

A point in a mass spectrum is represented by a coordinate (x, y), with x as m/z ratios and y as intensity in Dalton. In our algorithm, we implement the same sequence of pre-processing steps for each mass spectrum.

1. Generated by high resolution spectrometers, m/z ratios are floating points values. To increase process speed, we calibrate each m/z ratio to its nearest integers x; we then compute the median of all the intensities whose m/z ratios binned to x as the new intensity value y. A mass spectrum is then defined on points

$$(x_1, y_1), (x_2, y_2), \cdots (x_n, y_n).$$

2. Normalization is classically performed with respect to the total ion current (TIC), i.e., the sum of all the intensities in each mass spectrum, so that the total intensity in a mass spectrum is 1.

3. Noise-affecting intensities are smoothed out by a moving average technique, namely, for each point (x_i, y_i), we replace y_i by the average value y_i' of the intensities of all the points that have abscissas within a window $[x - ux, x - ux]$. We use this type of sliding window because the noise linearly amplifies with

the increase of m/z values. This gives us a smoothed mass spectrum defined on

$$(x_1, y_1'), (x_2, y_2'), \cdots (x_n, y_n').$$

Noise at (x_i, y_i') is computed as $ns_i = y_i - y_i'$.

4. The local noise values of (x_i, y_i') detected within a local noise window $[x_i - vx_i, x_i + vx_i]$ are truncated to eliminate the noise values below their 2.5% empirical quantile or above their 97.5% empirical quantile. Then the empirical standard deviation σ_i of these mildly censored local noise at x_i can be evaluated.

5. Baseline b_i at (x_i, y_i') is computed as $b_i = \max(0, bb_i)$ where $bb_i =$ median of $\{y_i' : i \in I, y_i' \leq 50\%(\max_{k \in I} y_k')\}$ and $I = \{j : x_j \in [x_i - wx_i, x_i + wx_i]\}$. A baseline removed mass spectrum is then defined on

   $$(x_1, \tilde{y}_1), (x_2, \tilde{y}_2), \cdots, (x_n, \tilde{y}_n),$$

   where $\tilde{y}_i = y_i' - b_i$.

6. Peak strength S_i at $(x_i, \tilde{y}_i)$ is defined as $S_i = \tilde{y}_i / \sigma_i$. A point $(x_i, \tilde{y}_i)$ is detected as a "peak" if it is a local maximum within the window $[x_i - tx_i, x_i + tx_i]$ and S_i is larger than a user-defined threshold th.

3.2 Reliability Scoring for Peaks

Peaks form the basis of discovered biomarkers and signatures. Peak intensities can be strongly impacted by noise during each experiment, so it is quite important to isolate not only the most discriminating peaks, but also the peaks that discriminate with the highest reliability across multiple experiment. To date, few studies have attempted to explore the noise characterization. We propose an innovative algorithm that identifies the distribution of such underlying noise and computes "peak reliability scores" which characterize the peak capacity to persist through random noise perturbation. Peak reliability score provides a way to evaluate the reliability of a peak across repeated experiments and also the foundation to compute the reliability score of signatures, which will be described in Section 7.

Suppose we have extracted n peaks $P_1, P_2, \ldots, P_n$ from mass spectrum M with intensities $\tilde{y}_1, \tilde{y}_2, \cdots, \tilde{y}_n$ after pre-processing. We first compute the histogram $\mathcal{H}_i$ of the local noise of each peak P_i. Then we repeat the following process K (eg. $K = 100$) times. At $K = k$, we

1. Sample a noise rns_i by Monte-Carlo simulation from $\mathcal{H}_i$, and then add it to $\tilde{y}_i$ to obtain a new intensity $\tilde{yy}_i = \tilde{y}_i + rns_i$;

2. Compute new peak strengths $SS_i = \tilde{yy}_i/\sigma_i (i = 1, 2, \cdots, n)$;

3. Rank $SS_i (i = 1, 2, \cdots, n)$ in ascending order; let r_i^k = percentage of ranking of SS_i, $0 < r_i^k < 1$. The larger the rank, the larger the peak strength.

A reliability score $\bar{r}_i$ of P_i is computed as

$$\bar{r}_i = \frac{1}{K} \sum_{k=1}^{K} r_i^k.$$

Let the random varaibles of the noise of the n peaks $P_1, P_2, \cdots, P_n$ be NS_1, NS_2, $\cdots$, NS_n with probability distributions $Q_1, Q_2, \cdots, Q_n$. Let $S_1, S_2, \cdots, S_n$ be their original strengths and also assume that $S_1 < S_2 < \cdots < S_n$. It is easy to see that the relative order of two peaks P_i and P_j $(i < j)$ is almost not affected by perturbation if and only if

$$P\{(\tilde{y}_j + NS_j)/\sigma_j < (\tilde{y}_i + NS_i)/\sigma_i\} \sim 1,$$

that is

$$P\{S_j - S_i < NS_i/\sigma_i - NS_j/\sigma_j\} \sim 1.$$

Figure 3.2 compares the histograms of NS_i/σ_i and NS_j/σ_j identified from two peaks P_i and P_j ($x_i = 8606$ and $x_j = 8937$) in the spectrum of Figure 3.1. The difference of strengths $S_j - S_i$ between the two peaks is about 100 while the scale of NS_i/σ_i or NS_j/σ_j is about 5. Therefore, noise perturbations cannot affect the relative order of the two peaks.

As to the two peaks in Figure 3.3 with their strength noise histograms showed in Figure 3.4, we can compute empirically

$$P\{NS_i/\sigma_i - NS_j/\sigma_j < S_j - S_i\} = P\{NS_i/\sigma_i - NS_j/\sigma_j < 0.49\} \approx 27.31\%,$$

which indicates that the relative order of the two peaks according to their strength S_i and S_j have a 27.31% chance to be incorrect. It can be seen from Figure 3.3 that both of these peaks have comparable large local noise levels with respect to their intensities, which highly affect their relative rank order after perturbation.

Peaks that are always top ranked across multiple noise perturbations are reliable peaks. Table 3.1 shows, as an example, the raw ranks, which are the ranks by

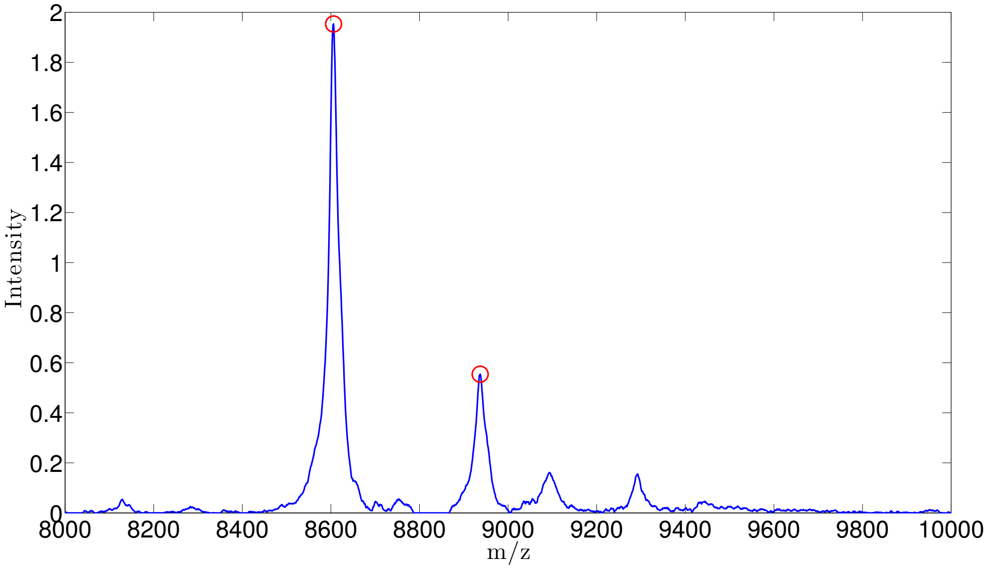


Figure 3.1: Two reliable peaks with stable relative order
Noise perturbations cannot affect the relative order of the two circled peaks.

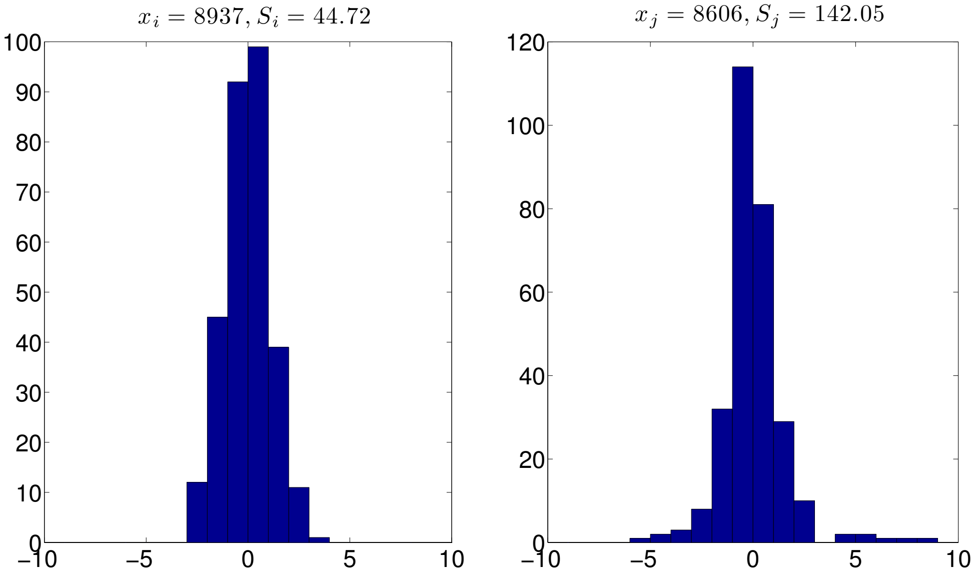


Figure 3.2: Histograms of NS_i/σ_i and NS_j/σ_j of P_i and P_j with stable relative order $S_j - S_i$ is about 100 while NS_i/σ_i and NS_j/σ_j can only reach 5. Therefore, noise perturbation will almost surely not affect their relative order.

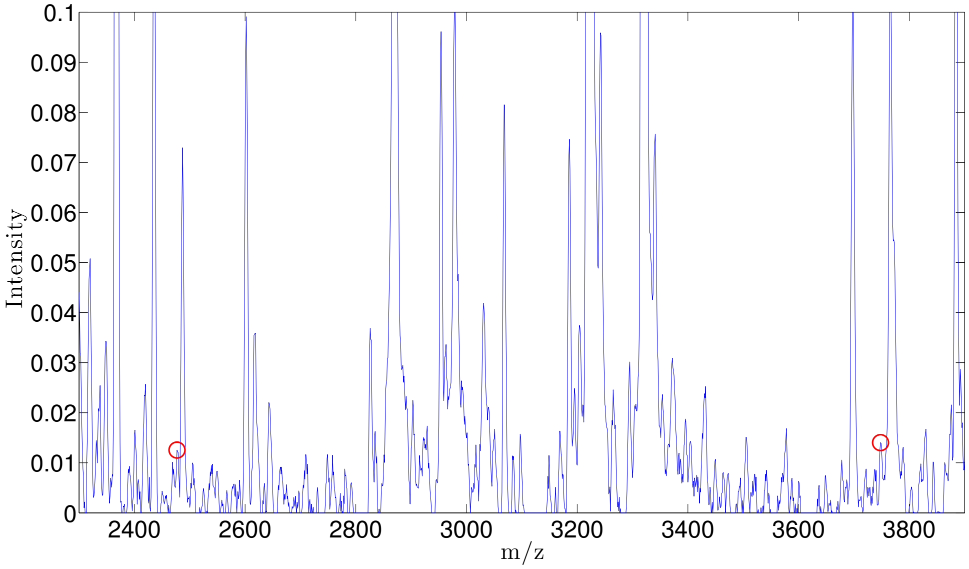


Figure 3.3: Two reliable peaks with unstable relative order
Noise perturbations can easily affect the relative order of the two circled peaks.

increasing peak strengths S_i, and reliability scores $\bar{r}_i$ of several peaks from of the spectrum in Figure 2.2. Some peaks have different ranks before and after perturbation by noise, some don't. Since noise to σ (σ is the standard deviation of noise) are usually within a moderate range with probability 1, peaks which are at a low strength level, are comparably unreliable in their raw ranks. Therefore, this reliability score technique is more helpful in filtering out reliable peaks among those with moderate strengths.

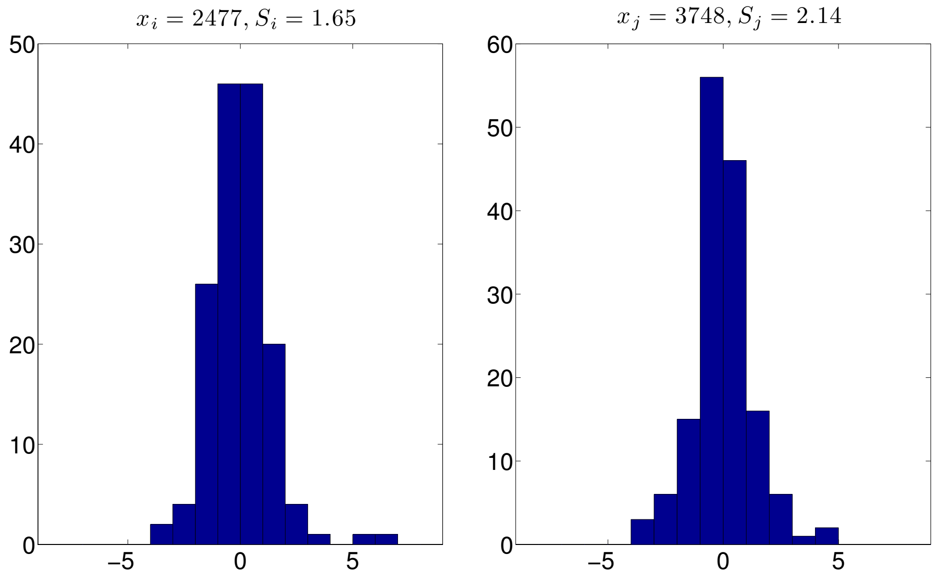


Figure 3.4: Histograms of NS_i/σ_i and NS_j/σ_j of P_i and P_j with unstable relative order

$S_j - S_i$ is less than 1 while NS_i/σ_i and NS_j/σ_j can reach 5. Therefore, their relative order can be easily affected by noise perturbation.

x_i	S_i	Raw rank	$\bar{r}_i$
8606	142.05	100%	100%
3226	38.98	97%	97%
6680	9.51	86%	86%
6417	4.72	75%	84%
1311	4.37	72%	85%
1562	2.23	35%	22%

Table 3.1: Raw ranks and reliability scores

This table compares the raw ranks (rank by increasing S_i) and reliability scores of several peaks from the spectrum in Figure 2.2. Reliability score is computed by taking account of noise perturbations, therefore can be quite different from raw rank, especially for peaks with middle level ranks.

3.3 Reference Peak Abscissas and Activation Frequencies

In repeated mass spectra acquisitions with the same spectrometer, m/z ratios corresponding to the same protein exhibit random small shifts around a true value, which generates difficulty in biomarker selection and comparison across all mass spectra. When the manufacturer accuracy of a MALDI or SELDI spectrometer is ρ, acquisition shifts around a true m/z ratio x will be within the range $x \pm \rho x$ (see Section 2.2). Therefore, the m/z values obtained after data pre-processing can not be directly used to compare spectra. To take account of these "error windows", we first construct a list of "reference peak abscissas" $Ab_1, Ab_2, \cdots, Ab_n$ at $a(1+\rho), a(1+\rho)^2, a(1+\rho)^3, \cdots, a(1+\rho)^n \leq b$, where a and b are the smallest and largest m/z observed among all spectra. This procedure will dramatically reduce the size of our feature space and provide a list of common m/z values over all mass spectra to facilitate spectrum-wise comparison and efficient classification.

Throughout the following chapters, we will use the following definitions.

Definition 3.3.1. For any reference peak abscissa Ab_j, and any mass spectrum M, we say that Ab_j is **activated** by M if there is at least one detected peak (see Section 3.1) in M positioned within the uncertainty window of Ab_j.

We define the **activation frequency** of a reference peak abscissa Ab_j within a group G by

$$fq_G(Ab_j) = \frac{n(Ab_j)}{N},$$

where $n(Ab_j)$ is the number of mass spectra of G which activate Ab_j, and N is the number of mass spectra in G.

We also define similarly the "activation frequency" of a pair of reference peak abscissas.

Definition 3.3.2. A pair of distinct reference peak abscissas (Ab_i, Ab_j) is said to be **activated** by M if both Ab_i and Ab_j are activated by M.

The **activation frequency** of a pair of abscissas (Ab_i, Ab_j) within a group G of mass spectra is defined as

$$fq_G(Ab_i, Ab_j) = \frac{n(Ab_i, Ab_j)}{N},$$

where $n(Ab_i, Ab_j)$ is the number of mass spectra of G which activate both Ab_i, Ab_j, and N is the number of mass spectra in G.

Chapter 4

Gibbs Distributions

Given the heterogeneity and complexity of cancer types, biomarkers generally have higher discriminating power combinatorially than individually [99]. The simultaneous use of a small subset of markers, has been suggested to improve sensitivity and specificity of cancer diagnosis [98], yet, the underlying co-occurence relationship between biomarkers are often only superficially investigated. In this chapter, we introduce a broad class of stochastic models, namely the Markov Random Field (MRF) along with its probability distribution, Gibbs distribution, to model groups of binary coded mass spectra to search for small sets of biomarkers in the context of algorithmic discrimination between cancer stages, or between cancer and control groups of patients. Generalizing from classical methodologies, three Gibbs parameters estimation methods, MLE, MPLE and MFE are elaborated. After fitting Gibbs models to given groups of mass spectra, we will be able to simulate large samples of virtual data, which, as will be seen in Chapter 8 and 9, is very useful to study the asymptotic performance of discrimination algorithms.

4.1 Introduction

Gibbs distributions were first introduced in the context of physics, and have been successfully used to model spatial dependencies for interacting systems distributed on lattices. Throughout this section, the notation and symbols in [19] are used.

Let S be a finite set with elements denoted by s, called sites. Let Λ be a finite set called phase space. A random field on S with phases in Λ is defined as a random variable $\boldsymbol{X} = \{\boldsymbol{X}(s)\}_{s \in S}$ taking values in the configuration space Λ^S.

Definition 4.1.1. A **neighborhood system** on S is a family $N = \{\mathcal{N}_s\}_{s \in S}$ of subsets in S such that: for all $s \in S$, (i) $s \notin \mathcal{N}_s$, (ii) $t \in \mathcal{N}_s \Rightarrow s \in \mathcal{N}_t$. The subset $\mathcal{N}_s$ is called the **neighborhood** of site s and the couple (S, N) is called a **graph**. A **clique** C of the graph (S, N) is a subset of S, for which any two distinct sites are mutual neighbors. Any singleton $\{s\}$ is also considered a clique.

Definition 4.1.2. The random field Λ^S is called a **Markov random field (MRF)** with respect to the neighborhood system N if for all sites $s \in S$

$$P\{\boldsymbol{X}(s) = \boldsymbol{x}(s) \mid \boldsymbol{X}(S \backslash s) = \boldsymbol{x}(S \backslash s)\} = P\{\boldsymbol{X}(s) = \boldsymbol{x}(s) \mid \boldsymbol{X}(\mathcal{N}_s) = \boldsymbol{x}(\mathcal{N}_s)\}.$$

Consider the probability distribution

$$\pi_T(\boldsymbol{x}) = \frac{1}{Z_T} e^{-\frac{1}{T}\mathcal{E}(\boldsymbol{x})} \tag{4.1}$$

on the configuration space Λ^S, where $T > 0$ is the temperature, $-\infty < \mathcal{E}(\boldsymbol{x}) < \infty$ is the energy of configuration $\boldsymbol{x}$, and $Z_T = \sum_{\boldsymbol{x}\in\Lambda^S} \pi_T(\boldsymbol{x})$ is called partition function.

Definition 4.1.3. A **Gibbs potential** on Λ^S relative to the neighborhood system N is a collection $\{V_C\}_{C\subset S}$ of functions $V_C : \Lambda^S \to \mathbb{R} \cup +\infty$.

A probability distribution (4.1) is called a **Gibbs distribution** if its energy function $\mathcal{E}$ is derived from the Gibbs potential $\{V_C\}_{C\subset S}$, i.e.,

$$\mathcal{E}(\boldsymbol{x}) = \sum_{C} V_C(\boldsymbol{x}). \tag{4.2}$$

Definition 4.1.4. The **local specification** of a Gibbs distribution is a family $\{\pi^s\}_{s\in S}$ with $\pi^s : \Lambda^S \to [0,\ 1]$ such that

$$\pi^s(\boldsymbol{x}) = P\{\boldsymbol{X}(s) = \boldsymbol{x}(s)|\boldsymbol{X}(\mathcal{N}_s) = \boldsymbol{x}(\mathcal{N}_s)\}. \tag{4.3}$$

Here we focus on binary phase space $\Lambda = \{0, 1\}$. There are multiple classes of classically studied potentials. We restrict our discussion here to a specific class – often called the autologistic model.

The autologistic model involves cliques of cardinality 1 and 2 and defines distribution (4.1) by

$$\pi_{\boldsymbol{\theta}}(\boldsymbol{x}) = \frac{1}{Z} e^{-\mathcal{E}(\boldsymbol{x})}, \quad \mathcal{E}(\boldsymbol{x}) = \sum_{s\in S} \theta_s \boldsymbol{x}(s) - \sum_{\{s,t\}\in\mathcal{C}} \theta_{st} \boldsymbol{x}(s)\boldsymbol{x}(t), \tag{4.4}$$

where $x(s) \in \Lambda = \{0, 1\}$, for all $s \in S$, $\mathcal{C}$ is the set of all distinct cliques of cardinality 2. We consider $\{s, t\}$ and $\{t, s\}$ as same clique. $\boldsymbol{\theta} = (\cdots\theta_s\cdots\theta_{st}\cdots)^*$ is a parameter, where $\theta_s, \theta_{st} \in \mathbb{R}$. The temperature T in (4.1) has been fixed at 1 and the normalizing constant is

$$Z = \sum_{\boldsymbol{x} \in \Lambda^S} e^{-\mathcal{E}(\boldsymbol{x})}.$$

From now on, we denote $\boldsymbol{x}(s)$ by x_s for simplicity.

4.2 Parameter Estimation

Two classical parameter estimation methods for Gibbs distributions are the maximum likelihood estimation (MLE) and the maximum pseudolikelihood estimation (MPLE). We also propose a new method – marginal fitting estimation (MFE).

4.2.1 Maximum Likelihood Estimation (MLE)

Gibbs distribution belongs to exponential families. MLE, as a widely used parameter estimation method, has some considerable merits for exponential families: the MLE estimator is known to be unique, consistent, asymptotic normal and most efficient [12] [20] [48] as the sample size tends to infinity.

4.2.1.1 MLE for Exponential Families

Recall that a canonical form of exponential families is

$$f(x \mid \theta) = e^{\theta T(x) - \psi(\theta)) h(x)},$$

where $\theta \in \Theta = \{\theta : \int e^{\theta T(x)} h(x) dx < \infty\}$.

Theorem 4.2.1. [12] [20] [48] *Let $x_1, \dots, x_n$ are i.i.d random variables with common density function $f(x \mid \theta)$ and let the true $\theta = \theta_0 \in \Theta^0$, the interior of Θ. Assume $\psi(\theta)$ is twice differentiable with respect to θ and that $\psi''(\theta) > 0 \ \forall \ \theta \in \Theta^0$. Then,*

1. for large n, a unique MLE $\hat{\theta}_n$ of θ exists, which satisfies

$$\mathrm{E}_{\hat{\theta}_\mathrm{n}}\mathrm{T(x)} = \frac{1}{\mathrm{n}}(\mathrm{T(x_1)} + \ldots + \mathrm{T(x_n)});$$

2. $\hat{\theta}_n$ is consistent, i.e.,

$$\hat{\theta}_n \xrightarrow[n\to\infty]{} \theta_0$$

in probability, where the underlying probability distribution has density $f(x \mid \theta_0)$;

3. $\hat{\theta}_n$ is asymptotically normal and most efficient, i.e.,

$$\sqrt{n}(\hat{\theta}_n - \theta_0) \xrightarrow[n\to\infty]{} N(0, I^{-1}(\theta_0))$$

in distribution, where $I(\theta_0) = \psi''(\theta_0)$.

4.2.1.2 MLE of Gibbs Distribution

The autologistic model (4.4) can be displayed in vector form

$$\pi(\boldsymbol{x}) = \frac{1}{Z}e^{-\boldsymbol{\theta}^*\boldsymbol{U}(\boldsymbol{x})}, \tag{4.5}$$

where $\boldsymbol{\theta}$ is defined as above and $\boldsymbol{U}(\boldsymbol{x}) = (\cdots \quad x_s \quad \cdots \ x_s x_t \ \cdots)^*, s \in S, \{s,t\} \in \mathcal{C}$.

Suppose we have n observations $\mathcal{D}_n = \{\boldsymbol{y}^{(1)}, \boldsymbol{y}^{(2)}, \cdots, \boldsymbol{y}^{(n)}\}$, $\boldsymbol{y}^{(i)} \in \{0,1\}^S$, then the likelihood function is defined as $L(\boldsymbol{\theta}, \mathcal{D}_n) = \prod_{i=1}^n \pi(\boldsymbol{y}^{(i)})$ and the MLE of $\boldsymbol{\theta}$ is

$$\hat{\boldsymbol{\theta}} = \arg\max_{\boldsymbol{\theta}} \log L(\boldsymbol{\theta}, \mathcal{D}_n), \tag{4.6}$$

Since

$$\log L(\boldsymbol{\theta}, \mathcal{D}_n) = -\sum_{i=1}^{n} \boldsymbol{\theta}^* \boldsymbol{U}(\boldsymbol{y}^{(i)}) - n \log Z \tag{4.7}$$

and

$$\frac{\partial Z}{\partial \boldsymbol{\theta}} = -\mathrm{E}_{\boldsymbol{\theta}} \boldsymbol{U},$$

the first derivative of $\log L(\boldsymbol{\theta}, \mathcal{D}_n)$ is

$$G_{\boldsymbol{\theta}} = \frac{\partial \log L(\boldsymbol{\theta}, \mathcal{D}_n)}{\partial \boldsymbol{\theta}} = n\mathrm{E}_{\boldsymbol{\theta}} \boldsymbol{U} - \sum_{\mathrm{i=1}}^{\mathrm{n}} \boldsymbol{U}(\boldsymbol{y}^{(\mathrm{i})}), \tag{4.8}$$

and the second derivative is

$$H_{\boldsymbol{\theta}} = \frac{\partial^2 \log L(\boldsymbol{\theta}, \mathcal{D}_n)}{\partial \boldsymbol{\theta}^2} = -n\mathrm{Var}_{\boldsymbol{\theta}} \boldsymbol{U}. \tag{4.9}$$

$H_{\boldsymbol{\theta}}$ is a negative semidefinite matrix, which implies that $\log L(\boldsymbol{\theta}, \mathcal{D}_n)$ is concave. Thus, there exists only one maximum $\hat{\boldsymbol{\theta}}$ of $\log L(\boldsymbol{\theta}, \mathcal{D}_n)$, at which $nE_{\hat{\boldsymbol{\theta}}} \boldsymbol{U} - \sum_{i=1}^n \boldsymbol{U}(\boldsymbol{y}^{(i)}) = 0$.

We can implement gradient descent method to compute the MLE numerically. The convergent sequence $\boldsymbol{\theta}(0), \boldsymbol{\theta}(1), \boldsymbol{\theta}(2), \cdots$ of approximate values of $hat\theta$ is constructed by the gradient iteration

$$\boldsymbol{\theta}(k+1) = \boldsymbol{\theta}(k) + \epsilon G_{\boldsymbol{\theta}(k)} \tag{4.10}$$

starting from a user selected $\theta(0)$ and using a small step size ϵ. One can stop such iterations when $G_{\boldsymbol{\theta}(k)}$ is less than a user-chosen sufficiently small threshold. In (4.8),

$E_{\boldsymbol{\theta}}\boldsymbol{U}$ can be computed empirically from samples constructed by Gibbs sampling. Namely, if $\boldsymbol{x}^{(1)}, \boldsymbol{x}^{(2)}, \cdots, \boldsymbol{x}^{(n)}$ are the simulated samples,

$$E_{\boldsymbol{\theta}}\boldsymbol{U} \approx \frac{1}{n}(\boldsymbol{x}^{(1)} + \boldsymbol{x}^{(2)} + \cdots + \boldsymbol{x}^{(n)}).$$

For any fixed $\boldsymbol{\theta}$, Gibbs sampling proceeds as follows.

1. Select randomly a configuration in Λ^S.

2. for $i = 1, 2, \cdots, n$ (n is the number of sites)

   Create a new configuration $\boldsymbol{y}$ by updating the ith coordinate of the current configuration $\boldsymbol{x}$ (1 is updated to 0 and 0 is updated to 1);

   Select a random number u with uniform distribution on [0,1];

   Replace the current configuration $\boldsymbol{x}$ by $\boldsymbol{y}$ if $u < P_{\boldsymbol{\theta}}\{\boldsymbol{X} = \boldsymbol{y} \mid \boldsymbol{X} = \boldsymbol{x}\}$ ($P_{\boldsymbol{\theta}}\{\boldsymbol{X} = \boldsymbol{y} \mid \boldsymbol{X} = \boldsymbol{x}\}$ is the conditional probability of $\boldsymbol{X} = \boldsymbol{y}$ given $\boldsymbol{X} = \boldsymbol{x}$); otherwise, keep $\boldsymbol{x}$ as the current configuration.

3. Repeat step (2) N times (N is called iteration step).

4. After b ($b < N$) iteration steps (b is called warm-up period) are completed, extract then one configuration after every c iteration steps have been completed (c is called spacing).

This gives us $(N - b)/c$ random configurations having approximately an autologistic distribution (4.5) with parameter $\boldsymbol{\theta}$.

4.2.2 Maximum Pseudolikelihood Estimation (MPLE)

Maximum pseudolikelihood estimation (MPLE), introduced by [13], has played an important role in parameter estimation of spatial models such as (4.4) before the advent of Monte Carlo methods. It is still a generally preferred method because it requires no simulation, leading to fast speed.

The pseudolikelihood function of one observation is defined as the product of all local specifications

$$L(\boldsymbol{\theta}, \boldsymbol{x}) = \prod_s P_{\boldsymbol{\theta}}(x_s | \boldsymbol{x}(\mathcal{N}_s)), \tag{4.11}$$

and the log pseudolikelihood function

$$logL(\boldsymbol{\theta}, \boldsymbol{x}) = -\sum_s x_s(\theta_s + \sum_{t \in \mathcal{N}_s} \theta_{st} x_t) - \sum_s log(1 + e^{-(\theta_s + \sum_{t \in \mathcal{N}_s} \theta_{st} x_t)}). \tag{4.12}$$

To maximize (4.12), we take its first derivative with respective to θ_s and θ_{st} for each s and t:

$$\begin{aligned}
\frac{\partial logL(\boldsymbol{\theta}, \boldsymbol{x})}{\partial \theta_s} &= -x_s + \frac{e^{-(\theta_s + \sum_{t \in \mathcal{N}_s} \theta_{st} x_t)}}{1 + e^{-(\theta_s + \sum_{t \in \mathcal{N}_s} \theta_{st} x_t)}}, \\
\frac{\partial logL(\boldsymbol{\theta}, \boldsymbol{x})}{\partial \theta_{st}} &= -2x_s x_t + \frac{x_t e^{-(\theta_s + \sum_{r \in \mathcal{N}_s} \theta_{sr} x_r)}}{1 + e^{-(\theta_s + \sum_{r \in \mathcal{N}_s} \theta_{sr} x_r)}} + \frac{x_s e^{-(\theta_t + \sum_{r \in \mathcal{N}_t} \theta_{tr} x_r)}}{1 + e^{-(\theta_t + \sum_{r \in \mathcal{N}_t} \theta_{tr} x_r)}},
\end{aligned}$$

and then take its second derivatives:

$$\begin{aligned}
\frac{\partial^2 logL(\boldsymbol{\theta}, \boldsymbol{x})}{\partial \theta_s^2} &= -\frac{e^{-(\theta_s + \sum_{r \in \mathcal{N}_s} \theta_{sr} x_r)}}{(1 + e^{-(\theta_s + \sum_{r \in \mathcal{N}_s} \theta_{sr} x_r)})^2}, \\
\frac{\partial^2 logL(\boldsymbol{\theta}, \boldsymbol{x})}{\partial \theta_{st}^2} &= -\frac{x_s^2 e^{-(\theta_t + \sum_{r \in \mathcal{N}_t} \theta_{tr} x_r)}}{(1 + e^{-(\theta_t + \sum_{r \in \mathcal{N}_t} \theta_{tr} x_r)})^2} - \frac{x_t^2 e^{-(\theta_s + \sum_{k \in \mathcal{N}_s} \theta_{sr} x_r)}}{(1 + e^{-(\theta_s + \sum_{r \in \mathcal{N}_s} \theta_{ik} x_k)})^2}, \\
\frac{\partial^2 logL(\boldsymbol{\theta}, \boldsymbol{x})}{\partial \theta_s \partial \theta_t} &= 0, \\
\frac{\partial^2 logL(\boldsymbol{\theta}, \boldsymbol{x})}{\partial \theta_s \partial \theta_{st}} &= -\frac{x_t e^{-(\theta_s + \sum_{k \in \mathcal{N}_s} \theta_{sr} x_r)}}{(1 + e^{-(\theta_s + \sum_{k \in \mathcal{N}_s} \theta_{sr} x_r)})^2}.
\end{aligned}$$

Suppose the size of S and $\mathcal{C}$ are l_1 and l_2 respectively. Let $\bar{X} = \begin{pmatrix} I \\ J \end{pmatrix}$ where I is the $l_1 * l_1$ identity matrix corresponding to the first l_1 parameters, and J is a sparse $l_2 * l_1$ matrix in the form of

$$J = \begin{array}{cc} & \begin{array}{ccccc} \phantom{\mathbf{0}} & \text{column } s & \phantom{\mathbf{0}} & \text{column } t & \phantom{\mathbf{0}} \end{array} \\ \text{row } \theta_{st} & \begin{pmatrix} \mathbf{0} & 0 & \mathbf{0} & 0 & \mathbf{0} \\ \mathbf{0} & x_t & \mathbf{0} & x_s & \mathbf{0} \\ \mathbf{0} & 0 & \mathbf{0} & 0 & \mathbf{0} \end{pmatrix} \end{array}.$$

Each line of J corresponds to a parameter θ_{st} for some s and t. In addition, let

$$\Omega_{\boldsymbol{\theta}} = \begin{pmatrix} \bar{\pi}^1_{\boldsymbol{\theta}}(\boldsymbol{x}) - x_1 \\ \vdots \\ \bar{\pi}^{l_1}_{\boldsymbol{\theta}}(\boldsymbol{x}) - x_{l_1} \end{pmatrix},$$

and

$$\Gamma_{\boldsymbol{\theta}} = diag\{\bar{\pi}^1_{\boldsymbol{\theta}}(1 - \bar{\pi}^1_{\boldsymbol{\theta}}), ..., \bar{\pi}^{l_1}_{\boldsymbol{\theta}}(1 - \bar{\pi}^{l_1}_{\boldsymbol{\theta}})\},$$

where

$$\bar{\pi}^s_{\boldsymbol{\theta}}(\boldsymbol{x}) = P_{\boldsymbol{\theta}}(x_s = 1 \mid \boldsymbol{x}(\mathcal{N}_s)) = \frac{e^{-(\theta_s + \sum_{r \in \mathcal{N}_s} \theta_{sr} x_r)}}{1 + e^{-(\theta_s + \sum_{r \in \mathcal{N}_s} \theta_{sr} x_r)}}, s = 1, 2, ..., l_1,$$

the gradient $G_{\boldsymbol{\theta}}$ and second derivative Hessian matrix of $\log L(\boldsymbol{\theta}, \boldsymbol{x})$ can be represented in vector forms

$$G_{\boldsymbol{\theta}} = \bar{X}\Omega_{\boldsymbol{\theta}}, \tag{4.13}$$

$$H_{\boldsymbol{\theta}} = -\bar{X}\Gamma_{\boldsymbol{\theta}}\bar{X}^*. \tag{4.14}$$

With n observations $\mathcal{D}_n = \{\boldsymbol{y}^{(1)}, \boldsymbol{y}^{(2)}, \cdots, \boldsymbol{y}^{(n)}\}$, (4.12) becomes

$$\sum_n \log L(\boldsymbol{\theta}, \boldsymbol{y}^{(n)}) = \sum_n \sum_s \log P_{\boldsymbol{\theta}}(\boldsymbol{y}^{(n)}(s) \mid \boldsymbol{y}^{(n)}(\mathcal{N}_s)). \tag{4.15}$$

The gradient and Hessian matrix of the new loglikelihood function are

$$G_{\boldsymbol{\theta}} = \sum_n G_{\boldsymbol{\theta}}^{(n)} = \sum_n \bar{X}^{(n)} \Omega_{\boldsymbol{\theta}}^{(n)}, \tag{4.16}$$

$$H_{\boldsymbol{\theta}} = \sum_n H_{\boldsymbol{\theta}}^{(n)} = -\sum_n \bar{X}^{(n)} \Gamma_{\boldsymbol{\theta}}^{(n)} \bar{X}^{(n)^*}, \tag{4.17}$$

where the superscript n of a matrix denotes its value evaluated at observation $\boldsymbol{y}^{(n)}$.

We can utilize the gradient descent method (4.10) or the Newton-Raphson descent method

$$\boldsymbol{\theta}(k+1) = \boldsymbol{\theta}(k) + H_{\boldsymbol{\theta}(k)}^{-1} G_{\boldsymbol{\theta}(k)} \tag{4.18}$$

to estimate the maximum $\hat{\boldsymbol{\theta}}$ of (4.12). One can stop iterating when $G_{\boldsymbol{\theta}(k)}$ is less than a user-chosen sufficiently small threshold.

4.2.3 Marginal Fitting Estimation (MFE)

We propose a new method to estimate parameter by fitting on marginal distributions. As before, we still assume that there are n observations $\mathcal{D}_n = \{\boldsymbol{y}^{(1)}, \boldsymbol{y}^{(2)}, \cdots, \boldsymbol{y}^{(n)}\}$. We compute the empirical marginal probabilities (dimension 1 and dimension 2) of each site s and each pair of sites $\{s, t\}$ by

$$q_s^{\delta}(\mathcal{D}_n) = \frac{\#\{\boldsymbol{y} \in \mathcal{D}_n : y_s = \delta\}}{n}, \; q_{st}^{\delta_1\delta_2}(\mathcal{D}_n) = \frac{\#\{\boldsymbol{y} \in \mathcal{D}_n : y_s = \delta_1, y_t = \delta_2\}}{n},$$

where $\delta \in \{0,1\}$, $\{\delta_1, \delta_2\} \in \{0,1\}^2$. Let

$$p_s^\delta(\boldsymbol{\theta}, \mathcal{D}_n) = \sum_{\boldsymbol{x} \in A(\mathcal{D}_n; s, \delta)} \frac{1}{Z} e^{-\boldsymbol{\theta}^* \boldsymbol{U}(\boldsymbol{x})}, \ p_{st}^{\delta_1 \delta_2}(\boldsymbol{\theta}, \mathcal{D}_n) = \sum_{\boldsymbol{x} \in B(\mathcal{D}_n; s, t, \delta_1, \delta_2)} \frac{1}{Z} e^{-\boldsymbol{\theta}^* \boldsymbol{U}(\boldsymbol{x})},$$

where $A(\mathcal{D}_n; s, \delta) = \{\boldsymbol{y} \in \mathcal{D}_n : y_s = \delta\}$ and $B(\mathcal{D}_n; s, t, \delta_1, \delta_2) = \{\boldsymbol{y} \in \mathcal{D}_n : y_s = \delta_1, y_t = \delta_2\}$. p_s^δ and $p_{st}^{\delta_1 \delta_2}$ are marginal probabilities of dimension 1 and dimension 2 computed from model (4.5) with estimated parameter $\boldsymbol{\theta}$ (we call them parameterized marginal probabilities).

The objective function of MFE is the sum of square errors between empirical probabilities and parameterized probabilities, which is defined as

$$\begin{aligned} SSE(\boldsymbol{\theta}, \mathcal{D}_n) = & \sum_s \sum_\delta (p_s^\delta(\boldsymbol{\theta}, \mathcal{D}_n) - q_s^\delta(\mathcal{D}_n))^2 \\ & + \sum_{\{s,t\} \in \mathcal{C}} \sum_{\{\delta_1, \delta_2\} \in \{0,1\}^2} (p_{st}^{\delta_1 \delta_2}(\boldsymbol{\theta}, \mathcal{D}_n) - q_{st}^{\delta_1 \delta_2}(\mathcal{D}_n))^2 \\ & = 2 \sum_s (p_s^0(\boldsymbol{\theta}, \mathcal{D}_n) - q_s^0(\mathcal{D}_n))^2 + \sum_{\{s,t\} \in \mathcal{C}} \sum_{\{\delta_1, \delta_2\} \in \{0,1\}^2} (p_{st}^{\delta_1 \delta_2}(\boldsymbol{\theta}, \mathcal{D}_n) - q_{st}^{\delta_1 \delta_2}(\mathcal{D}_n))^2. \end{aligned} \tag{4.19}$$

The parameter is estimated by minimizing the error $SSE(\boldsymbol{\theta}, \mathcal{D}_n)$. We take the first derivative of $SSE(\boldsymbol{\theta}, \mathcal{D}_n)$ with respect to $\boldsymbol{\theta}$

$$\begin{aligned} G_{\boldsymbol{\theta}} = \frac{\partial SSE(\boldsymbol{\theta}, \mathcal{D}_n)}{\partial \boldsymbol{\theta}} = & 4 \sum_s (p_s^0(\boldsymbol{\theta}, \mathcal{D}_n) - q_s^0(\mathcal{D}_n)) \frac{\partial p_s^0(\boldsymbol{\theta}, \mathcal{D}_n)}{\partial \boldsymbol{\theta}} \\ & + 2 \sum_{s,t} \sum_{\delta_1, \delta_2} (p_{st}^{\delta_1 \delta_2}(\boldsymbol{\theta}, \mathcal{D}_n) - q_{st}^{\delta_1 \delta_2}(\mathcal{D}_n)) \frac{\partial p_{st}^{\delta_1 \delta_2}(\boldsymbol{\theta}, \mathcal{D}_n)}{\partial \boldsymbol{\theta}}. \end{aligned} \tag{4.20}$$

Since

$$\frac{\partial p_s^0(\boldsymbol{\theta}, \mathcal{D}_n)}{\partial \boldsymbol{\theta}} = p_s^0(\boldsymbol{\theta}, \mathcal{D}_n)(\mathrm{E}_{\boldsymbol{\theta}} \boldsymbol{U} - \mathrm{E}_{\boldsymbol{\theta}}(\boldsymbol{U} \mid \mathrm{x_s} = 0)), \tag{4.21}$$

$$\frac{\partial p_{st}^{\delta_1 \delta_2}(\boldsymbol{\theta}, \mathcal{D}_n)}{\partial \boldsymbol{\theta}} = p_{st}^{\delta_1 \delta_2}(\boldsymbol{\theta}, \mathcal{D}_n)(\mathrm{E}_{\boldsymbol{\theta}} \boldsymbol{U} - \mathrm{E}_{\boldsymbol{\theta}}(\boldsymbol{U} \mid \mathrm{x_s} = \delta_1, \mathrm{x_t} = \delta_2)), \tag{4.22}$$

plugging (4.21) and (4.22) into (4.20) and letting $p_s^0(\boldsymbol{\theta}, \mathcal{D}_n)T_s^0$ and $p_{st}^{\delta_1\delta_2}(\boldsymbol{\theta}, \mathcal{D}_n)T_{st}^{\delta_1\delta_2}$ be the right side of (4.21) and (4.22), we have

$$
\begin{aligned}
G_{\boldsymbol{\theta}} =& 4\sum_s p_s^0(\boldsymbol{\theta}, \mathcal{D}_n)(p_s^0(\boldsymbol{\theta}, \mathcal{D}_n) - q_s^0(\mathcal{D}_n))T_s^0 \\
&+ 2\sum_{s,t}\sum_{\delta_1,\delta_2} p_{st}^{\delta_1\delta_2(\boldsymbol{\theta}, \mathcal{D}_n)}(p_{st}^{\delta_1\delta_2}(\boldsymbol{\theta}, \mathcal{D}_n) - q_{st}^{\delta_1\delta_2}(\mathcal{D}_n))T_{st}^{\delta_1\delta_2}.
\end{aligned}
$$

The MFE estimator $\hat{\boldsymbol{\theta}}$ is numerically computed by gradient descent method (4.10), to find $\hat{\theta}$ such that $G_{\hat{\boldsymbol{\theta}}}$ is approximately 0. At any $\boldsymbol{\theta}(k)$ during the iterations, p_s^0, $p_{st}^{\delta_1\delta_2}$, $\mathrm{E}_{\boldsymbol{\theta}}\boldsymbol{U}$, $\mathrm{E}_{\boldsymbol{\theta}}(\boldsymbol{U} \mid \mathrm{x_s} = 0)$, $\mathrm{E}_{\boldsymbol{\theta}}(\boldsymbol{U} \mid \mathrm{x_s} = \delta_1, \mathrm{x_t} = \delta_2)$, $s, t = 1, ..., l_1, \delta_1, \delta_2 = 0, 1$ are computed empirically from the samples simulated by Gibbs sampling. For example, if $V = \{\boldsymbol{x}^{(1)}, \boldsymbol{x}^{(2)}, \cdots, \boldsymbol{x}^{(n)}\}$ is the simulated sample set,

$$
\begin{aligned}
&p_s^0 \approx \frac{1}{n}\#\{\boldsymbol{x} \in V : x_s = 0\}, \\
&\mathrm{E}_{\boldsymbol{\theta}}\boldsymbol{U} \approx \frac{1}{\mathrm{n}}(\boldsymbol{x}^{(1)} + \boldsymbol{x}^{(2)} + \cdots + \boldsymbol{x}^{(\mathrm{n})}), \\
&\mathrm{E}_{\boldsymbol{\theta}}(\boldsymbol{U} \mid \mathrm{x_s} = 0) \approx \frac{\sum_{\boldsymbol{x}\in \mathrm{V}, \mathrm{x_s}=0} \boldsymbol{U}(\boldsymbol{x})}{\#\{\boldsymbol{x} \in \mathrm{V} : \mathrm{x_s} = 0\}}.
\end{aligned}
$$

Gibbs sampling can be proceeded as described in the outline of the MLE algorithm.

Although we have only attempted to fit marginals of dimensions 1 and 2 , this method is extendable to marginals of higher dimensions, but of course at higher computational expense.

Chapter 5

Fitting Gibbs Distributions to Mass Spectrometry Datasets

In this section, we apply the parameter estimation methods described in Chapter 4 to fit Gibbs distributions to real mass spectrometry datasets. Taking a group G of 80 mass spectra acquired from patients with late colorectal cancer as an example, we will show how one can model the empirical distribution of a group of binary coded mass spectra by a Gibbs distribution. To show the versatility of our methodology, we randomly select a small number of reference peak abscissas generated in Section 3.3 as key features. However, to ensure the significance of the model, we should avoid selecting the abscissas that are activated in none of the mass spectra of the group G. Therefore, here we randomly select 19 reference peak abscissas which have at least 15% activation frequencies in G:

$\mathcal{A} = \{3033, 8734, 7363, 8325, 6650, 4642, 2195, 1209, 3088, 6630, 2764, 3588, 1691,$

2926, 2156, 3878, 4118, 5725, 8275}.

We index these abscissas by $Ab_1, ..., Ab_{19}$. We code each mass spectrum M by a vector $\boldsymbol{x} = \boldsymbol{x}(M) \in \Lambda^S$, $s \in S = \{1, 2, ..., 19\}$ with $\Lambda = \{0, 1\}$: the sth coordinate of $\boldsymbol{x}(M)$ equals 1 if Ab_s is activated by M and equals 0 otherwise.

5.1 Clique Discovery

We use mutual information (discrete case) to quantify the strength of the stochastic link between two sites. The entropy of a discrete random variable ξ with possible values $\xi_1, ..., \xi_n$ is defined as

$$H(\xi) = -\sum_{\xi_i} p(\xi_i) log(p(\xi_i)),$$

and the conditional entropy of ξ with respect to another random variable η taking values $\eta_1, ..., \eta_m$ respectively is defined by

$$H(\xi|\eta) = -\sum_{i,j} p(\xi_i, \eta_j) log(p(\xi_i|\eta_j)).$$

For two random variables ξ, η, the mutual information is defined by

$$I(\xi;\eta) = H(\xi) - H(\xi|\eta) = \sum_{i,j} p(\xi_i, \eta_j) log \frac{p(\xi_i, \eta_j)}{p(\xi_i)p(\eta_j)}, \tag{5.1}$$

$$I(\xi;\eta) = I(\eta;\xi),$$

$$I(\xi;\eta) \geq 0.$$

"$I(\xi;\eta) = 0$ "indicates that ξ and η are independent.

Within the observation space $V = \{\boldsymbol{x}^{(1)} = \boldsymbol{x}(M_1), \boldsymbol{x}^{(2)} = \boldsymbol{x}(M_2), \cdots, \boldsymbol{x}^{(n)} = \boldsymbol{x}(M_n)\} \in \{0,1\}^{19}$ of the binary coded mass spectra $M_1, M2, \cdots, M_n$ of G, we compute the mutual information of any two coordinates X_s and X_t of a random variable $\boldsymbol{X} \in \{0,1\}^{19}$ $(s, t \in \{1, 2, \cdots, 19\})$. The larger the mutual information, the stronger the predictive link between two sites. As an example, some of the pairs of abscissas that have large mutual information are listed in Table 5.1, along with their empirical joint probabilities $q_{st}^{\delta_1,\delta_2}, \delta_1, \delta_2 \in \{0,1\}$. $q_{st}^{\delta_1,\delta_2}$ is computed by

$$q_{st}^{\delta_1,\delta_2} = \frac{\#\{\boldsymbol{x} \in V : x_s = \delta_1, x_t = \delta_2\}}{n}.$$

s	t	q_{st}^{00}	q_{st}^{01}	q_{st}^{10}	q_{st}^{11}	$I(s;t)$
5	10	0.36	0.01	0.06	0.56	0.42
3	19	0.39	0.18	0.09	0.35	0.12
7	13	0.50	0.26	0.03	0.21	0.12
1	11	0.26	0.00	0.43	0.31	0.12
3	13	0.19	0.38	0.34	0.10	0.10
15	18	0.33	0.28	0.05	0.35	0.10
8	14	0.73	0.18	0.01	0.09	0.09
4	13	0.40	0.48	0.13	0	0.09
1	19	0.21	0.05	0.26	0.48	0.08
13	14	0.48	0.05	0.26	0.21	0.08

Table 5.1: Empirical joint probabilities and mutual informations of pairs of sites

We consider the two sites whose mutual information are larger than 0.04 as significantly related pairs and recognize them as cliques of cardinal 2. We choose 0.04 as the cutoff because every pair that has mutual information less than 0.04 is decided to be independent by χ^2 test. There are 38 cliques $\{s, t\}$ of cardinal 2. The neighbors of each site are listed in Table 5.2.

s	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19
	3	18	1	1	10	3	9	14	3	3	1	4	3	3	1		6	2	1
	4		6	12	14	17	13	15	7	5	15	13	4	4	5		14	15	3
	11		9	13	15		14		13	15	19		7	5	8				4
	15		10	14					14				9	7	9				9
	19		13	19					15				12	8	10				11
			14						19				14	9	11				13
			19										19	13	18				
														15					
														17					

Table 5.2: Neighbors of sites
This table lists, for each site s, its neighbors sites. The site 16 has no neighbors because it is not included in any cliques of cardinal 2.

In an autologistic model such as (4.4), there is no canonical parameter that give zero marginals (e.g., in Table 5.1, $q^{0,1}_{1,11} = 0$ and $q^{11}_{4,13} = 0$). When zero marginals are obtained empirically, one can attempt to estimate a model which gives very small parametrized marginals correspondlingly. This approach is reasonable considering the variation of empirical probabilities computed from limited samples.

5.2 Estimated Parameters

We give the parameter estimation results for each one of the three methods MLE, MPLE and MFE. All of them use iterations, which require a starting point in the parameter space. We here choose a starting parameter point $\boldsymbol{\theta}(0) = (\cdots \theta_s \cdots \theta_{st} \cdots)^*$, where

$$\theta_s = -log(\frac{P\{x_s = 1\}}{P\{x_s = 0\}}), s = 1, 2, ..., 19,$$

and $\theta_{st} = 0$, for all $\{s, t\} \in \mathcal{C}$. Namely, this parameter perfectly fits all marginals of dimension 1 but assumes that pairs of sites are independent.

5.2.1 Maximum Likelihood Estimation (MLE) Results

We implemented a gradient descent method for MLE, in which we took 1000 iteration steps with a step size $\epsilon = 0.05$. During the Gibbs sampling, we picked a warm-up period of $b = 500$ and subsampled 2000 samples with a regular spacing $c = 4$. The computing time on a standard laptop was about 20 minutes. The estimated $\hat{\theta}_s$ and $\hat{\theta}_{st}$ in $\boldsymbol{\theta}$ are given in Table 5.3 and Table 5.4.

s	1	2	3	4	5	6	7	8	9	10
$\hat{\theta}_s$	-0.68	0.29	0.29	2.49	0.85	-2.96	2.66	2.52	-0.16	2.02
s	11	12	13	14	15	16	17	18	19	
$\hat{\theta}_s$	1.92	1.86	-0.33	1.58	0.43	-3.67	0.74	0.70	0.78	

Table 5.3: Gibbs parameter estimated by MLE (coefficients of singletons)
For each site s, this table lists the estimated coefficient $\hat{\theta}_s$ by MLE.

In order to monitor the maximization of the objective function (4.7), we can monitor its values during iterations. However, the computation is expensive due to the requirement of visiting all configurations for computing Z. To circumvent this, we monitor the gradient of the objective function, whose norm is supposed to converge to zero. However, the maximum cannot be reached exactly. We could only expect the norm of the gradient to become smaller than some reasonable small threshold. We plot the decrease of the norm of the gradient in Figure 5.1. We stopped when

$\{s,t\}$	{5,10}	{3,19}	{7,13}	{1,11}	{3,13}	{15,18}	{8,14}	{4,13}
$\hat{\theta}_{st}$	-2.88	-1.27	-1.44	-1.12	0.98	-1.71	-1.63	1.50
$\{s,t\}$	{1,19}	{13,14}	{13,19}	{10,15}	{7,9}	{9,14}	{1,15}	{5,15}
$\hat{\theta}_{st}$	-1.12	-0.95	1.05	-0.95	-1.02	-0.81	1.75	-0.87
$\{s,t\}$	{11,19}	{3,9}	{3,14}	{2,18}	{4,12}	{8,15}	{3,6}	{14,15}
$\hat{\theta}_{st}$	-1.16	0.76	1.11	-1.19	-1.40	1.18	1.15	1.06
$\{s,t\}$	{9,13}	{9,19}	{7,14}	{9,15}	{6,17}	{1,3}	{4,19}	{12,13}
$\hat{\theta}_{st}$	-0.50	0.82	-0.56	1.10	-0.68	-1.14	-0.67	1.16
$\{s,t\}$	{3,10}	{5,14}	{14,17}	{1,4}	{4,14}	{11,15}		
$\hat{\theta}_{st}$	-0.78	1.13	-1.00	-0.60	0.95	1.22		

Table 5.4: Gibbs parameter estimated by MLE (coefficients of neighbors)
For each pairs of sites $\{s,t\}$, this table lists the coefficient $\hat{\theta}_{st}$ of $x_s x_t$ estimated by MLE.

the norm of the gradient became smaller than 0.1.

5.2.2 Maximum Pseudolikelihood Estimation (MPLE) Results

We implemented gradient descent method for MPLE and took 200 iteration steps with a step size $\epsilon = 0.01$. The computing time was about 30 seconds. The estimated vector of parameters $th\hat{e}ta_s$ and $\hat{\theta}_{st}$ are given in Table 5.5 and 5.6.

The objective function (4.15) is expensive to calculate during the optimization process, so instead of monitoring it, we output the norm of its gradient (4.16) during the process, which is plotted in Figure 5.2. The gradient norm of gradient is supposed

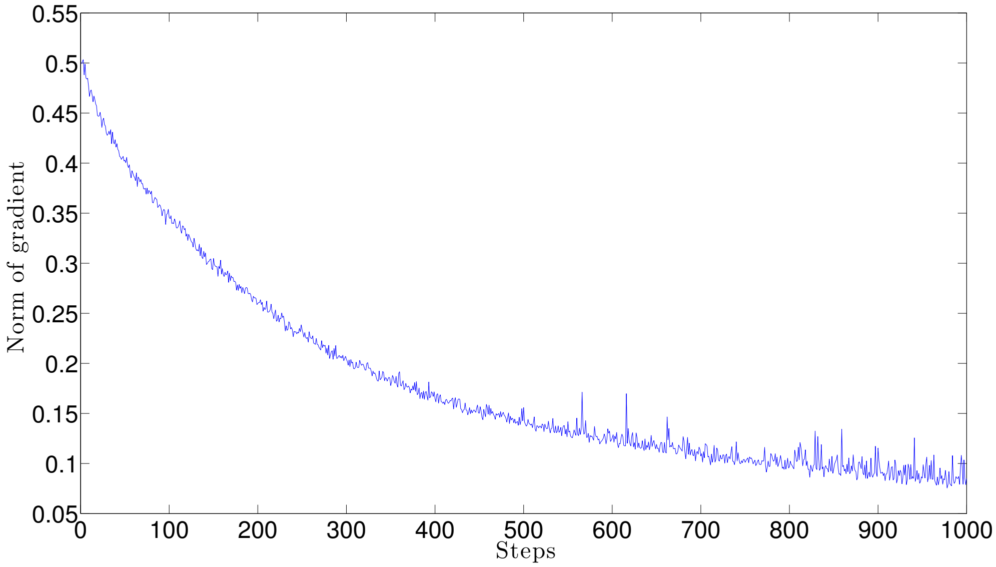


Figure 5.1: Decrease of the norm of the gradient in MLE

s	1	2	3	4	5	6	7	8	9	10
$\hat{\theta}_s$	-0.75	0.49	-0.18	3.91	1.68	-3.55	3.88	3.25	-0.73	3.92
s	11	12	13	14	15	16	17	18	19	
$\hat{\theta}_s$	3.41	1.95	-0.63	2.03	0.88	-3.66	2.27	0.98	1.24	

Table 5.5: Gibbs parameter estimated by MPLE (coefficients of singletons) For each site s, this table lists the estimated coefficient $\hat{\theta}_s$ by MPLE.

$\{s,t\}$	{5,10}	{3,19}	{7,13}	{1,11}	{3,13}	{15,18}	{8,14}	{4,13}
$\hat{\theta}_{st}$	-4.99	-1.33	-2.32	-2.45	1.50	-2.14	-2.76	2.73
$\{s,t\}$	{1,19}	{13,14}	{13,19}	{10,15}	{7,9}	{9,14}	{1,15}	{5,15}
$\hat{\theta}_{st}$	-1.46	-0.84	0.70	-1.75	-1.75	-0.77	2.64	-1.20
$\{s,t\}$	{11,19}	{3,9}	{3,14}	{2,18}	{4,12}	{8,15}	{3,6}	{14,15}
$\hat{\theta}_{st}$	-1.33	0.98	1.18	-1.49	-2.05	2.43	2.29	0.51
$\{s,t\}$	{9,13}	{9,19}	{7,14}	{9,15}	{6,17}	{1,3}	{4,19}	{12,13}
$\hat{\theta}_{st}$	0.20	1.10	-0.52	1.33	-2.21	-1.88	-1.20	1.94
$\{s,t\}$	{3,10}	{5,14}	{14,17}	{1,4}	{4,14}	{11,15}		
$\hat{\theta}_{st}$	-1.17	0.92	-1.20	-1.61	1.79	0.90		

Table 5.6: Gibbs parameter estimated by MPLE (coefficients of neighbors)
For each pairs of sites $\{s,t\}$, this table lists the coefficient $\hat{\theta}_{st}$ of $x_s x_t$ estimated by MPLE.

to decrease to zero, but we could only expect the gradient norm to become smaller than some reasonable small threshold. We stopped when the gradient norm became smaller than 1.

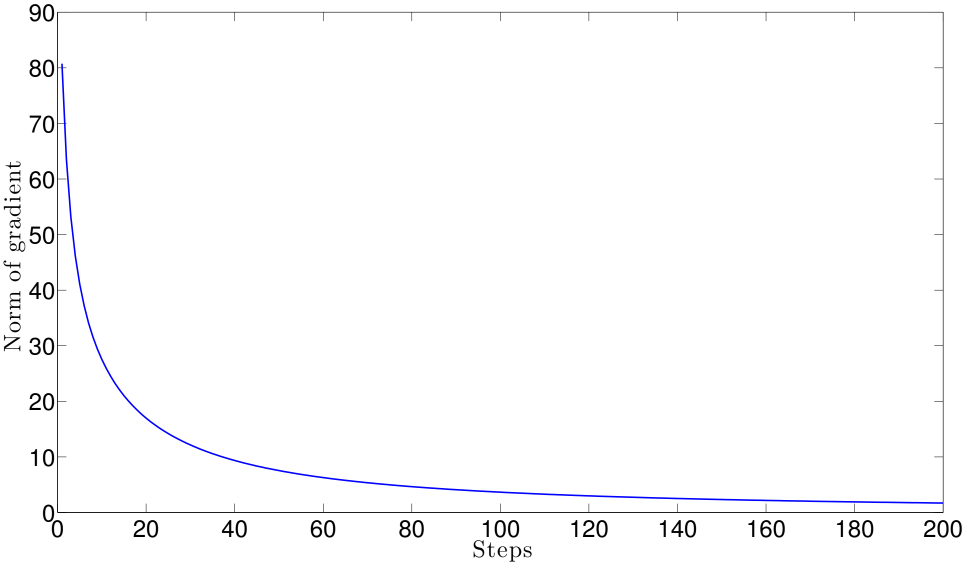


Figure 5.2: Decrease of the gradient norm for MPLE

5.2.3 Marginal Fitting Estimation (MFE) Results

The starting point gives us a good estimation on marginals of dimension 1 but not on those of dimension 2. Since the pairs of sites that have small mutual information are remotely affected by each other and practically independent, there is no need to take account of their joint marginals in the cost function SSE 4.19. Specifically, the joint marginals of 65 pairs that have mutual information larger than 0.008 are included in S in addition to the 17 one-dimensional marginals.

We took $N = 2000$ iteration steps with a step size $\epsilon = 0.05$. During the Gibbs sampling, we picked a warm-up period of $b = 500$ and subsampled 2000 samples with a regular spacing $c = 4$. We stopped when S became smaller than 0.4. The computing time was about 2.5 hours. The estimated parameter vector $\hat{theta}$ is given in Table 5.7 and 5.8. The evolution of the cost function during minimization is shown in Figure 5.2.3.

s	1	2	3	4	5	6	7	8	9	10
$\hat{\theta}_s$	-0.69	0.04	0.47	2.05	0.37	-2.81	2.27	2.34	0.01	1.43
s	11	12	13	14	15	16	17	18	19	
$\hat{\theta}_s$	1.45	1.89	-0.30	1.38	0.55	-3.69	0.15	0.42	0.43	

Table 5.7: Gibbs parameter estimated by MFE (coefficients of singletons) For each site s, this table lists the estimated coefficient $\hat{\theta}_s$ by MFE.

5.3 Quality of Fit

We propose a methodology for quantifying the quality of fit of a model on a real set $\mathcal{X}$ of n observed configurations. We first simulate 1000 virtual sets $\mathcal{V}_1, \mathcal{V}_2, \cdots, \mathcal{V}_{1000}$, where each $\mathcal{V}_j$ contains n random configurations generated by the Gibbs distribution $\pi_{\hat{\theta}}$. We then compute a log likelihood value $\hat{L}_i$ on each sample set $\mathcal{V}_i$ as well as a log likelihood value L on $\mathcal{X}$. The histogram of all $\hat{L}_1, \hat{L}_2, \cdots, \hat{L}_{1000}$ approximate the distribution of the log likelihood. The quantile of L on the histogram can be used to compare the quality of fit of different parameters. The closer the quantile to 50%, the better quality of fit the parameter has.

$\{s,t\}$	{5,10}	{3,19}	{7,13}	{1,11}	{3,13}	{15,18}	{8,14}	{4,13}
$\hat{\theta}_{st}$	-1.77	-1.16	-0.94	-0.69	1.06	-1.22	-0.52	0.69
$\{s,t\}$	{1,19}	{13,14}	{13,19}	{10,15}	{7,9}	{9,14}	{1,15}	{5,15}
$\hat{\theta}_{st}$	-0.90	-0.88	1.20	-0.94	-0.79	-0.88	1.32	-1.01
$\{s,t\}$	{11,19}	{3,9}	{3,14}	{2,18}	{4,12}	{8,15}	{3,6}	{14,15}
$\hat{\theta}_{st}$	-0.94	0.85	1.07	-0.82	-0.26	0.47	0.47	0.98
$\{s,t\}$	{9,13}	{9,19}	{7,14}	{9,15}	{6,17}	{1,3}	{4,19}	{12,13}
$\hat{\theta}_{st}$	-0.88	0.83	-0.61	0.90	-0.03	-0.70	-0.48	0.39
$\{s,t\}$	{3,10}	{5,14}	{14,17}	{1,4}	{4,14}	{11,15}		
$\hat{\theta}_{st}$	-0.81	1.39	-0.86	-0.21	0.44	1.18		

Table 5.8: Gibbs parameter estimated by MFE (coefficients of neighbors)
For each pairs of sites $\{s, t\}$, this table lists the coefficient $\hat{\theta}_{st}$ of $x_s x_t$ estimated by MFE.

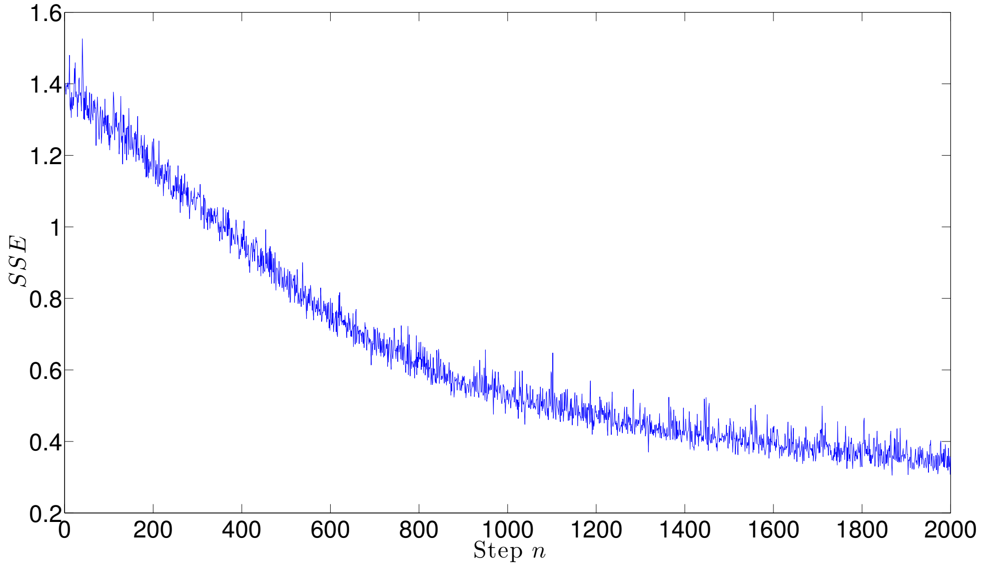


Figure 5.3: Minimization of the cost function for MFE

We then utilize this technique to compare the quality of fit of our Gibbs distribution with the three parameters $\hat{\boldsymbol{\theta}}_{MLE}$, $\hat{\boldsymbol{\theta}}_{MPLE}$ and $\hat{\boldsymbol{\theta}}_{MFE}$. The histogram of the 1000 log likelihoods with the three parameters are displayed in Figures 5.4, 5.5, and 5.6. The log likelihood value on the true dataset is indicated by red lines on the three figures. The three quantile 99.7%, 40.2%, and 99.8% naturally suggest that the Gibbs model with $\hat{\boldsymbol{\theta}}_{MPLE}$ has a better quality of fit than that with the other two parameters on our dataset.

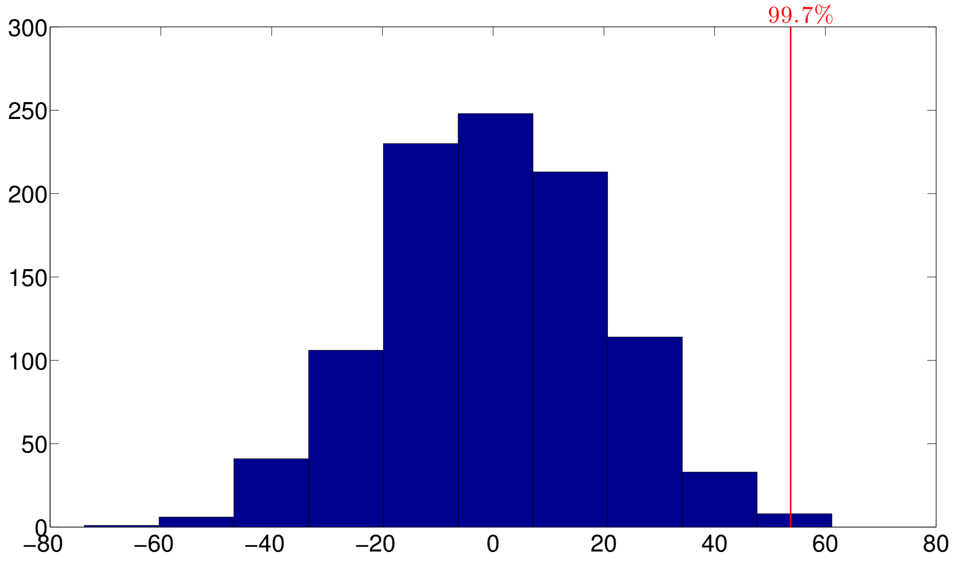


Figure 5.4: Quality of fit of $\pi_{\hat{\boldsymbol{\theta}}_{MLE}}$
The red line indicates the quantile of the log likelihood on the true dataset.

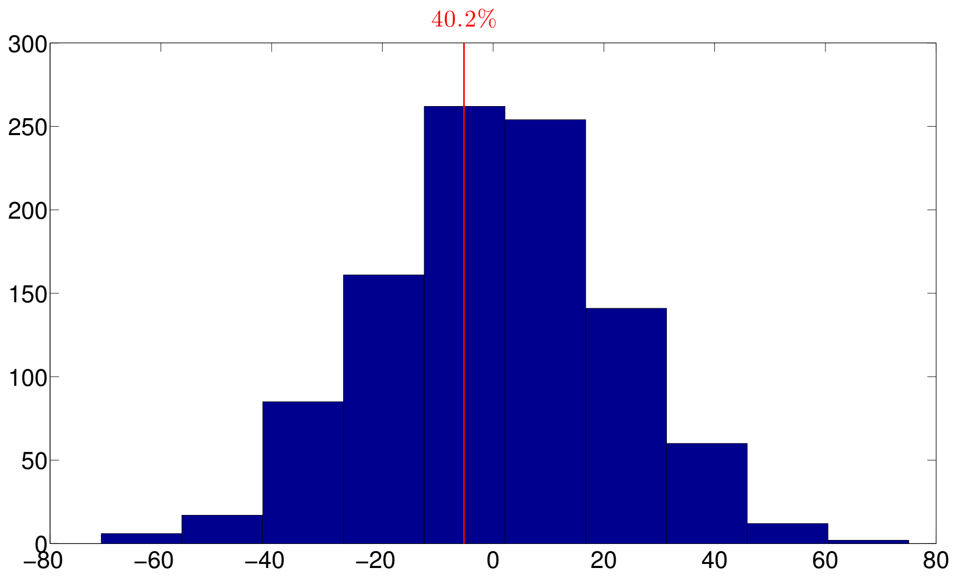


Figure 5.5: Quality of fit of $\pi_{\hat{\boldsymbol{\theta}}_{MPLE}}$
The red line indicates the quantile of the log likelihood on the true dataset.

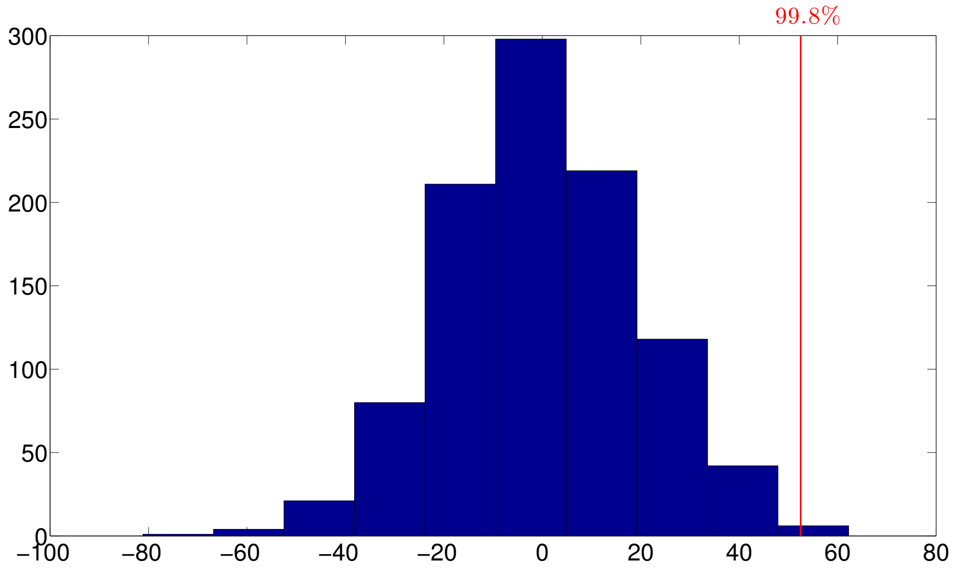


Figure 5.6: Quality of fit of $\pi_{\hat{\boldsymbol{\theta}}_{MFE}}$
The red line indicates the quantile of the log likelihood on the true dataset.

Chapter 6

Optimal Discrimination between Gibbs Distributions

The theoretical and numerical construction of Gibbs models on mass spectra leads us to the conclusion that we can characterize any mass spectrometry group by a Gibbs distribution. Knowing the distribution of two groups, there is an optimal classifier discriminating between the two groups derived from Neyman-Pearson lemma. We derive such an optimal classifier to two Gibbs distributed groups. We also introduce the Kullback-Leibler distance to quantify the difference between Gibbs distributions.

6.1 Discriminating Power of a Classifier

Let x be an observations. x can belong to one of two groups G^+ (labeled by $+1$) and G^- (labeled by -1) with distribution densities $\pi^+(x)$ and $\pi^-(x)$. Any classifier g on

x produces two types of errors

$$L^+(g) = P(g(x) = -1|x \in G^+),\ L^-(g) = P(g(x) = +1|x \in G^-). \qquad (6.1)$$

Sensitivity $Sens(g)$ and specificity $Spec(g)$ are commonly used statistical measures of the performance of a binary test, which are equivalent to $1 - L^+(g)$ and $1 - L^-(g)$ respectively. To compare performances among classifiers, we define the "discriminating power" of a classifier by the average of sensitivity and specificity

$$DP = \frac{Sens(g) + Spec(g)}{2}. \qquad (6.2)$$

6.2 Optimal Classifier and Optimal Discriminating Power

Let's first introduce the classical Neymann-Pearson approach to hypothesis testing. For any observation x of the random variable $X \in \mathcal{X}$, consider the two hypothesis

$$H_0 : x \in G^+,\ H_1 : x \in G^-.$$

Lemma 6.2.1. Neyman-Pearson lemma[47] *Given a threshold $t > 0$, the test which rejects H_0 in favor of H_1 when*

$$\frac{\pi^+(x)}{\pi^-(x)} < t$$

gives the most powerful test of size (Type I error) L^+. Namely, it maximizes the power of test $\rho = 1 - L^-$ as L^+ is fixed. L^+ and L^- are defined by Equation 6.1.

With a similar proof of Neyman-Pearson lemma, we have the following theorem.

Theorem 6.2.2. *The decision rule*

$$\mathring{g}(x) = \begin{cases} +1, & \textit{if } \dfrac{\pi^+(x)}{\pi^-(x)} > 1, \\ -1, & \textit{otherwise.} \end{cases} \tag{6.3}$$

is the optimal classifier that minimizes the error $(L^+(g) + L^-(g))/2$*, or equivalently, maximizes the discriminating power*

$$DP(g) = \frac{Sens(g) + Spec(g)}{2} = 1 - (L^+(g) + L^-(g))/2.$$

Proof: Let $\rho = 1 - L^-(g)$ and state the problem as seeking $\mathring{g}$ such that

$$\mathring{g} = \arg\min_g L^+(g) + L^-(g) = \arg\max_g \rho - L^+(g)$$

In fact, one seeks the region $R = \{x : H_1 \text{ is accepted}\}$ that maximizes the objective function

$$\begin{aligned} \mathcal{L}(L^+(g), \rho) &= \rho - L^+(g) \\ &= \int_R (\pi^-(x) - \pi^+(x))dx. \end{aligned}$$

The region that maximizes $\mathcal{L}(L^+(g), \rho)$ is

$$R = \{\frac{\pi^+(x)}{\pi^-(x)} < 1\},$$

because $\pi^-(x) - \pi^+(x) > 0$ on R and $\pi^-(x) - \pi^+(x) \leq 0$ on $(-\infty, \infty)\backslash R$. This gives the classifier (6.3).

We call the classifier (6.3) the "optimal classifier" and its discriminating power the "optimal discriminating power".

6.3 Optimal Discrimination between Two Gibbs Distributions

Given two Gibbs distributions $\pi_{\boldsymbol{\theta}^+}(\boldsymbol{x})$ and $\pi_{\boldsymbol{\theta}^-}(\boldsymbol{x})$ on a configuration $\boldsymbol{x}$

$$\pi_{\boldsymbol{\theta}^+}(\boldsymbol{x}) = \frac{1}{Z^+}e^{-\boldsymbol{\theta}^{+*}\boldsymbol{U}(\boldsymbol{x})} = \frac{1}{Z^+}e^{-\sum_{s\in S}\theta_s^+ x_s - \sum_{\{s,t\}\in\mathcal{C}}\theta_{st}^+ x_s x_t}, \tag{6.4a}$$

$$\pi_{\boldsymbol{\theta}^-}(\boldsymbol{x}) = \frac{1}{Z^-}e^{-\boldsymbol{\theta}^{-*}\boldsymbol{U}(\boldsymbol{x})} = \frac{1}{Z^-}e^{-\sum_{s\in S}\theta_s^- x_s - \sum_{\{s,t\}\in\mathcal{C}}\theta_{st}^- x_s x_t}, \tag{6.4b}$$

where $\boldsymbol{\theta}^+ = (\cdots\ \theta_s^+\ \cdots\ \theta_{st}^+\ \cdots)^*, \boldsymbol{\theta}^- = (\cdots\ \theta_s^-\ \cdots\ \theta_{st}^-\ \cdots)^*$ are parameters, $\boldsymbol{U}(\boldsymbol{x}) = (\cdots\ x_s\ \cdots\ x_s x_t\ \cdots)^*$, S and $\mathcal{C}$ are the cardinal-1 and cardinal-2 cliques of the two groups. Since

$$\frac{\pi_{\boldsymbol{\theta}^+}(\boldsymbol{x})}{\pi_{\boldsymbol{\theta}^-}(\boldsymbol{x})} > 1$$

implies

$$(\boldsymbol{\theta}^{-*} - \boldsymbol{\theta}^{+*})\boldsymbol{U}(\boldsymbol{x}) > \log Z^+ - \log Z^-,$$

the optimal classifier is

$$\mathring{g}(\boldsymbol{x}) = \begin{cases} +1, & \text{if } \mathring{f}(\boldsymbol{x}) > 0, \\ -1, & \text{otherwise.} \end{cases} \tag{6.5}$$

where $\mathring{f}(\boldsymbol{x}) = (\boldsymbol{\theta}^{-*} - \boldsymbol{\theta}^{+*})\boldsymbol{U}(\boldsymbol{x}) - \log Z^+ + \log Z^-$.

If the singletons or cliques are different between two distributions, we can always construct a set of cliques of cardinal 1 and 2 which is the union of the corresponding sets of cliques for π^+ and π^-. In those cases, $\boldsymbol{\theta}^-$ and $\boldsymbol{\theta}^+$ may have some of their coordinates equal to zero.

6.4 Kullback-Leibler Distances between Two Gibbs Distributions

We here utilize Kullback-Leibler (KL) divergence (relative entropy) to quantify the distance between two probability distributions P and Q. In a discrete measure space, the divergence is defined by

$$D(P\|Q) = \sum_x P(x) \ln \frac{P(x)}{Q(x)}.$$

Since $D(P\|Q)$ is non-symmetric and non-negative, we define the Kullback-Leibler distance between P and Q to be

$$\mathrm{dis(P,Q) = D(P\|Q) + D(Q\|P).}$$

This distance takes values in the range $[0, \infty]$ and $dis(\pi_1, \pi_2) = 0$ if and only if $\pi_1 = \pi_2$.

Given two Gibbs distributions

$$\pi_1(\boldsymbol{x}) = \frac{e^{-\boldsymbol{\theta}_1^* \boldsymbol{U}_1(\boldsymbol{x})}}{Z_1},$$
$$\pi_2(\boldsymbol{x}) = \frac{e^{-\boldsymbol{\theta}_2^* \boldsymbol{U}_2(\boldsymbol{x})}}{Z_2},$$

we compute their distance by

$$\begin{aligned}
&\mathrm{dis}(\pi_1, \pi_2) = \mathrm{D}(\pi_1\|\pi_2) + \mathrm{D}(\pi_2\|\pi_1) \\
&= \sum_{\boldsymbol{x}} \frac{e^{-\boldsymbol{\theta}_1^* \boldsymbol{U}_1(\boldsymbol{x})}}{Z_1} (\ln \frac{Z_2}{Z_1} + \boldsymbol{\theta}_2^* \boldsymbol{U}_2(\boldsymbol{x}) - \boldsymbol{\theta}_1^* \boldsymbol{U}_1(\boldsymbol{x})) \\
&+ \sum_{\boldsymbol{x}} \frac{e^{-\boldsymbol{\theta}_2^* \boldsymbol{U}_2(\boldsymbol{x})}}{Z_2} (\ln \frac{Z_1}{Z_2} + \boldsymbol{\theta}_1^* \boldsymbol{U}_1(\boldsymbol{x}) - \boldsymbol{\theta}_2^* U_2(\boldsymbol{x})) \\
&= \sum_{\boldsymbol{x}} (\boldsymbol{\theta}_2^* \boldsymbol{U}_2(\boldsymbol{x}) - \boldsymbol{\theta}_1^* \boldsymbol{U}_1(\boldsymbol{x}))(\pi_1(\boldsymbol{x}) - \pi_2(\boldsymbol{x})).
\end{aligned}$$

Thus, the distance between two Gibbs distributions is

$$\operatorname{dis}(\pi_1, \pi_2) = \mathrm{E}_{\pi_1}(\boldsymbol{\theta}_2^* \boldsymbol{U}_2 - \boldsymbol{\theta}_1^* \boldsymbol{U}_1) + \mathrm{E}_{\pi_2}(\boldsymbol{\theta}_1^* \boldsymbol{U}_1 - \boldsymbol{\theta}_2^* \boldsymbol{U}_2). \tag{6.6}$$

If $\boldsymbol{U}(\boldsymbol{x}) = \boldsymbol{U}_1(\boldsymbol{x}) = \boldsymbol{U}_2(\boldsymbol{x})$, for all $\boldsymbol{x}$, (6.6) becomes

$$\operatorname{dis}(\pi_1, \pi_2) = \mathrm{E}_{\pi_1}((\boldsymbol{\theta}_2^* - \boldsymbol{\theta}_1^*)\boldsymbol{U}) + \mathrm{E}_{\pi_2}((\boldsymbol{\theta}_1^* - \boldsymbol{\theta}_2^*)\boldsymbol{U}). \tag{6.7}$$

The KullbackLeibler divergence $D(p(x|H_1)\|p(x|H_0))$ can also be viewed as the mean information for discrimination between H_1 and H_0 per observation from $p(x|H_1)$ [69]. The Kullback-Leibler Distance we define here sums the mean information per observation from both $p(x|H_1)$ and $p(x|H_0)$ for discrimination between H_1 and H_0.

Chapter 7

Signatures and Signature-based Classification

One of the drawbacks of using machine learning algorithms to generate proteomic classifiers is that the biological interpretation of "black-box" classifiers is generally hard or impossible to decipher. To develop clinically usable mass spectrometry analysis tools, a key step is to discover "biomarker signature profiles", i.e. combinations of a small number of protein biomarkers which strongly discriminate between cancer states. To circumvent this, we develop two innovative signature discovery algorithms to automatically identify small groups of biomarkers with nearly optimal power of discrimination between two groups G^+ and G^-. These two methods will be illustrated in details in Chapters 8 and 9. In this chapter, we set up the framework of signature discovery algorithm by giving the definition of signatures and sketching the general procedure of signature-based classification.

7.1 Selection of Biomarker Target Pools by NP Lemma

A biomarker, generally refers to as an m/z value in proteomic study, which can be used as an indicator of some biological states. However, individual m/z values often are only weakly discriminating. Therefore, here we take into consideration not only single m/z values, but also pairs of m/z values. The reason adding pairs of m/z values into the scope will be further explained in next section. Our approach is also easily extendable to groups of three or more m/z values, although there is a trade off between computational cost and increased discriminating power.

A biomarker B is defined here as either a single reference peak abscissa Ab or as a pair of distinct reference peak abscissas (Ab_i, Ab_j), called a single-peak biomarker or a double-peak biomarker. The activation frequency for a single or a pair of reference peak abscissas in Section 3.3 also applies to a biomarker. To select a small set of optimal biomarkers, called "biomarker target pool" from the whole set of biomarkers for highly efficient signature discovery, we introduce a threshold on ratios of biomarker activation frequencies.

Activation frequencies statistically describe the discriminating nature of a biomarker across mass spectra in distinct groups. Considering the discrimination task between two distinct groups G^+ and G^-, each biomarker B has then two well defined activation frequencies: $fq^+(B)$ among the mass spectra of G^+, and $fq^-(B)$ among the mass spectra of G^-.

The NP lemma (Lemma 6.2.1) indicates that the most powerful test to discriminate between G^+ and G^- on the basis of presence or absence of a single biomarker B is achieved by adequate thresholding for the ratio of activation frequencies $fq^+(B)/fq^-(B)$.

Definition 7.1.1. We say that B is a $\boldsymbol{G^+}$ **biomarker** if $fq^+(B)/fq^-(B) > 1$ and a $\boldsymbol{G^-}$ **biomarker** if $fq^-(B)/fq^+(B) > 1$.

For a fixed k, we rank all G^+ single-peak biomarkers and double-peak G^+ biomarkers by descending $fq^+(B)/fq^-(B)$ respectively, and also rank all single-peak G^- biomarkers and double-peak G^- biomarkers by descending $fq^-(B)/fq^+(B)$ respectively; within each of the four ranked biomarker groups, we pick the top k biomarkers. The integer k is a parameter and will be optimized later (refer to Section 7.4).

7.2 Binary Coding of Mass Spectra

We code every mass spectrum based on r pre-selected biomarkers $B_1, B_2, \cdots, B_r$. Call $\mathcal{W}$ the set of all "binary vectors" of dimension r, having all their coordinates equal to either 0 or 1. We systematically code each mass spectrum M as a binary vector $\boldsymbol{W} = \boldsymbol{W}(M)$ belonging to $\mathcal{W}$, by setting each coordinate W_j of $\boldsymbol{W}$ equal to 1 if the biomarker B_j is activated by M and to 0 if B_j is not activated by M. This binary coding transforms each high-dimensional spectrum M into a low-dimensional vector $\boldsymbol{W}(M)$ of dimension r.

Assume in a Gibbs model (4.4), a clique $\{s\}$ or $\{s, t\}$ corresponds to a biomarker defined on Ab_s or (Ab_s, Ab_t), then $x_s = 1$ or $x_s x_t = 1$ corresponds to the activation of Ab_s or (Ab_s, Ab_t). In Chapter 6, the optimal classifier to discriminate between two Gibbs distributed groups involves an extended configuration space $\mathcal{U}$ (with configurations such as $(\cdots, x_s, \cdots, x_s x_t, \cdots)$) instead of the original configuration space $\mathcal{X}$ (with configurations such as $(x_1, \cdots, x_s, \cdots)$). Therefore, we take into consideration of double-peak biomarker in addition to single-peak biomarkers in our coding scheme, for the reason that a binary configuration space $\mathcal{W}$ constructed by coding mass spectra with both single-peak and double-peak biomarkers is just an extended configuration space $\mathcal{V}$ constructed by coding mass spectra with only single-peak biomarkers.

Two groups G^+ and G^- of mass spectra will give two subsets of $\mathcal{W}$. In the following sections, we will be dealing with the two subsets.

7.3 Signatures and Signature-based Classifiers

It has been generally accepted that using a small number of biomarkers combinatorially yields higher discriminating power than when they are used individually. Therefore, within the biomarker target pool just constructed, we aim to extract small lists of the biomarkers to construct high performance classifiers. Furthermore, for clinical usage and biology studies, one naturally requests more interpretability and biological significance to be provided by the discrimination method. Therefore, the discovery of "signature profiles" yielding efficient cancer group classifiers is currently a key step to facilitate clinical diagnosis.

Given a training dataset involving two pre-classified groups of mass spectra G^+ and G^-, a signature Sig will be any fixed list of biomarkers picked within the target pool $TP(r) = \{B_1, B_2, \cdots, B_r\}$, namely, $Sig \subseteq TP(r)$. One naturally seeks signatures which characterize the key patterns of protein expression levels and hence can potentially discriminate between distinct cancer patient groups. An interesting goal is to associate a weight to each one of these biomarkers.

To each signature $Sig = \{B_1, B_2, \cdots, B_q\}$ $(q < r)$, for example, we associate a function f_{Sig} on a binary coded mass spectrometry configuration space $\mathcal{W}$ based on $\{B_1, B_2, \cdots, B_q\}$

$$f_{Sig} : \mathcal{W} \longmapsto \mathbb{R}.$$

This function defines a classifier g_{Sig} on two subsets of $\mathcal{W}$, labeled by +1 and -1, from two groups of mass spectra G^+ and G^-

$$g_{Sig}(\boldsymbol{x}) = \begin{cases} +1, \text{ if } f_{Sig}(\boldsymbol{x}) > 0, \\ -1, \text{ if } f_{Sig}(\boldsymbol{x}) \leq 0. \end{cases} \tag{7.1}$$

The function f_{Sig} is related to Sig because its domain $\mathcal{W}$ is constructed by coding mass spectra with biomarkers in Sig. For any mass spectrum M, which is coded into $\boldsymbol{W}(M) \in \mathcal{W}$, (7.1) is equivalent to the decision rule

$$M \in \begin{cases} G^+, \text{ if } f_{Sig}(\boldsymbol{W}(M)) > 0, \\ G^-, \text{ if } f_{Sig}(\boldsymbol{W}(M)) \leq 0. \end{cases} \tag{7.2}$$

As proved in Chapter 6, the optimal $\mathring{f}_{Sig}$ between two Gibbs distributed binary sets in $\mathcal{W}$ should be a linear function

$$\mathring{f}_{Sig}(\boldsymbol{x}) = \boldsymbol{a}^* \boldsymbol{x} + b = \sum_{i=1}^{q} a_i x_i + b. \tag{7.3}$$

Therefore, we restrict our search for $\mathring{f}_{Sig}$ on the class of linear functions in our study. For $B_1, B_2, \cdots, B_q$ of Sig, the weights a_i evaluates the impact of the activation of B_i. The discriminating power of a signature is defined to be the discriminating power of its associated classifier. Therefore, searching a signature with high discriminating power between two groups of mass spectra G^+ and G^- is equivalent to searching for one signature associated classifier on their binary coded configuration space with high discriminating power.

Typically, there are two types of biomarkers in a signature for the discrimination task G^+ versus G^-: the biomarkers whose activation by a binary vector $\boldsymbol{W} = \boldsymbol{W}(M)$ points towards the decision $M \in G^+$ and the biomarkers whose activation indicates the group G^-. A natural choice is that the weight a_j of a G^+ biomarker should be positive and weight a_j of a G^- biomarker should be negative. This conclusion can also be inferred from the results presented in Chapter 6.

In a Gibbs model (4.4), a clique x_s or $x_s x_t$ corresponding to a biomarkers defined on Ab_s or (Ab_s, Ab_t) with high activation frequency should have large marginals $P\{x_s = 1\}$ or $P\{x_s = 1, x_t = 1\}$, therefore is expected to have large coefficient $-\theta_s$ or $-\theta_{st}$ in a Gibbs model

$$\pi_{\boldsymbol{\theta}}(\boldsymbol{x}) = \frac{1}{Z} e^{-\sum_{s\in S} \theta_s x_s - \sum_{\{s,t\}\in \mathcal{C}} \theta_{st} x_s x_t}.$$

Assume that we have fitted a Gibbs model

$$\pi_{\boldsymbol{\theta}^+}(\boldsymbol{x}) = \frac{1}{Z^+} e^{-\boldsymbol{\theta}^{+*}\boldsymbol{U}(\boldsymbol{x})}$$

to the data set G^+ as in Chapter 4, then the coefficients of $-\boldsymbol{\theta}^+$ corresponding to G^+ biomarkers should be large and those corresponding to G^- biomarkers should be small. Similarly, in $-\boldsymbol{\theta}^-$ of a G^-model

$$\pi_{\boldsymbol{\theta}^-}(\boldsymbol{x}) = \frac{1}{Z^-} e^{-\boldsymbol{\theta}^{-*}\boldsymbol{U}(\boldsymbol{x})},$$

the coefficients corresponding to G^- biomarkers should be large and those corresponding to G^+ biomarkers should be small. Therefore, the coefficents of $\boldsymbol{\theta}^- - \boldsymbol{\theta}^+$ that corresponding to G^+ biomarkers should be positive and that corresponding to G^- biomarkers should be negative.

7.4 Performance Evaluation and Model Selection

For a data mining model generated from a dataset, its capacity to fit into new example sets is known as its generalization capacity, which is an important goal in data mining [15].

A practical method to compute the generalization capacity of a model is to divide the available examples into two completely distinct sets, one is used to train a model, called training set, and the other one is used as a testing set to validate the model, called testing set. To get an accurate evaluation of the generalization capacity of this model, one needs a large number of training and testing examples.

In the case of limited example size, k-fold cross validation [83] is a powerful model validation technique to assesses the performance of a model. in this approach, one randomly partitions original samples into k equal size example subsets $V_1, V_2, \cdots, V_k$, of which, a single subset V_i is retained as the validation dataset and the rest subsets are used as training set. This process is then repeated k times until each of the

$V_i (i = 1, \cdots, k)$ subsets has been used exactly once as the validation set. Leave-one-out cross validation is one particular case of k-fold cross validation method where $k = 1$.

We propose here a modified leave-one-out cross validation method, called leave-two-out cross validation to further explore the potential of cross validations on datasets of small size. Suppose G^+ and G^- have N_1 and N_2 examples. In the "leave-two-out" technique, we select at random two examples, x^+ in G^+ and x^- in G^- as the validation data and the other examples are used as a training set. This process is then repeated $N_1 * N_2$ times until each pair of examples (one from G^+ and one from G^-) is used exactly once as the validation set.

To evaluate the generalization capacity of our signature discovery model for the discrimination task G^+ vs G^-, we implement our leave-two-out cross validation technique. Still assume that G^+ and G^- have N_1 and N_2 subjects. We select at random two subjects A^+ in G^+ and A^- in G^- as the validation data and the other subjects are used as a training set to generate a signature Sig and a signature related classifier g_{Sig}. After each pairs of subjects is used exactly once, we obtained $N_1 * N_2$ results of classification of subjects in G^+ and also $N_1 * N_2$ results of classification of subjects in G^-. The sensitivity and specificity can be then computed from these results.

In some cases, where we have two replicate spectra M_1 and M_2 per subject, we classify a subject according to the average value of $f_{Sig}(\boldsymbol{W}(M_1))$ and $f_{Sig}(\boldsymbol{W}(M_2))$ of any classifier g_{Sig} we obtained. Using more replicate spectra for each subject is more reliable than using only one replicate per subject in in order to reduce the effects of experimental variations and errors.

The integer k (see Section 7.1) controls the complexity of our model. To seek for optimal "signatures" combining small numbers of biomarkers and having high discrimination capacity, we consider a range of appropriate values for k, and find the best one that gives the model with largest generalization capacity for each particular discrimination task [15].

7.5 Signature Reliability Score

Peak detection during pre-processing is impacted by data acquisition noise. Therefore, the activation of any specific biomarker is not fully stable across repeated data acquisitions. Since biomarkers are the bases of a signature, the reliability of a signature is then questionable. To quantify the stability of a signature across multiple data acquisitions, we compute a signature "reliability score" as follows.

During pre-processing, we have already computed a reliability score for each detected strong peak (See Section 3.2). The reliability score $R(Sig)$ of any signature Sig is then defined as the average of $R(P)$ over all peaks P (detected in the last step of Pre-processing) that are within the error window of any biomarkers in Sig. Among all signatures with similar discriminating power between G^+ and G^-, we naturally prefer to select signatures with higher reliability scores.

Chapter 8

Signature Discovery by Robust Log-likelihood (RLL)

As numerically proved in Chapter 5, Gibbs models can be used to describe the distribution of homogeneous groups of mass spectra based on the activation/non-activation of a moderate number of well selected biomarkers. Given two groups of mass spectra G^+ and G^- with Gibbs distributions π^+ and π^-, we have derived in Chapter 6 that the optimal classifier to discriminate between the two groups is (6.5). In this chapter, we introduce a signature discovery method to search for this optimal classifier and simultaneously generate a signature profile for two groups in a discrimination task. This signature discovery technique will be called a "robust log-likelihood" (RLL) algorithm.

8.1 Robust Log-likelihood Algorithm

Recall that discriminating between two Gibbs models π^+ and π^-, the optimal classifier

$$\mathring{g}(\boldsymbol{x}) = \begin{cases} +1, & \text{if } \mathring{f}(\boldsymbol{x}) > 0, \\ -1, & \text{otherwise.} \end{cases} \tag{8.1}$$

where

$$\mathring{f}(\boldsymbol{x}) = (\boldsymbol{\theta}^- - \boldsymbol{\theta}^+)^* \boldsymbol{U}(\boldsymbol{x}) - \log Z^- - \log Z^+,$$

can be naturally generated when models π^+ and π^- are already explicitly parametrized by $\boldsymbol{\theta}^+$ and $\boldsymbol{\theta}^-$

$$\begin{aligned} \pi^+ &= \pi_{\boldsymbol{\theta}^+}(\boldsymbol{x}) = \frac{1}{Z^+} e^{-\boldsymbol{\theta}^{+*}\boldsymbol{U}(\boldsymbol{x})}, \\ \pi^- &= \pi_{\boldsymbol{\theta}^-}(\boldsymbol{x}) = \frac{1}{Z^-} e^{-\boldsymbol{\theta}^{-*}\boldsymbol{U}(\boldsymbol{x})}. \end{aligned}$$

Estimating a Gibbs model is not very accurate, especially when we have only a small number of mass spectra in our dataset. Therefore, after code each mass spectrum M into a binary vector $\boldsymbol{W}(M) = (W_1, W_2, \cdots, W_r)$ in a binary space $\mathcal{W}(r)$ as in Section 7.2, we can avoid Gibbs modeling and blindly search for a function f in a family of linear functions $\mathcal{F}(r) = \{f : f(\boldsymbol{x}) = \boldsymbol{a}\boldsymbol{x} + b = \sum_{i=1}^{r} a_i x_i + b\}$ on $\mathcal{W}(r)$, with which the classifier (8.1) has a largest discriminating power. We call this signature discovery technique robust log-likelihood (RLL).

Recall that a G^+ biomarker is a biomarker B with $fq^+(B)/fq^-(B) > 1$ and a G^- biomarker is a biomarker B with $fq^-(B)/fq^+(B) > 1$, where $fq^+(B)$ and $fq^-(B)$ are activation frequencies of B in G^+ and G^-. In $\mathring{f}$, the coefficients corresponding to

G^+ biomarkers should be positive and those corresponding to G^- biomarkers should be negative.

The optimal classifier (8.1) indicates that the best choice of $\boldsymbol{a}$ and b should be respectively proportional to $\boldsymbol{\theta}^- - \boldsymbol{\theta}^+$ and to $\log Z^+ - \log Z^-$. Theoretically, if the underlying Gibbs distributions are already parametrized (by direct estimation of coefficiemts for instance) in π^+ and π^-, there should exists a $\lambda \in \mathbb{R}$ such that

$$\boldsymbol{a} = \lambda(\boldsymbol{\theta}^- - \boldsymbol{\theta}^+), \;\; b = \lambda(\log Z^+ - \log Z^-). \tag{8.3}$$

Given a target pool $TP(r)$ of r biomarkers, our signature discovery by RLL for two groups of mass spectra G^+ and G^- proceeds as follows.

1. Code all mass spectra of G^+ and G^- into binary vectors in $\mathcal{W}(r)$ based on current r biomarkers. The vectors corresponding to mass spectra of G^+ are labeled by +1 and those corresponding to mass spectra of G^- are labeled by -1.

2. Search for a function $f(\boldsymbol{x}) = \sum_{i=1}^{r} a_i x_i + b$ in the family of linear functions $\mathcal{F}(r)$ with which the classifier (8.1) classifies two classes of vectors in $\mathcal{W}(r)$ with a largest discriminating power.

3. Then

   - If any G^+ biomarker has a negative coefficient or any G^- biomarker has a negative coefficient, delete it from current biomarker pool $TP(r)$; The remaining $r'(r' < r)$ biomarkers constitute a new biomarker target pool, denoted by $TP(r')$. Let $TP(r) = TP(r')$ and go to Step 1.

- Otherwise, end of the algorithm.

The remaining G^+ biomarkers and G^- biomarkers at the end of the algorithm define a signature $Sig \subseteq TP(r)$. Their corresponding coefficients in the final classifier are their weights.

8.2 Performance of RLL Classifiers (Simulated Data)

With large enough number of training samples, an efficient classification algorithm is expected to find the nearly optimal classifier (see Section 6.3) when the biomarker pool is correctly given. A correct biomarker pool should include biomarkers corresponding to all cliques in the two Gibbs models. The following experiment evaluates the performance of RLL signature discovery algorithm on a large simulated dataset and demonstrates its efficiency numerically. Although the following simulation based analysis is carried out only for two particular Gibbs distributions, it is easily extendable to any pair of Gibbs distributions.

Two groups V^+ (labeled by +1) and V^- (labeled by -1) of random binary vectors of lengths 10 are simulated from two pre-defined virtual Gibbs distributions

$$\pi_{\boldsymbol{\theta}^+}(\boldsymbol{x}) = \frac{1}{Z} e^{-\boldsymbol{\theta}^+ \boldsymbol{U}(\boldsymbol{x})}$$

and

$$\pi_{\boldsymbol{\theta}^-}(\boldsymbol{x}) = \frac{1}{Z} e^{-\boldsymbol{\theta}^- \boldsymbol{U}(\boldsymbol{x})}.$$

There are 10 sites $\{1, 2, \cdots, 10\}$ and 6 cardinal-2 cliques $\{1, 2\}$, $\{1, 3\}$, $\{2, 3\}$, $\{6, 7\}$,

$\{6,8\}$ and $\{7,8\}$ in the two models. $\boldsymbol{\theta}^+$, $\boldsymbol{\theta}^-$ and $\boldsymbol{U}(\boldsymbol{x})$ are set to

$$\begin{aligned}
\boldsymbol{\theta}^+ &= (-1 \;\cdots -1 \;\; -0.3 \cdots -0.3 \quad -1 \quad -1 \quad -1 \quad -0.3 \quad -0.3 \quad -0.3)^*,\\
\boldsymbol{\theta}^- &= (-0.3 \cdots -0.3 \;\; -1 \;\cdots\; -1 \quad -0.3 \quad -0.3 \quad -0.3 \quad -1 \quad -1 \quad -1)^*,\\
\boldsymbol{U}(\boldsymbol{x}) &= (x_1 \quad \cdots \quad x_5 \quad\; x_6 \quad \cdots \quad x_{10} \quad x_1x_2 \quad x_1x_3 \quad x_2x_3 \quad x_6x_7 \quad x_6x_8 \quad x_7x_8)^*.
\end{aligned}$$

The distance between the two Gibbs distributions computed by equation (6.7) is $dis(\pi_{\boldsymbol{\theta}^+}, \pi_{\boldsymbol{\theta}^-}) = 3.43$. The optimal discriminating power (see formula (6.1)) between the two Gibbs distributions when $\boldsymbol{\theta}^+$ and $\boldsymbol{\theta}^-$ are fully known is $\overset{\circ}{DP} = 81.6\%$.

Using these two explicit models, we simulated 1000 random binary configurations $\boldsymbol{x}^{(1)}, \cdots, \boldsymbol{x}^{(1000)}$ from $\pi_{\boldsymbol{\theta}^+}$ and 1000 random binary configurations $\boldsymbol{y}^{(1)}, \cdots, \boldsymbol{y}^{(1000)}$ from $\pi_{\boldsymbol{\theta}^-}$. This gives us two virtual data sets V^+ and V^- of binary vectors in $\mathbb{R}^{10}$. We randomly chose a training set TR^+ of size 500 within V^+ and a training set TR^- of size 500 within V^-.

Then we implemented our RLL signature discovery algorithm on $TR^+ \cup TR^-$ to train a classifier g. The biomarker target pool was simply set to be all the biomarkers corresponding to 10 singletons and 6 pairwise cliques. We didn't include extra biomarkers because in that case, a new problem related to verification of the efficiency of selecting biomarkers will be raised. Here we ONLY want to check the efficiency of our RLL algorithm. We utilize linear kernel SVM to facilitate linear classification in this algorithm. Other algorithms, such as Perceptron [76], that support linear classification can be used too. A Matlab toolbox LIBLINEAR-1.91[89] is used to implement linear kernel SVM.

The discriminating power of the classifier g tested on the rest of the 1000 samples

$V^+ \cup V^- \backslash (TR^+ \cup TR^-)$ is 81.6 %, which coincides with the optimal discriminating power $\overset{\circ}{DP}$. All the 16 biomarkers have been included by RLL algorithm in the discovered signature. In addition, the parameters in the classifier g approximately satisfy equations (8.3). These has successfully verified the efficiency of our RLL signature discovery.

8.3 Convergence Rates of RLL Classifiers (Simulated Data)

A natural question now is how fast the RLL signature discovery algorithm converges with respect to the sample size n. The answer to this question enables us to estimate the smallest but sufficient number of samples to reach any classification performance level with a certain algorithm. Due to the small size of real mass spectrometry samples, we still need to study simulated datasets. This section provides a numerical approach to analyze the convergence rate of an algorithm on a given dataset of binary coded mass spectra in the context of large deviation theory.

8.3.1 Large Deviation Framework

We follow the framework in [8]. Suppose T_N is a consistent estimate of g. Namely, for any $\epsilon > 0$,

$$\alpha_N(\epsilon) \to 0 \text{ as } N \to \infty,$$

where

$$\alpha_N(\epsilon) = P_\theta\{|T_N - g| > \epsilon\}. \tag{8.4}$$

Large deviation theory implies that if the distribution of the g has finite exponential moments, then T_N is asymptotically normal distributed,

$$-\frac{\log \alpha_N(\epsilon)}{N} \to \gamma(\epsilon), \text{ as } N \to \infty,$$

where $\gamma(t)$ is the Cramer transform of distribution of g

$$\gamma(t) = \sup_{\xi>0}(\xi t - \lambda(\xi)), \ \lambda(\xi) = \ln E(e^{\xi g}).$$

This indicates that the convergence rate of T_N to g is $e^{-N\gamma(\epsilon)}$ when N is large enough.

8.3.2 Numerical Study

We still study the two Gibbs models in Section 8.2. For increasing N, we systematically implemented the following simulations. For each N, we perform 1000 simulations $SIM_1, SIM_2, \cdots, SIM_{1000}$. In each simulation SIM_j, we generate $N/2$ binary vectors of size 10 drawn from the Gibbs distribution π^+ and $N/2$ binary vectors of size 10 drawn from the Gibbs distribution π^-. We then implemented our RLL signature discovery on this training set TR_j of total size N samples. The biomarker target pool is still set to be all the biomarkers corresponding to the 10 single sites and 6 pairwise cliques. The discriminating power $DP(j)$ of the obtained classifier g_j can be computed by utilizing the formula (6.1). After repeating this for $j = 1, \cdots, 1000$, we obtain 1000 simulated values $DP(j)$ of the classifier g_j. We can then compute the mean discriminating power $DP_N = [DP(1) + \cdots + DP(1000)]/1000$ and the associated standard error Err_N of $DP(j)(j = 1, \cdots, 1000)$. DP_N and Err_N for several values of N are listed in Table 8.1. With the increasing size N of the training set, the discriminating power DP_N achieved by RLL converges to the optimal discriminating power 81.6% achievable when π^+ and π^- are already known and the standard error of estimation Err_N on DP_N decreases to zero. The optimal discriminating power 81.6% is practically reached for $N = 200$. We also checked the obtained signatures after every training process. The signatures generated by RLL are short when N is small. With the increasing of N, the signatures contain more and more biomarkers. When N is around 1400, the signatures converge to the true one, which includes the whole set of biomarkers.

In the formula (8.3.1), let $T_N = DP_N$, then Figure 8.1 shows the value of $-log(\alpha_N(\epsilon))$ as a function of N when $\epsilon = 0.1\%, 0.5\%, 1\%, 5\%$. Comparing the four plots in Figure 8.1, we conclude that $\gamma(\epsilon)$ is an increasing function.

Note that every curve stops earlier before as N increases. The reason is that as N increases, the theoretical $\alpha_N(\epsilon)$ decreases; when $\alpha_N(\epsilon)$ is small enough, its estimation using only 1000 simulations is essentially always equal to 0. In order to observe the large deviation behavior at large values of N, one needs to sample huge numbers of simulations SIM_j, which are very expensive computations. Due to the limitation of current computer capacity and time restraints, we didn't complete this study. But by comparing figures like Figure 8.1 for different algorithms, we can still get an idea about the different convergence rates of various types of algorithms (see Figure 9.1).

n	$DP_N(\%)$	$Err_N(\%)$
2	70.8	7.4
4	73.4	5.2
6	74.6	4.8
8	75.2	4.1
10	76.5	3.8
12	76.6	3.2
14	77.2	2.9
16	77.3	2.5
18	77.2	2.3
20	77.6	2.4
40	78.5	1.7
60	79.2	1.4
80	79.5	1.1
100	80.0	1.0
120	80.2	0.8
140	80.6	0.8
160	80.6	0.8
180	80.8	0.7
200	80.9	0.6
400	81.3	0.3
600	81.5	0.2
800	81.6	0.1
1000	81.6	0.1
1200	81.6	0.1
1400	81.6	0.0
1600	81.6	0.0
1800	81.6	0.0
2000	81.6	0.0

Table 8.1: Performance of RLL signature discovery

At each N, we simulate a training set of size N using π^+ and π^- and test our RLL signature discovery on this dataset. This process is repeated 1000 times for each N. We can then estimate the mean discriminating power DP_N achieved by RLL algorithms with N training examples. We estimate also the standard error Err_N of DP_N. We report DP_N and Err_N as percentages in the above table.

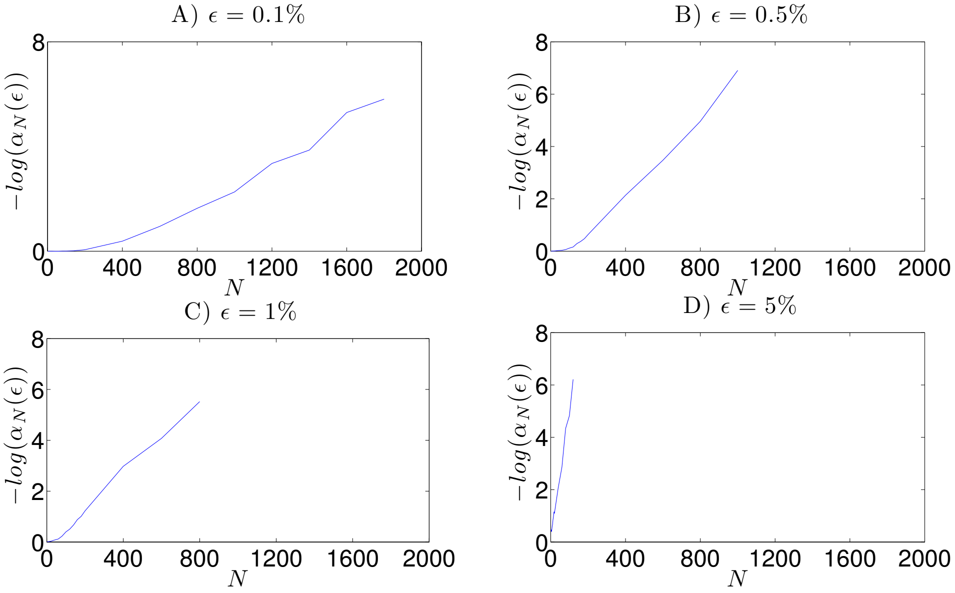


Figure 8.1: Convergence rate of RLL signature discovery

Chapter 9

Signature Discovery by Maximizing Detecting Power (MDP)

In this chapter, we present another innovative algorithm for signature discovery. We first seek two signatures, Sig^+ optimized to detect G^+ patients and Sig^- optimized to detect G^- patients. We then combine Sig^+ and Sig^- to generate an optimal signature for discrimination between G^+ and G^-. Separate searches for Sig^+ and Sig^- are implemented by a stochastic optimization method, Simulated Annealing, to handle efficiently the high combinatorial complexity. As our results show, Simulated Annealing is a powerful optimization tool for signature discovery. Other stochastic optimization methods, such as genetic algorithms, possibly could be substituted to Simulated Annealing in our algorithm, but we have not studied these alternative stochastic optimization techniques.

9.1 Stochastic Optimization of Group Detecting Power

9.1.1 The Two Scoring Functions of a Signature

Given a training dataset involving two pre-classified groups of mass spectra G^+ and G^-, and a biomarker target pool $TP(r) = \{B_1, B_2, \cdots, B_r\}$, we associate to any tentative signature $Sig \subseteq TP(r)$ two scoring functions

$$M \mapsto s^+(M, Sig), \ M \mapsto s^-(M, Sig),$$

defined for all mass spectra M as follows.

Definition 9.1.1. Given any spectrum M, we count, within the given fixed signature Sig, the number $u^+(M)$ of G^+ biomarkers activated by M and the number $v^+(M)$ of G^- biomarkers which are NOT activated by M. The $\boldsymbol{G^+}$ **score** of M for signature Sig is then defined by $s^+(M, Sig) = (u^+(M)+v^+(M))/card(Sig)$, $card(Sig)$ denotes the number of biomarkers in Sig.

We count, within Sig, the number $u^-(M)$ of G^- biomarkers activated by M and the number $v^-(M)$ of G^+ biomarkers which are NOT activated by M. The $\boldsymbol{G^-}$ **score** of M for signature Sig is then defined by $s^-(M, Sig) = (u^-(M)+v^-(M))/card(Sig)$.

We then define the "G^+ detecting power" and "G^- detecting power" of the signature Sig.

Definition 9.1.2. For each fixed threshold $0 < c < 1$, the signature Sig determines a G^+ classifier by assigning any observed spectrum M to G^+ if $s^+(M, Sig) \geq c$ and to G^- otherwise. Denote the discriminating power of this classifier by $DP(c, Sig)$. There is an easily computable optimal threshold c maximizing $DP(c, Sig)$. This maximized performance $J^+(Sig) = \max_{0<c<1} DP(c, Sig)$ defines the $\boldsymbol{G^+}$ **detecting power** of Sig. Exchanging G^+ and G^-, as well as the scores $s^+(M, Sig)$ and $s^-(M, Sig)$ we similarly define the $\boldsymbol{G^-}$ **detecting power** $J^-(Sig)$ of Sig.

Among all signatures Sig included within our biomarkers target pool $TP(r)$, we will now seek two signatures Sig^+ and Sig^- respectively by maximizing separately the detecting powers $J^+(Sig)$ and $J^-(Sig)$.

But $J^+(Sig)$ and $J^-(Sig)$ have many local maxima, and the set of all signatures included within $TP(r)$ has very large cardinal. To solve this combinatorial challenge, we implement the separate maximizations of $J^+(Sig)$ and of $J^-(Sig)$ by Simulated Annealing as described in the following.

9.1.2 Simulated Annealing (SA) Algorithm

Simulated Annealing, a powerful stochastic optimization method, searches for a global optimum of a function U defined on a large and often discrete space $\mathcal{E}$ [6] [41][1]. It randomly explores the configurations in the configuration space $\mathcal{E}$ with a probabilistic acceptance rule parametrized by a very slowly decreasing virtual "temperature", and converges almost surely to a global optimum of U.

Given an arbitrary function $U : \mathcal{E} \mapsto \mathbb{R}$, where $\mathcal{E}$ is an arbitrary finite set, called configuration space and U is called the energy function, the generic Simulated Annealing algorithm generates a stochastic sequence $X_n \in \mathcal{E}$, such that X_n concentrates on the set of absolute minima of U as n tends to infinity. There are three mathematical structures to be defined in order to implement a Simulated Annealing (SA) algorithm: the transition probabilities, a neighborhood system and a cooling schedule. The transition probability $q(i, j)$ is the probability that SA algorithm explores the configuration j when its current configuration is i. In principle the choice of transition probabilities $q(i, j)$ is arbitrary but q should be symmetric and irreducible. The set of the neighbors of configuration i is the set $V_i = \{j \in \mathcal{E} \mid q(i, j) > 0\}$. Define a cooling schedule $\{T_n\}$, where the sequence of numbers $T_n > 0$ are called temperatures, and satisfies $T_n \geq T_{n+1}$, $\forall n \geq 0$. The cooling sequence T_n should in principle decrease very slowly to zero.

The SA algorithm for maximization of $U(x)$ starts from an arbitrary configuration X_0. At step n, one selects a random neighbor Y_n of X_n, such that $P(Y_n = j \mid X_n) = q(X_n, j)$. Then if $U(Y_n) > U(X_n)$, set $X_{n+1} = Y_n$; if $U(Y_n) \leq U(X_n)$, let $p = e^{-\frac{1}{T_n}(U(X_n)-U(Y_n))}$ and make a random choice between $X_{n+1} = Y_n$ and $X_{n+1} = X_n$ with

$$P(X_{n+1} = Y_n) = p, \ P(X_{n+1} = X_n) = 1 - p.$$

9.1.3 Implementation of Signature Search by Simulated Annealing

We optimize $J^+(Sig)$ by Simulated Annealing as follows.

1. Select any initial signature $Sig_0 \subseteq TP(r)$

2. Fix a periodic sequence $j(n)$ visiting integers 1 to r repeatably

3. At step n, define a new signature $\widetilde{Sig}$ by:

   - If $B_{j(n)} \in Sig_n$, let $\widetilde{Sig} = Sig_n \cup \{B_{j(n)}\}$

   - Otherwise, let $\widetilde{Sig} = Sig_n \backslash B_{j(n)}$

   Then

   - If $J^+(\widetilde{Sig}) > J^+(Sig_n)$, set $Sig_{n+1} = \widetilde{Sig}$

   - Otherwise, select a random number u with uniform distribution:

     - if $u < e^{-\frac{1}{0.95^n}(J^+(Sig_n) - J^+(\widetilde{Sig}))}$, set $Sig_{n+1} = \widetilde{Sig}$

     - Otherwise, set $Sig_{n+1} = Sig_n$

4. Repeat Steps 3 until J^+ stabilizes

Typically, J^+ will stabilize after $200 \times r$ repetitions of Steps 3. This completes one time of Simulated Annealing search and gives one (or more) signature(s) that maximize(s) $J^+(Sig)$. To enhance performance, we implement multiple Simulated Annealing searches and retain the signature Sig^+ achieving the highest maximum for $J^+(Sig)$.

The optimization of $J^-(Sig)$ is implemented similarly.

9.1.4 Signature-based Classifier

After computing signatures Sig^+ and Sig^-, one can focus on the two scores $s^+ = s^+(Sig^+, M)$ and $s^- = s^-(Sig^-, M)$ of a mass spectrum M. Consider the decision

$$M \in \begin{cases} G^+, \text{ if } s^+ \geq C\ s^- + D, \\ G^-, \text{ otherwise,} \end{cases} \tag{9.1}$$

C, D can be easily computed by any algorithm that searches for linear classifier (such as SVM with linear kernel).

When we have two replicate spectra M_1, M_2 per patient, we replace $s^+(Sig^+, M_1)$ and $s^+(Sig^+, M_2)$ by their average s^+ and do the same for s^-, before constructing as above the best linear separator based on the sign of $s^+ - (C\ s^- + D)$.

To fit into the framework of Chapter 7, we state here that the decision rule (9.1) is a particular case of (7.2). Recall that in Section 7.3, the optimal decision rule (7.2) is

$$M \in \begin{cases} G^+, \text{ if } f_{Sig}(\boldsymbol{W}(M)) > 0, \\ G^-, \text{ if } f_{Sig}(\boldsymbol{W}(M)) \leq 0. \end{cases}$$

where

$$\mathring{f}_{Sig}(\boldsymbol{x}) = \boldsymbol{a}^* \boldsymbol{x} + b = \sum_{i=1}^{q} a_i x_i + b.$$

Note that given two lists of biomarkers Sig^+ and Sig^- and a mass spectrum M, both $s^+(M)$ and $s^-(M)$ can be represented as linear combinations of the coordinates of the binary vector $\boldsymbol{W}(M)$. This is simply because $u^+(M)$, $v^+(M)$, $u^-(M)$ and $v^-(M)$ are all linear combinations of some coordinates of $\boldsymbol{W}(M)$. Therefore, $s^+ - (C\ s^- + D)$ can be rephrased as a linear function in the form of $\mathring{f}_{Sig}(\boldsymbol{x})$.

9.2 Performance of MDP Classifiers (Simulated Data)

We follow the same process in Section 8.2 to evaluate the efficiency of MDP signature discovery algorithm on simulated data. Similarly, using these two explicit models, we simulated 1000 random binary configurations $\boldsymbol{x}^{(1)}, \cdots, \boldsymbol{x}^{(1000)}$ from $\pi_{\boldsymbol{\theta}^+}$ and 1000 random binary configurations $\boldsymbol{y}^{(1)}, \cdots, \boldsymbol{y}^{(1000)}$ from $\pi_{\boldsymbol{\theta}^-}$. This gives us two virtual data sets V^+ and V^- of binary vectors in $\mathbb{R}^{10}$. We randomly chose a training set TR^+ of size 500 within V^+ and a training set TR^- of size 500 within V^-.

Then we implemented our MDP signature discovery algorithm on $TR^+ \cup TR^-$ to train a classifier g. The biomarker target pool was simply set to be all the biomarkers corresponding to 10 singletons and 6 pairwise cliques. The discriminating power of the classifier g tested on the rest of the 1000 samples $V^+ \cup V^- \backslash (TR^+ \cup TR^-)$ is is 80.7%, which, taking account of the error of estimations, is not really distinguishable from the optimal discriminating 81.6%. We can say that these simulations do validate the efficiency of this signature discovery method.

9.3 Convergence Rates of MDP Classifiers (Simulated Data)

Under the same framework of Section 8.3, we discuss about the convergence rate of this signature discovery algorithm. Similarly, for increasing N, we systematically implemented the following simulations. For each N, we perform 1000 simulations $SIM_1, SIM_2, \cdots, SIM_{1000}$. In each simulation SIM_j, we generate $N/2$ binary vectors of size 10 drawn from the Gibbs distribution π^+ and $N/2$ binary vectors of size 10 drawn from the Gibbs distribution π^-. We then implemented our MDP signature discovery on this training set TR_j of total size N samples. The biomarker target pool is still set to be all the biomarkers corresponding to the 10 single sites and 6 pairwise cliques. The discriminating power $DP(j)$ of the obtained classifier g_j can be computed by utilizing the formula (6.1). After repeating this for $j = 1, \cdots, 1000$, we obtain 1000 simulated values $DP(j)$ of the classifier g_j. We can then compute the mean discriminating power $DP_N = [DP(1) + \cdots + DP(1000)]/1000$ and the associated standard error Err_N of $DP(j)(j = 1, \cdots, 1000)$. DP_N and Err_N for several values of N are listed in Table 8.1. With the increasing of sample size, the mean of the discriminating power DP_N converges to 80.7% which is very close to the optimal discriminating power 81.6%. The error of estimation on DP_N decreases and stabilizes at 1.1%. We also checked the signature Sig^+, Sig^-, and Sig for each N. The signatures obtained for small N are short. For increasing N, the signatures contain more and more biomarkers. When n reaches 2000, the signatures are almost the same as the true one, which includes the whole set of biomarkers.

In the fomula (8.3.1), let $T_N = DP_N$, Figure 9.1 shows the value of $-log(\alpha_N(\epsilon))$ as a function of N when $\epsilon = 0.1\%, 0.5\%, 1\%, 5\%$. The comparison of the four plots indicates that $\gamma(\epsilon)$ does not change much at low level ϵ but has a dramatic increasing around $\epsilon = 5\%$. However, comparing the four plots to those in Figure 8.1, we conclude that signature discovery by MDP, at least in this simple case, converges

n	$DP_N(\%)$	$Err_n(\%)$
2	71.1	10.1
4	76.3	6.2
6	77.9	3.6
8	78.3	3.1
10	78.1	3.1
12	78.1	3.2
14	78.2	3.0
16	78.6	2.6
18	78.2	3.0
20	78.2	3.0
40	78.9	2.0
60	79.6	1.9
80	79.6	1.7
100	80.0	1.6
120	80.2	1.4
140	80.3	1.5
160	80.2	1.3
180	80.4	1.3
200	80.6	1.1
400	80.7	1.1
600	80.7	1.1
800	80.7	1.1
1000	80.7	1.1
1200	80.6	1.1
1400	80.6	1.1
1600	80.7	1.1
1800	80.6	1.1
2000	80.7	1.1

Table 9.1: Performance of MDP signature discovery

At each N, we simulate a training set of size N using π^+ and π^- and test our MDP signature discovery on this dataset. This process is repeated 1000 times for each N. We can then estimate the mean discriminating power DP_N achieved by MDP algorithms with N training examples. We estimate also the standard error Err_N of DP_N. We report DP_N and Err_N as percentages in the above table.

slower than signature discovery by RLL on all ϵ levels.

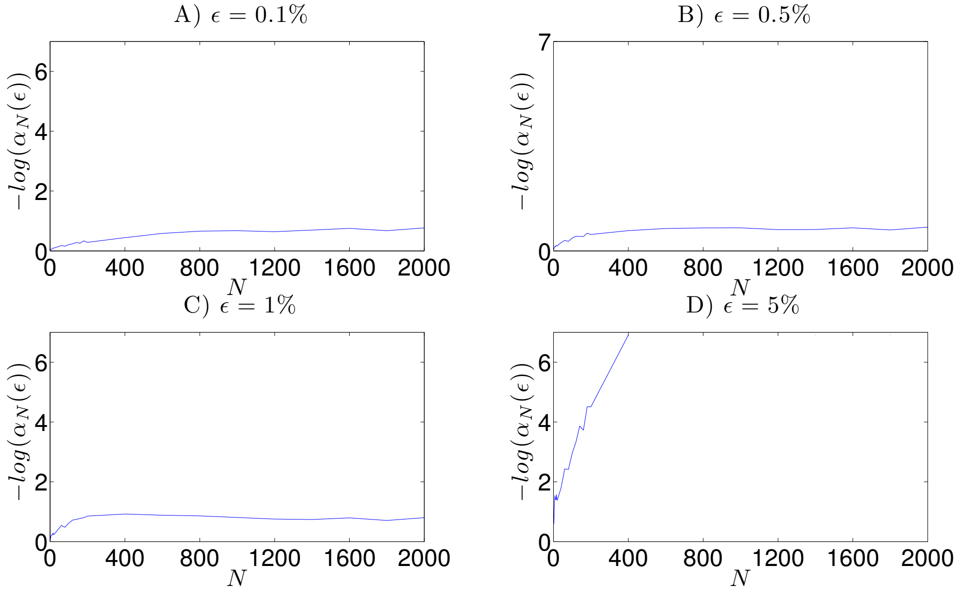


Figure 9.1: Convergence rate of MDP signature discovery

Chapter 10

Mass Spectrometry Data Study

Colorectal cancer (CRC) and Ovarian cancer (OC) are two of the commonly diagnosed type of cancer worldwide. In this study, we validate our signature discovery algorithms described in Chapter 8 and 9 on a new experimental MALDI-TOF dataset, acquired from 3 groups of colorectal cancer patients (3 stages) and one control group, and two well-known SELDI-TOF datasets on ovarian cancer patients and control patients. For all the homogeneous patient groups in these datasets we have generated explicit signatures functional in each discrimination task with high discriminating power.

10.1 Mass Spectrometry Datasets

10.1.1 Mass Spectra of Colorectal Cancer

104 colorectal cancer (CRC) samples and 15 control samples were provided by Section of Surgical Clinic II, Department of Surgical Oncology and Gastroenterological Sciences, University of Padova, Padova, Italy. Between 2002 and 2005, the 104 cancer patients underwent surgeries and histopathological diagnosis. Among them, 27 were diagnosed with colorectal pre-cancer lesion (Adenoma), 40 with early colorectal cancer (stage I or II), and 37 with late colorectal cancer (stage III or IV). The 15 healthy patients all received colonoscopy and were diagnosed to be unaffected.

A 10 ml blood sample was collected from each patient into a DB Vacutainer during the surgery or colonoscopy and transferred to the laboratory within 4 hours of collection, to be centrifuged at 3,000 rpm for 10 min. Plasma samples were then collected from the supernatant and stored in aliquots at -80°C in the Tumor Tissue Biobank of Surgical Clinic I as well as during transportation, until analysis.

For efficient removal of high molecular weight proteins and for specific isolation and enrichment of LMW species present in 15μl of plasma, we used (see [16]) a novel three steps size-exclusion strategy based on Mesoporous Silica Chips, fabricated by Dept of Nanomedicine (Methodist Hospital Research Institute, Houston, Texas). Mass spectra were acquired in linear positive-ion mode (range 800-10,000 ”m/z” ratio) on a Voyager-DE-STR MALDI TOF Mass Spectrometer (Applied Biosystems, Framingham, MA, USA) at Research Center of Protein Chemistry Core Laboratory

(University of Texas Health, Houston, Texas). The manufacturer provided spectrometer accuracy was $\rho = 0.3\%$ (see Section 3.3). Only one blood plasma sample was extracted from each subject, but two "replicate" mass spectra were acquired from each blood plasma sample.

In total, 238 mass spectra replicates were acquired from 4 patients groups, with 2 mass spectra replicates per patient: the Control group CTR of 15 patients, the Adenoma group ADE of 27 patients with precancer lesions, the group ECR of 40 patients with Early ColoRectal cancer (stage I-II), the group LCR of 37 patients with Late ColoRectal cancer (stage III-IV). We also studied the whole cancerous group CRC of 104 patients pooling together all three cancer groups ADE, ECR, and LCR. Each mass spectrum provides about 36,900 m/z values between 800-10,000 on the x-axis and the associated "peptide intensities" on the y-axis.

10.1.2 Mass Spectra of Ovarian Cancer

Two well-known ovarian cancer (OVC) datasets can be freely downloaded from NCI-FDA clinical proteomics databank (http://home.ccr.cancer.gov /ncifdaproteomics /ppatterns.asp). The dataset obtained on 04/03/02 consists of 116 control (normal or benign) samples and 100 cancer samples, collected using the WCX2 protein chip and a Ciphergen PBS1 SELDI-TOF mass spectrometer. The dataset obtained on 08/07/02 gathers 91 control and 162 cancer samples, also collected using the WCX2 chip but with an upgraded PBSII SELDI-TOF mass spectrometer. The samples were prepared manually for the dataset of 04/03/02 and by robotic hardware for dataset of 08/07/02. These two mass spectrometers had $\rho = 0.1\%$ accuracy (see Section 3.3) and each mass spectrum listed 15,154 distinct m/z values in the 0-20,000 range. In dataset of 04/03/02, baselines have been removed prior to display for public access.

The dataset of 04/03/02 has been studied in [65], [102], and [5], which reported classification accuracies of 97.5%, 100%, and 86.66%. The dataset of 08/07/02 was studied by [102], [82], [4], and [5], which reported classification accuracies of 100%.

In our study, we denote the ovarian and control groups in the dataset of 04/03/02 by "OVC04" and "CTR04" and those in the dataset of 08/07/02 by "OVC08" and "CTR08".

10.2 Signature Discovery for Cancer Discrimination

Throughout this thesis, we consider on CRC dataset, the 4 discrimination tasks ADE vs ECR, ADE vs LCR, ECR vs LCR, and CRC vs CTR, where CRC denotes the union group of the three cancer groups: ADE, ECR, and LCR; and consider on the two OVC datasets, the 2 tasks "OVC04 vs CTR04" and "OVC08 vs CTR08". Generically, we denote a discrimination task by G^+ vs G^-.

We validated our two signature discovery algorithms elaborated in Chapter 8 and Chapter 9 on CRC and OVC datasets. There are slight differences between the two experiments. We systematically implemented the two separate steps pre-processing and signature discovery on the two datasets and present them in two sections.

10.2.1 Pre-processing

The 238 MALDI mass spectra in our colorectal dataset were restricted to m/z ratios from 800 to 10,000. We plot one spectrum in each of the colorectal groups (ADE, ECR, LCR, and CTR) in Figures 10.1, 10.2, 10.3, and 10.4.

There are obviously several large peaks that are common to all the four mass spectra, which might represent some peptides that can be seen in every patient's plasma proteins. They are not useful for discrimination between groups. Only those that tends to present in one group rather than the other have high value to achieve a discrimination goal. Those peaks might have comparably smaller intensity, which are not easy to be detected only by looking at these spectra. Our pre-processing algorithm intends to detect all the significant peaks (no matter with "small" or "large" intensity) in every spectrum, which will be used later for signature discovery and group classification.

At m/z abscissa x, the half window widths ux, vx, wx, and tx (see Section 3.1) for spectrum smoothing, noise extraction, baseline computation, and peaks selection were implemented with values $u = 0.0003, v = 0.017, w = 0.025$, and $t = 0.0005$. Our peak detection was based on a peak strength threshold $th = 2$. After setting these parameters, our pre-processing algorithm was implemented on all the 238 mass spectra sequentially and automatically.

We take a piece of the mass spectrum in Figure 10.4 from 1800 to 2000 m/z ratios as an example to explain the pre-processing procedures. Normalization, smoothing, and baseline removal have vertically rescaled mass spectra and have diminished the

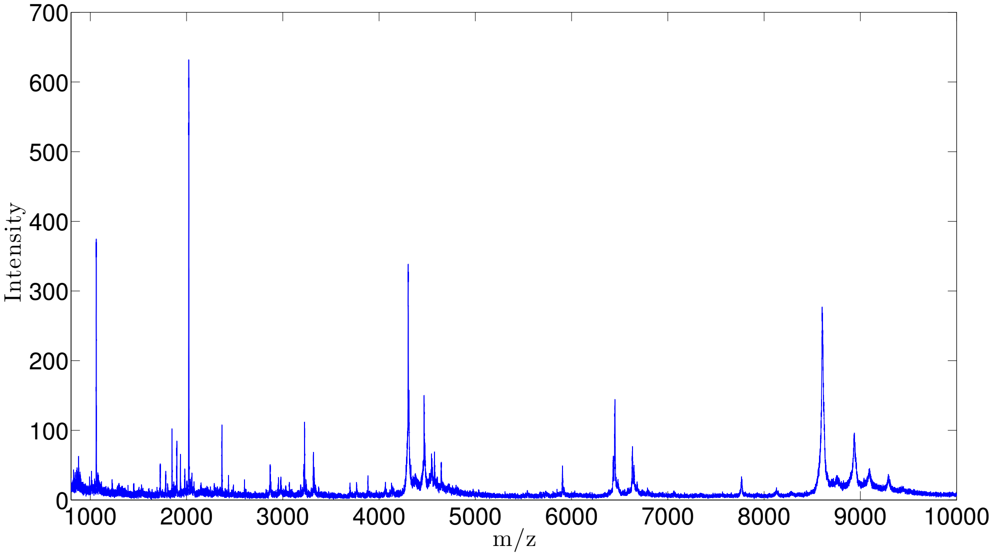


Figure 10.1: A raw mass spectrum of ADE

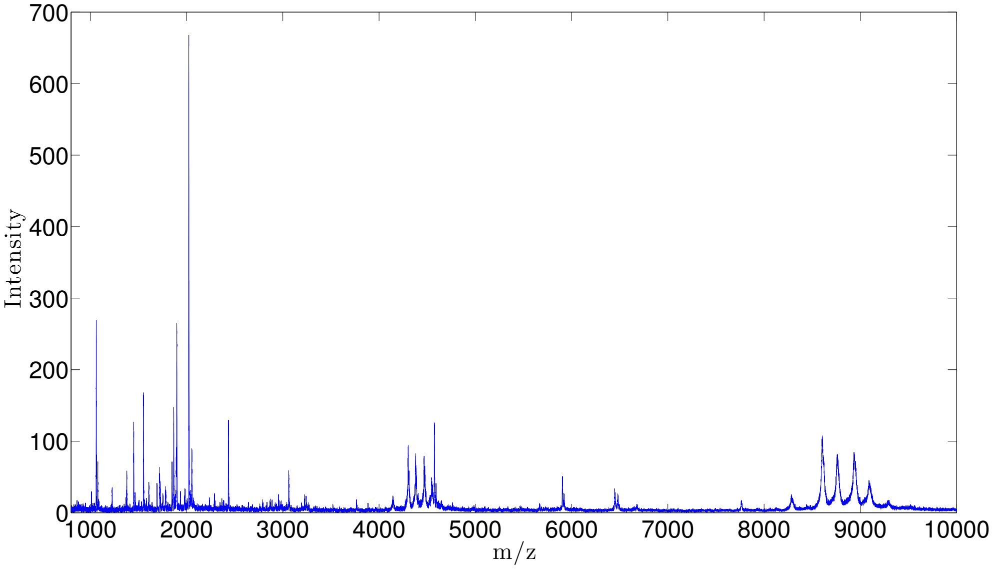


Figure 10.2: A raw mass spectrum of ECR

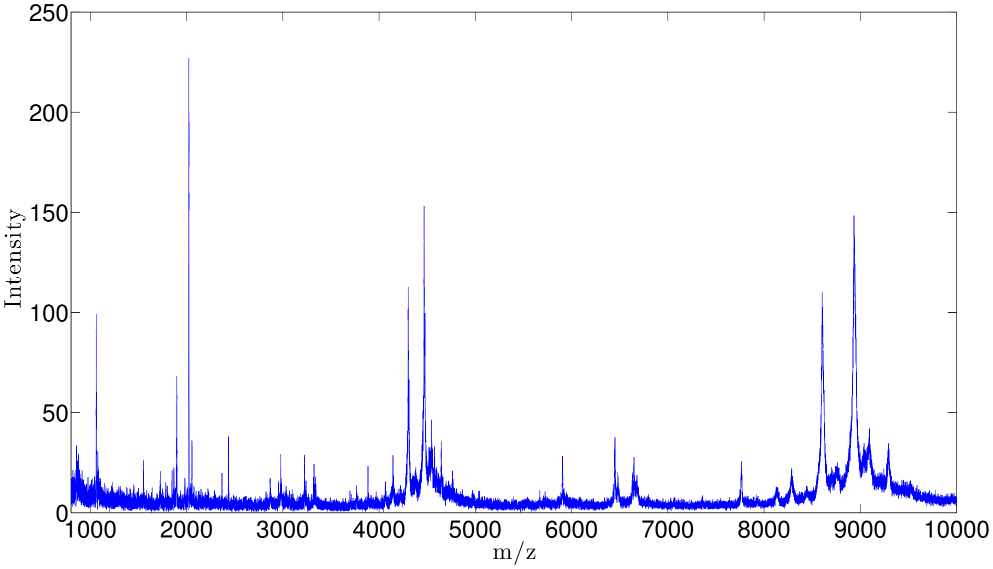


Figure 10.3: A raw mass spectrum of LCR

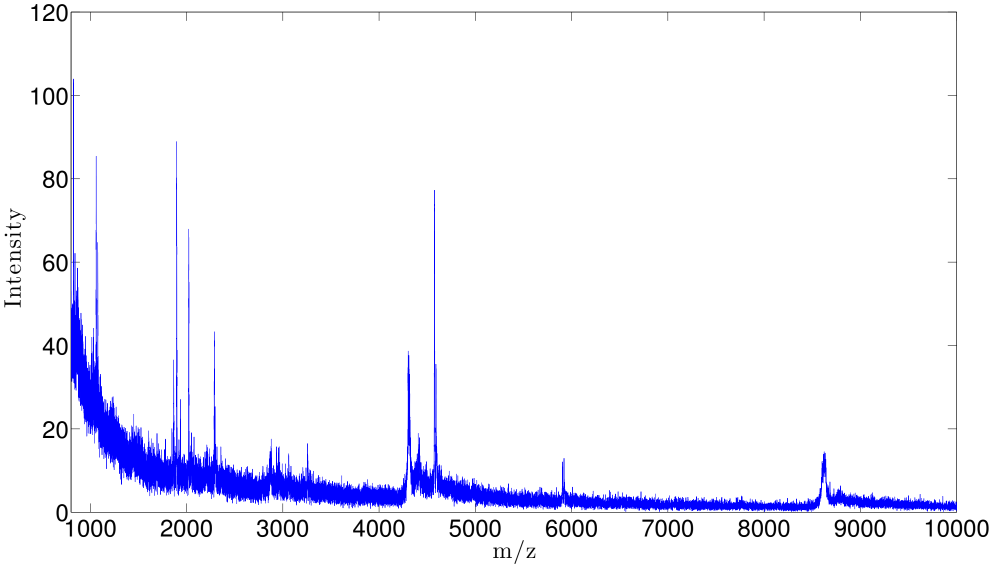


Figure 10.4: A raw mass spectrum of CTR

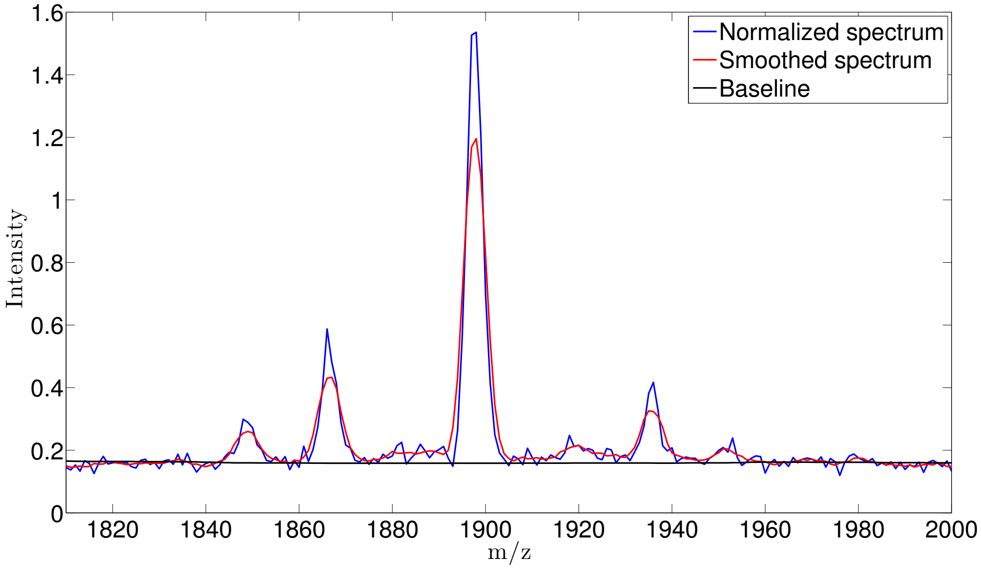


Figure 10.5: Pre-processing on a mass spectrum of CTR

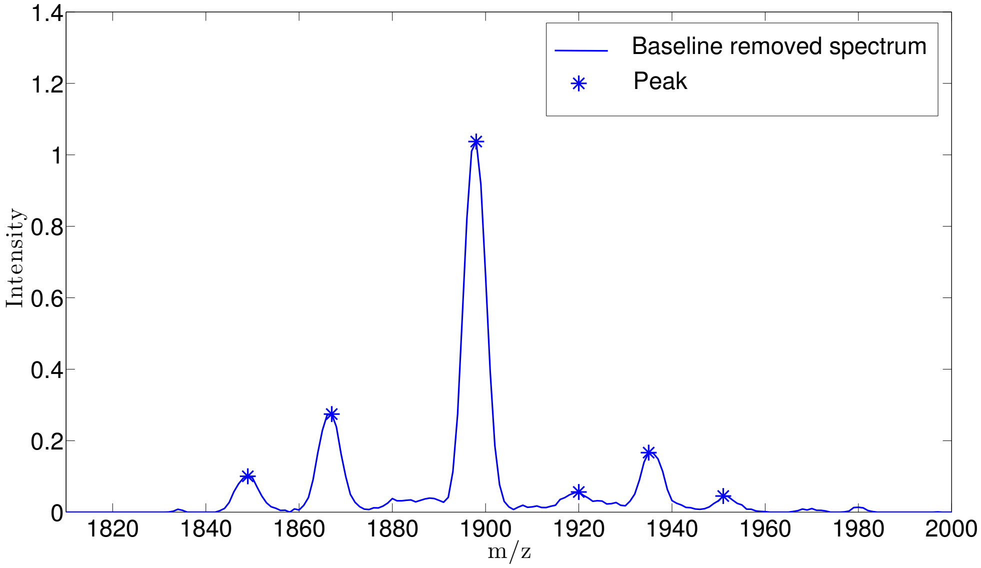


Figure 10.6: Detected peaks of a mass spectrum of CTR

effect of vertical variations. Figure 10.5 exhibits the normalized, smoothed, and baseline removed version of this mass spectrum. 6 peaks were detected on this piece of spectrum as shown in Figure 10.6. Each of the peaks has a stronger strength than 2.

Corresponding to the spectra in Figures 10.1, 10.2, 10.3, and 10.4, Figures 10.7, 10.8, 10.9, and 10.10 present the pre-processed spectra.

For the groups ADE, ECR, LCR, and CTR, the detected peaks per spectrum ranged between 250-374, 247-384, 230-375, and 219-318.

Two mass spectrum replicates from one subject are expected to have two similar sets of peaks. Figure 10.11 shows the detected peaks in two mass spectrum replicates from one subject of CTR between the m/z range of 4550 and 4650. Six peaks are detected on each of the two spectra. The two sets of peaks are not perfectly matched, but considering spectrum variations, it is reasonable to believe that they represent the same set of peptide particles. If one peak that is detected from one spectrum replicate is not present in other replicates of the same subject, this peak might be a noisy peak. Therefore, using more replicates for one subject minimizes the possibility of detecting false peaks.

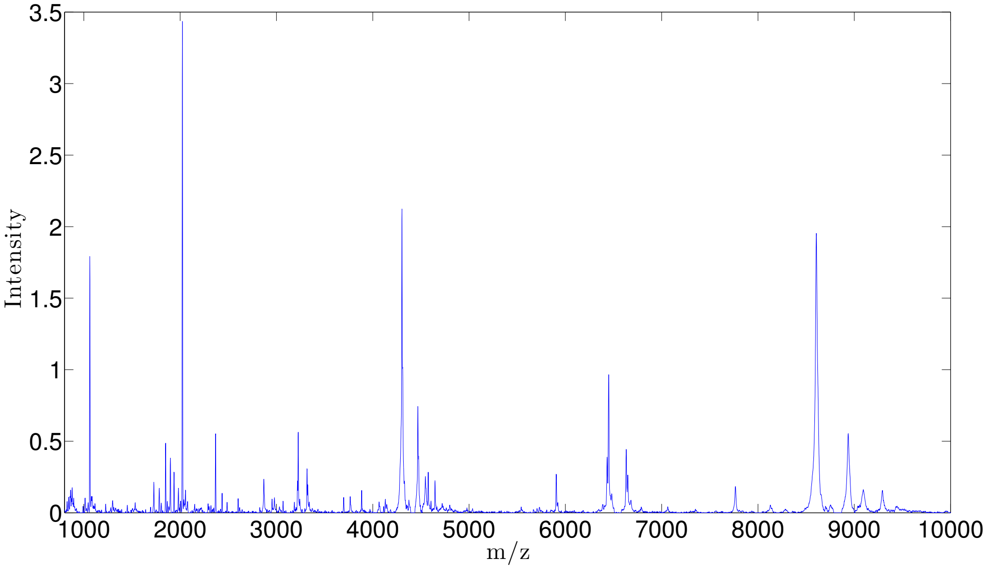


Figure 10.7: A pre-processed spectrum of ADE

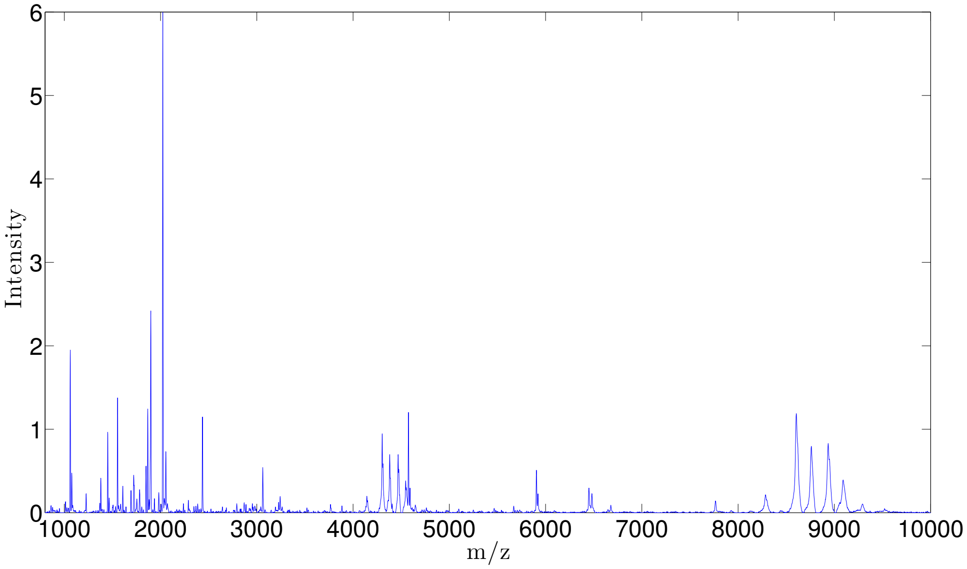


Figure 10.8: A pre-processed spectrum of ECR

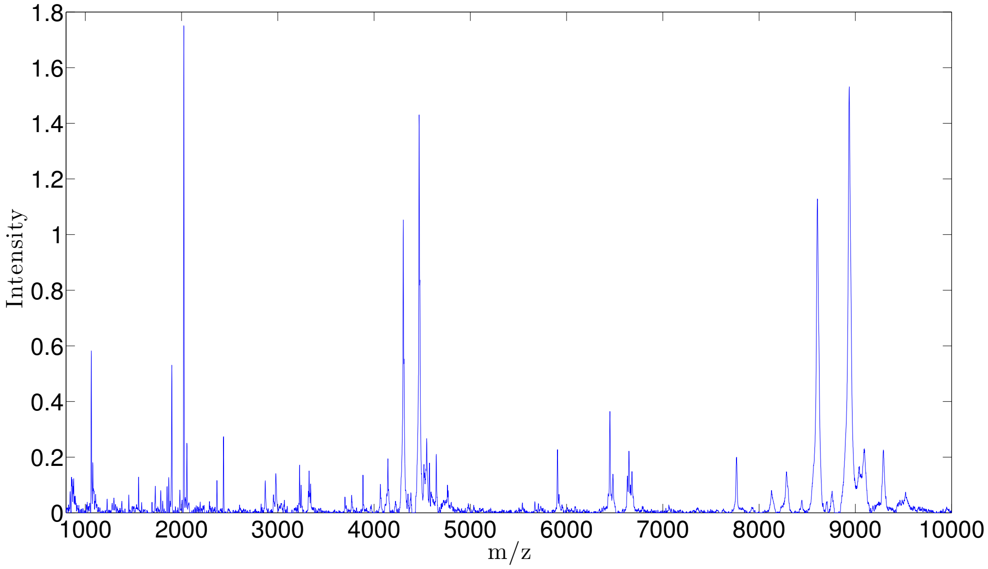


Figure 10.9: A pre-processed spectrum of LCR

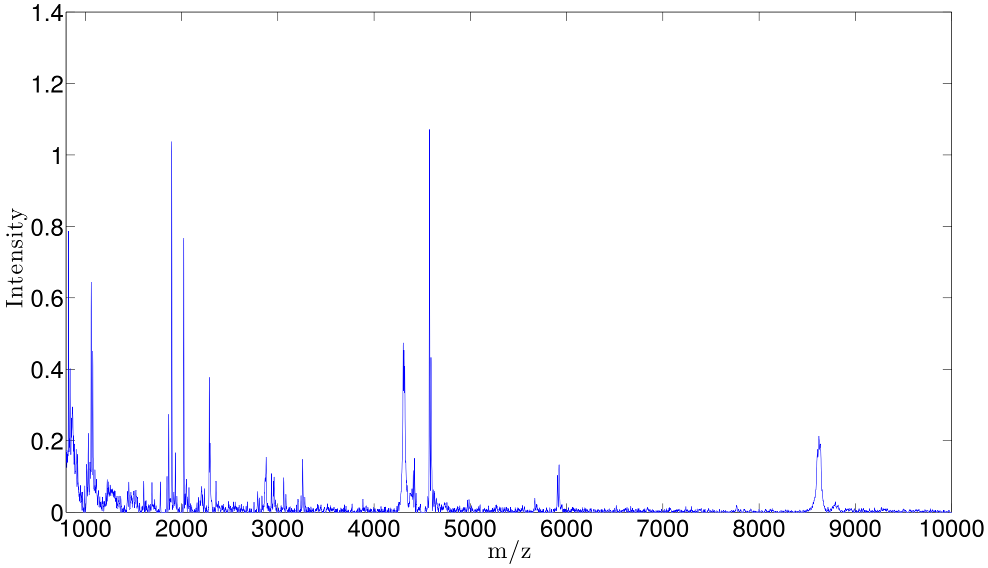


Figure 10.10: A pre-processed spectrum of CTR

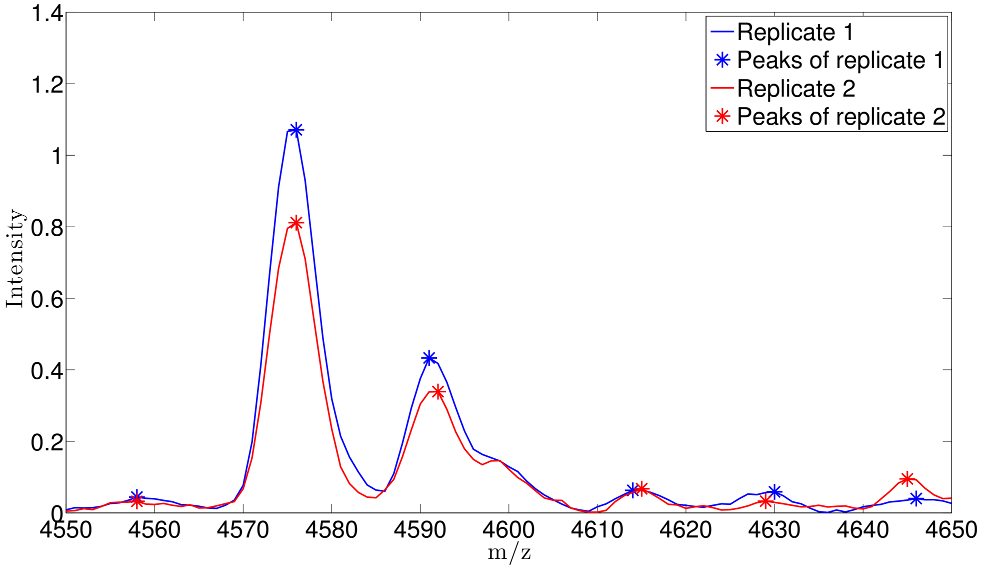


Figure 10.11: Detected peaks of two mass spectrum replicates

The 216 and 253 MALDI mass spectra in our ovarian datasets of 04/03/02 and 08/07/02 were restricted to m/z ratios from 0 to 20,000. A raw mass spectrum example from each of the groups OVC04, CTR04, OVC08 and CTR08 is displayed in a figure as an example (see Figures 10.12, 10.13, 10.14, and 10.15).

We implemented similarly the pre-processing procedure on each mass spectrum in OVC04, CTR04, OVC08, and CTR08. The spectrum smoothing, noise extraction window ratio, baseline computation window and the peak strength threshold are $u = 0.0003$, $v = 0.017$, $w = 0.025$, $t = 0.0005$, and $th = 2$ as for the previous dataset. Baseline removal was skipped on OVC04 and CTR04 because it had been performed before the dataset of 04/03/02 was posted.

For the groups OVC04, CTR04, OVC08, and CTR08, the detected peaks per spectrum ranged between 929-766, 737-954, 648-838, and 626-791.

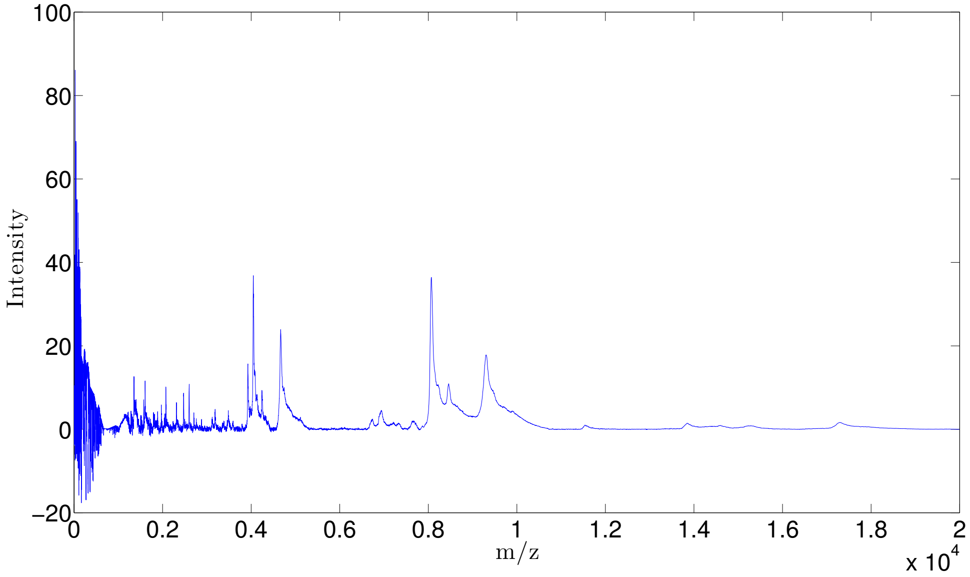


Figure 10.12: A raw mass spectrum of OVC04

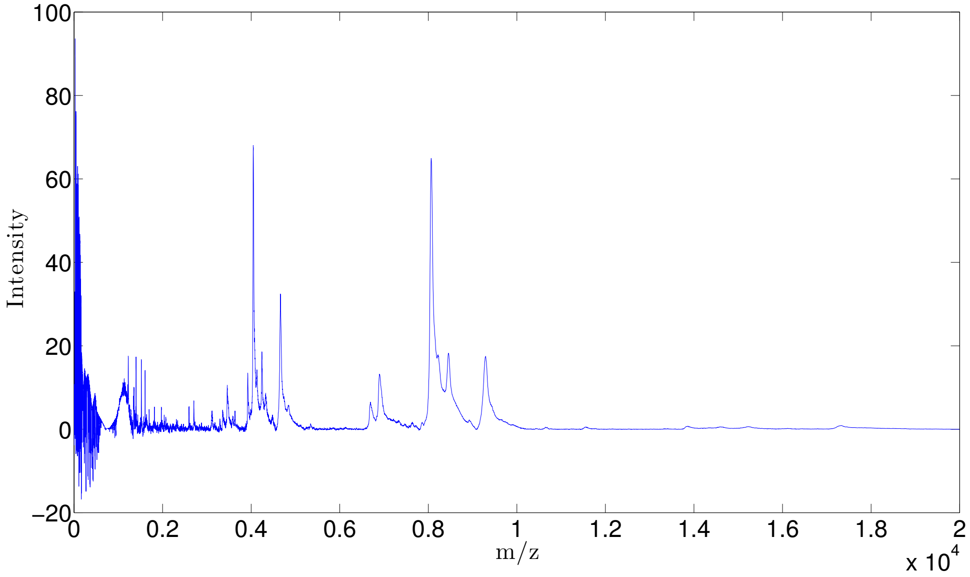


Figure 10.13: A raw mass spectrum of CTR04

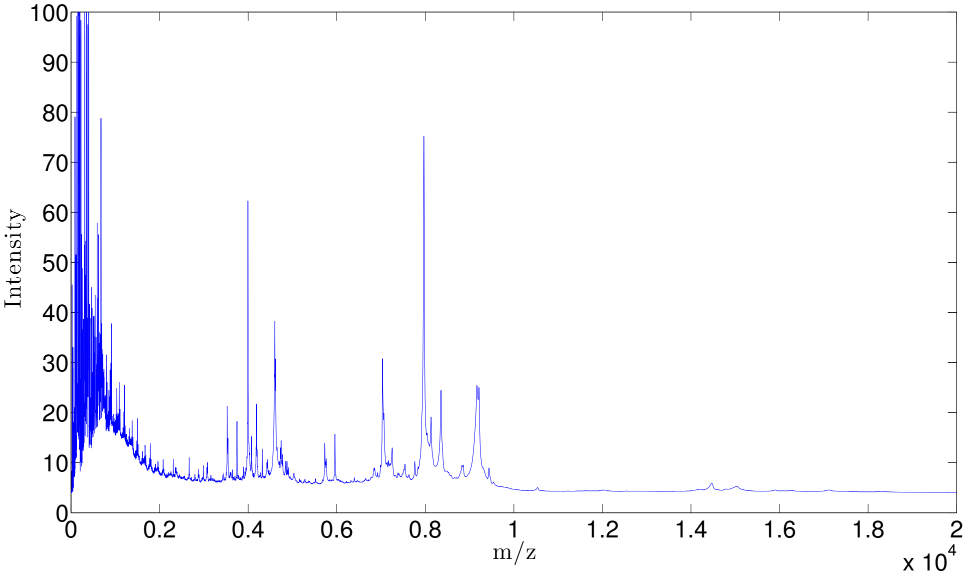


Figure 10.14: A raw mass spectrum of OVC08

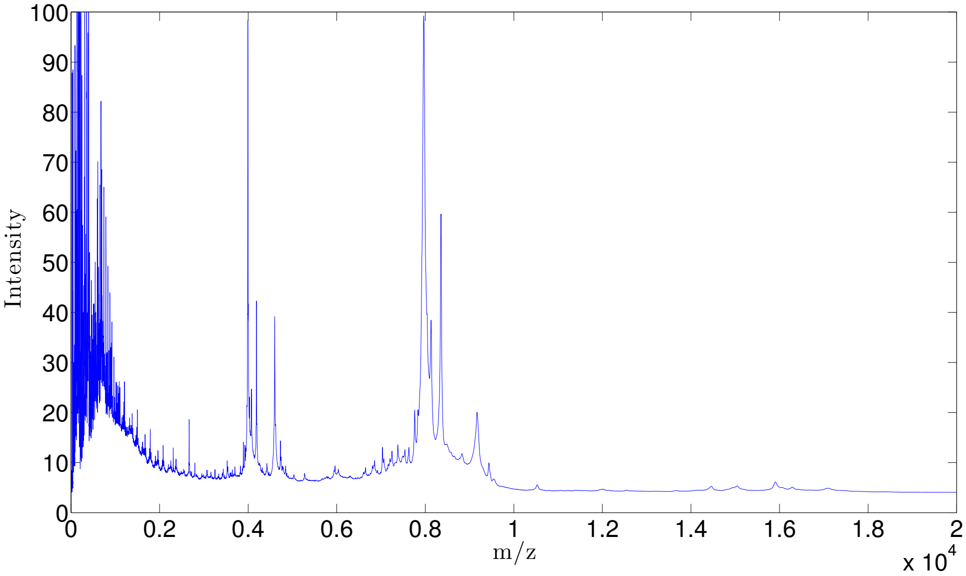


Figure 10.15: A raw mass spectrum of CTR08

10.2.2 Target Pools of Biomarkers

Both of the two signature discovery methods start with the selection of biomarker target pools. We present here the highly ranked biomarkers in each of the biomarker pools selected for each of the discrimination task before implementing signature discovery algorithm.

Given 0.3% as the accuracy of MALDI mass spectrometer, we focused on 844 reference peak abscissas $Ab_j = 800 \times 1.003^j$ for all 4 discriminating tasks in colorectal dataset, ranging from 800 to 10,000. Tables 10.1, 10.2, 10.3, and 10.4 list for the task ADE vs ECR, ADE vs LCR, ECR vs LCR, and CRC vs CTR, the top 10 ranked G^+ and G^- biomarkers based on the ratios of activation frequencies fq^+/fq^- and fq^-/fq^+ respectively. In general, the biomarkers in CRC vs CTR have larger fq^+/fq^- or fq^-/fq^+ ratios than those in the other three tasks. That is the reason why the discrimination task CRC vs CTR is easier than the other three.

ADE Biomarkers					
Ab	$f^{ADE}(Ab)$	$f^{ECR}(Ab)$	(Ab_i, Ab_j)	$f^{ADE}(Ab_i, Ab_j)$	$f^{ECR}(Ab_i, Ab_j)$
1146	0.45	0.18	(1565, 1146)	0.39	0.07
1192	0.31	0.12	(1368, 1192)	0.28	0.06
1845	0.88	0.55	(1851, 1146)	0.44	0.12
1579	0.55	0.30	(1392, 1146)	0.23	0.05
2475	0.58	0.32	(6397, 1146)	0.45	0.13
1397	0.29	0.14	(1579, 1146)	0.29	0.07
1188	0.33	0.16	(1368, 1188)	0.29	0.07
6377	0.77	0.50	(1372, 1192)	0.29	0.07
1851	0.90	0.62	(2475, 1579)	0.39	0.10
6397	0.74	0.49	(2475, 1845)	0.56	0.17
ECR Biomarkers					
Ab	$f^{ADE}(Ab)$	$f^{ECR}(Ab)$	(Ab_i, Ab_j)	$f^{ADE}(Ab_i, Ab_j)$	$f^{ECR}(Ab_i, Ab_j)$
8813	0.10	0.27	(6915, 4282)	0.12	0.38
7984	0.10	0.24	(3985, 4282)	0.15	0.43
2782	0.13	0.29	(1470, 8630)	0.17	0.45
4168	0.26	0.45	(3145, 3985)	0.13	0.39
4282	0.31	0.51	(7254, 8630)	0.20	0.51
3985	0.47	0.70	(1977, 1261)	0.09	0.27
6283	0.28	0.47	(6283, 4168)	0.10	0.31
2773	0.31	0.50	(6283, 3985)	0.13	0.38
1261	0.17	0.31	(3599, 3985)	0.15	0.40
1288	0.17	0.31	(918, 1470)	0.09	0.26

Table 10.1: Biomarkers for ADE vs ECR

This table lists the top ranked ADE biomarkers, which have largest f^{ADE}/f^{ECR} ratios and the top ranked ECR biomarkers, which have largest f^{ECR}/f^{ADE} ratios. These biomarkers contain both single-peak biomarkers Ab and double-peak biomarkers (Ab_i, Ab_j). f^{ADE} and f^{ECR} are activation frequencies computed empirically.

			ADE Biomarkers		
Ab	$f^{ADE}(Ab)$	$f^{LCR}(Ab)$	(Ab_i, Ab_j)	$f^{ADE}(Ab_i, Ab_j)$	$f^{LCR}(Ab_i, Ab_j)$
1146	0.45	0.09	(6377, 1146)	0.44	0.05
1143	0.28	0.08	(1845, 1146)	0.42	0.06
1150	0.31	0.11	(1116, 1146)	0.36	0.05
1192	0.31	0.12	(4180, 1146)	0.36	0.05
1570	0.48	0.23	(6731, 1146)	0.36	0.05
1343	0.34	0.15	(2001, 1146)	0.39	0.06
2001	0.61	0.33	(7542, 1146)	0.39	0.06
7542	0.69	0.39	(5187, 1146)	0.39	0.06
1112	0.31	0.14	(2007, 1146)	0.39	0.06
1136	0.31	0.14	(4193, 1146)	0.33	0.05
			LCR Biomarkers		
Ab	$f^{ADE}(Ab)$	$f^{LCR}(Ab)$	(Ab_i, Ab_j)	$f^{ADE}(Ab_i, Ab_j)$	$f^{LCR}(Ab_i, Ab_j)$
1812	0.17	0.38	(1296, 3145)	0.15	0.52
891	0.10	0.26	(8630, 3145)	0.17	0.54
3145	0.31	0.57	(7254, 8630)	0.20	0.61
8813	0.10	0.25	(3420, 8630)	0.20	0.61
8630	0.56	0.89	(819, 8630)	0.09	0.32
4168	0.26	0.49	(2651, 3145)	0.10	0.35
3844	0.13	0.29	(4081, 7587)	0.18	0.54
839	0.15	0.32	(4282, 8630)	0.15	0.46
1246	0.09	0.21	(5156, 4168)	0.09	0.31
7984	0.10	0.23	(1296, 1812)	0.10	0.34

Table 10.2: Biomarkers for ADE vs LCR

This table lists the top ranked ADE biomarkers, which have largest f^{ADE}/f^{LCR} ratios and the top ranked LCR biomarkers, which have largest f^{LCR}/f^{ADE} ratios. These biomarkers contain both single-peak biomarkers Ab and double-peak biomarkers (Ab_i, Ab_j). f^{ADE} and f^{LCR} are activation frequencies computed empirically.

ECR Biomarkers					
Ab	$f^{ECR}(Ab)$	$f^{LCR}(Ab)$	(Ab_i, Ab_j)	$f^{ECR}(Ab_i, Ab_j)$	$f^{LCR}(Ab_i, Ab_j)$
1953	0.21	0.06	(1570, 4180)	0.29	0.08
7542	0.67	0.39	(3524, 1288)	0.30	0.09
8376	0.55	0.31	(2589, 1112)	0.23	0.06
1112	0.29	0.14	(2998, 4180)	0.45	0.16
4180	0.65	0.39	(2998, 8376)	0.40	0.14
6320	0.43	0.25	(6301, 8401)	0.19	0.05
921	0.44	0.26	(3524, 921)	0.40	0.15
2589	0.66	0.44	(5187, 8376)	0.40	0.15
6894	0.61	0.42	(5187, 1953)	0.18	0.05
6915	0.68	0.48	(6711, 1112)	0.24	0.08
LCR Biomarkers					
Ab	$f^{ECR}(Ab)$	$f^{LCR}(Ab)$	(Ab_i, Ab_j)	$f^{ECR}(Ab_i, Ab_j)$	$f^{LCR}(Ab_i, Ab_j)$
7889	0.09	0.26	(1210, 2143)	0.05	0.25
2807	0.15	0.34	(3856, 2807)	0.07	0.28
2143	0.26	0.48	(1213, 2143)	0.05	0.21
3856	0.43	0.69	(1032, 3856)	0.06	0.23
1210	0.26	0.45	(9386, 1812)	0.09	0.32
1032	0.13	0.26	(3856, 7889)	0.05	0.20
1029	0.32	0.52	(9386, 5829)	0.07	0.26
819	0.17	0.32	(4684, 7889)	0.06	0.22
1213	0.26	0.43	(3080, 7889)	0.06	0.22
6226	0.34	0.52	(3164, 3856)	0.14	0.40

Table 10.3: Biomarkers for ECR vs LCR

This table lists the top ranked ECR biomarkers, which have largest f^{ECR}/f^{LCR} ratios and the top ranked LCR biomarkers, which have largest f^{LCR}/f^{ECR} ratios. These biomarkers contain both single-peak biomarkers Ab and double-peak biomarkers (Ab_i, Ab_j). f^{ECR} and f^{LCR} are activation frequencies computed empirically.

CRC Biomarkers					
Ab	$f^{CRC}(Ab)$	$f^{CTR}(Ab)$	(Ab_i, Ab_j)	$f^{CRC}(Ab_i, Ab_j)$	$f^{CTR}(Ab_i, Ab_j)$
4155	0.96	0.20	(1556,4155)	0.88	0.13
1802	0.62	0.13	(1556,3697)	0.84	0.13
3183	0.77	0.18	(2832, 4155)	0.84	0.13
3708	0.85	0.23	(5725, 3708)	0.84	0.13
3697	0.90	0.25	(1556, 3708)	0.80	0.13
1380	0.53	0.13	(5725, 8301)	0.79	0.13
8451	0.59	0.15	(4057, 3697)	0.79	0.13
8276	0.78	0.23	(5725, 3697)	0.89	0.15
1551	0.91	028	(3697, 4155)	0.88	0.15
1983	0.63	0.18	(4143, 3183)	0.77	0.13
CTR Biomarkers					
Ab	$f^{CRC}(Ab)$	$f^{CTR}(Ab)$	(Ab_i, Ab_j)	$f^{CRC}(Ab_i, Ab_j)$	$f^{CTR}(Ab_i, Ab_j)$
1439	0.09	0.85	(8813,805)	0.02	0.82
1657	0.04	0.45	(805,1439)	0.02	0.80
1202	0.05	0.52	(1243,2497)	0.02	0.80
1652	0.03	0.35	(2331,1439)	0.02	0.77
805	0.09	0.82	(2359,1439)	0.02	0.75
2359	0.09	0.77	(2497,805)	0.02	0.75
2497	0.11	0.80	(2331,805)	0.02	0.75
1435	0.12	0.80	(1243,805)	0.02	0.82
1246	0.12	0.82	(2338,805)	0.02	0.70
1662	0.08	0.57	(1246,1439)	0.02	0.80

Table 10.4: Biomarkers for CRC vs CTR

This table lists the top ranked CRC biomarkers, which have largest f^{CRC}/f^{CTR} ratios and the top ranked CTR biomarkers, which have largest f^{CTR}/f^{CRC} ratios. These biomarkers contain both single-peak biomarkers Ab and double-peak biomarkers (Ab_i, Ab_j). f^{CRC} and f^{CTR} are activation frequencies computed empirically.

Since all mass spectra in two SELDI datasets have already been aligned to a reference list of 15,154 peak abscissas, there is no need to construct a new list of reference peak abscissas. Thus, we took advantage of this consensus peak abscissa list and computed the activation frequencies for each of the abscissas by an alternative strategy: a strong peak detected in a mass spectrum M that located within $x \pm 0.1\% x$ range indicates the activation of abscissa x in M, given that the mass spectrometer accuracy is 0.1%.

Tables 10.5 and 10.6 list the top 10 ranked G^+ and G^- biomarkers of the discrimination tasks OVC04 vs CTR04 and OVC08 vs CTR08. In general, the biomarkers in OVC08 vs CTR08 have larger fq^+/fq^- or fq^-/fq^+ ratios than those in OVC04 vs CTR04. Therefore, OVC08 vs CTR08 is an easier discrimination task.

OVC04 Biomarkers					
Ab	$f^{OVC04}(Ab)$	$f^{CTR04}(Ab)$	(Ab_i, Ab_j)	$f^{OVC04}(Ab_i, Ab_j)$	$f^{CTR04}(Ab_i, Ab_j)$
328	0.32	0.03	(3249, 494)	0.57	0.07
582	0.27	0.04	(593, 547)	0.32	0.03
494	0.71	0.18	(383, 328)	0.31	0.03
276	0.28	0.06	(6883, 494)	0.40	0.05
2412	0.33	0.07	(1076, 494)	0.53	0.07
959	0.36	0.08	(607, 242)	0.27	0.03
2128	0.27	0.06	(855, 3249)	0.32	0.04
1576	0.35	0.09	(1598, 3249)	0.26	0.03
3249	0.77	0.24	(242, 1576)	0.26	0.03
1941	0.24	0.06	(1443, 3249)	0.34	0.05
CTR04 Biomarkers					
Ab	$f^{OVC04}(Ab)$	$f^{CTR04}(Ab)$	(Ab_i, Ab_j)	$f^{OVC04}(Ab_i, Ab_j)$	$f^{CTR04}(Ab_i, Ab_j)$
649	0.04	0.37	(594, 678)	0.04	0.36
1246	0.05	0.37	(678, 661)	0.04	0.36
661	0.06	0.42	(2139, 661)	0.04	0.33
1468	0.06	0.40	(661, 649)	0.04	0.31
1569	0.04	0.29	(2336, 678)	0.05	0.36
1540	0.06	0.38	(824, 661)	0.04	0.30
1144	0.05	0.32	(784, 1144)	0.04	0.28
1083	0.04	0.25	(2987, 661)	0.04	0.28
377	0.05	0.29	(1098, 1144)	0.04	0.27
1292	0.05	0.28	(970, 661)	0.04	0.27

Table 10.5: Biomarkers for OVC04 vs CTR04

This table lists the top ranked OVC04 biomarkers, which have largest f^{OVC04}/f^{CTR04} ratios and the top ranked CTR04 biomarkers, which have largest f^{CTR04}/f^{OVC04} ratios. These biomarkers contain both single-peak biomarkers Ab and double-peak biomarkers (Ab_i, Ab_j). f^{OVC04} and f^{CTR04} are activation frequencies computed empirically.

OVC08 Biomarkers					
Ab	$f^{OVC08}(Ab)$	$f^{CTR08}(Ab)$	(Ab_i, Ab_j)	$f^{OVC08}(Ab_i, Ab_j)$	$f^{CTR08}(Ab_i, Ab_j)$
435	0.74	0.04	(546,435)	0.70	0.04
442	0.72	0.06	(647,435)	0.62	0.04
828	0.71	0.06	(442,435)	0.59	0.04
647	0.82	0.09	(828,442)	0.55	0.04
10539	0.43	0.04	(1146,828)	0.55	0.04
668	0.41	0.04	(870,435)	0.52	0.04
1336	0.44	0.05	(5283,647)	0.47	0.04
2450	0.50	0.06	(693,442)	0.46	0.04
762	0.45	0.06	(13016,442)	0.46	0.04
227	0.45	0.07	(762,435)	0.41	0.04
CTR08 Biomarkers					
Ab	$f^{OVC08}(Ab)$	$f^{CTR08}(Ab)$	(Ab_i, Ab_j)	$f^{OVC08}(Ab_i, Ab_j)$	$f^{CTR08}(Ab_i, Ab_j)$
667	0.02	0.64	(544,667)	0.02	0.60
681	0.02	0.54	(650,667)	0.02	0.56
555	0.02	0.44	(831,544)	0.03	0.65
1115	0.03	0.44	(4001,667)	0.02	0.54
695	0.04	0.49	(681,667)	0.02	0.48
4001	0.05	0.67	(3681,650)	0.02	0.48
650	0.06	0.70	(840,831)	0.03	0.57
1333	0.04	0.47	(877,650)	0.02	0.42
877	0.04	0.46	(695,667)	0.02	0.41
463	0.02	0.30	(15564,667)	0.02	0.40

Table 10.6: Biomarkers for OVC08 vs CTR08

This table lists the top ranked OVC08 biomarkers, which have largest f^{OVC08}/f^{CTR08} ratios and the top ranked CTR08 biomarkers, which have largest f^{CTR08}/f^{OVC08} ratios. These biomarkers contain both single-peak biomarkers Ab and double-peak biomarkers (Ab_i, Ab_j). f^{OVC08} and f^{CTR08} are activation frequencies computed empirically.

10.2.3 Kullback-Leibler Distances

Kulback-Leibler distance (6.7) measures the distance between two Gibbs distributions; therefore, it evaluates the separability of two groups (see Section 6.4). Quantifying the separability of two groups before implementing the construction of a good classifier is an interesting and useful step for any classification problem. However, there is a delicate question about how to perform a fair comparison because the Kullback-Leibler distance can be largely affected by the dimension of the parameter vector. The larger the dimension of the two parameters $\boldsymbol{\theta}^+$ and $\boldsymbol{\theta}^-$ are, the larger the distance tends to be.

In our study, we have computed the Kulback-Leibler distance for the tasks ADE vs ECR, ADE vs LCR, ECR vs LCR, and CRC vs CTR. In each task G^+ vs G^-, we constructed two separate Gibbs models on G^+ and G^- with the 3 top ranked G^+ biomarkers and 3 top ranked G^- biomarkers of single peaks. We consider in addition each pair of G^+ biomarkers and each pair of G^- biomarkers as a clique of cardinal 2. The dimension of all the parameter vectors is then 12 is then 12. We computed the Kulback-Leibler distance of the two Gibbs models by (6.7). The distances between ADE and ECR, ADE and LCR, and ECR and LCR were quite similar, respectively equaling 0.68, 0.79, and 0.69. The distance between CRC vs CTR was much larger, equaling 2.07. This predicts that discrimination between CRC vs CTR is a much easier classification task than the other three. This conclusion is compatible with the results we have got from the following signature discovery algorithms.

10.2.4 Signature Discoveries

10.2.4.1 Signature Discovery by Robust Log-likelihood (RLL)

For each benchmark discrimination task ADE vs ECR, ADE vs LCR, ECR vs LCR, CRC vs CTR, OVC04 vs CTR04, and OVC08 vs CTR08, we successively extracted target pools $TP(4k)$ of $4k$ highly discriminating biomarkers, with $4k = 4, 8, \cdots, 80$. For each benchmark task and each k, we implemented our RLL signature discovery within $TP(4k)$ and computed the discriminating power of this signature based classifier. The discriminating power of each signature-based classifier reached its plateau at $4k = 48, 72, 64, 8, 48$, and 28 for the tasks ADE vs ECR, ADE vs LCR, ECR vs LCR, CRC vs CTR, OVC04 vs CTR04, and OVC08 vs CTR08. Table 10.7, 10.8, 10.9, 10.10, 10.11 and 10.12 list the optimal signatures obtained for each of the tasks. Each signature is presented by two columns, the left and right column respectively listing the G+ and G- biomarkers G^+ and G^- biomarkers. The corresponding weights of all of all the biomarkers in a classifier are listed as well. Under each signature, we have also reported its reliability score $R(Sig)$ computed by the technique of Section 7.5.

10.2.4.2 Signature Discovery by Maximizing Detecting Power (MDP)

We successively implement our signature discovery by MDP described in Section 9.1.1 with target pools $TP(4k)$ of $4k = 4, 8, \cdots, 48$ highly discriminating biomarkers on discrimination tasks CRC vs CTR, ADE vs ECR, ADE vs LCR, ECR vs LCR, OVC04 vs CTR04, and OVC08 vs CTR08. For each benchmark task and each k,

ADE		ECR	
biomarker	weight	biomarker	weight
1188	0.24	1288	-0.36
1579	0.39	2782	-0.27
1851	0.20	3985	-0.07
2475	0.24	6283	-0.49
5064	0.37	7984	-0.13
(1368, 1188)	0.24	8630	-0.28
(1368, 1192)	0.24	8813	-0.97
(1372, 1192)	0.24	(855, 4560)	-0.23
(1372, 3675)	1.33	(855, 7254)	-0.60
(1392, 1146)	0.42	(2651, 870)	-0.39
(1565, 1146)	0.79	(2651, 873)	-0.39
(1851, 1146)	0.15	(3145, 3985)	-0.50
(2001, 1146)	0.74	(3420, 918)	-0.22
(2475, 1579)	0.30	(6283, 4168)	-0.39
(2475, 1845)	0.18	(6915, 4282)	-0.26
(6397, 1146)	0.12		
$R(Sig) = 0.6$			

Table 10.7: Signature of ADE vs ECR by RLL

ADE biomarker and ECR biomarkers of the signature are displayed separately with their corresponding weights. The signature reliability score $R(Sig)$ is also computed. The full list of reference peak abscissas included in the signature is: 855, 870, 873, 918, 1188, 1192, 1146, 1288, 1368, 1372, 1392, 1565, 1579, 1845, 1851, 2001, 2475, 2651, 2782, 3145, 3420, 3675, 3985, 4168, 4282, 4560, 5064, 6283, 6397, 6915, 7254, 7984, 8630, 8813.

ADE		LCR	
biomarker	weight	biomarker	weight
1112	0.26	891	-0.11
1116	0.06	2651	-0.20
1143	0.29	3844	-0.22
1146	0.09	7587	-0.52
1192	0.40	8630	-0.52
1570	0.07	8813	-0.78
2000	0.04	(817, 8630)	-0.42
2007	0.10	(819, 8630)	-0.42
4193	0.87	(873, 1029)	-0.28
5187	0.53	(1089, 3844)	-0.22
8032	0.10	(2296, 3145)	-0.49
(1116, 1146)	0.14	(3420, 8630)	-0.38
(1845, 1146)	0.01	(3985, 3145)	-0.05
(1851, 1146)	0.09	(4081, 7587)	-0.18
(2001, 1146)	0.01	(6264, 8630)	-0.32
(2007, 1146)	0.01	(7254, 8630)	-0.03
(2892, 1146)	0.14		
(3524, 1146)	0.22		
(4180, 1146)	0.14		
(4193, 1146)	0.14		
(5187, 1146)	0.01		
(5202, 1146)	0.01		
(6377, 1146)	0.22		
(6397, 1146)	0.22		
(6731, 1146)	0.22		
(7386, 1146)	0.09		
(7542, 1146)	0.22		
(8104, 1146)	0.01		
(8376, 1146)	0.22		
$R(Sig) = 0.6$			

Table 10.8: Signature of ADE vs LCR by RLL

ADE biomarker and LCR biomarkers of the signature are displayed separately with their corresponding weights. The signature reliability score $R(Sig)$ is also computed. The full list of reference peak abscissas included in the signature is: 817, 819, 873, 891, 1029, 1089, 1112, 1116, 1143, 1146, 1192, 1570, 1845, 1851, 2000, 2001, 2007, 2296, 2651, 2892, 3145, 3420, 3524, 3844, 3985, 4081, 4180, 4193, 4180, 4193, 5187, 5202, 6264, 6377, 6397, 6731, 7254, 7386, 7254, 7542, 7587, 8032, 8104, 8376, 8630, 8813.

ECR		LCR	
biomarker	weight	biomarker	weight
921	0.25	819	-0.46
2558	0.61	1210	-0.13
2589	0.24	1213	-0.36
6894	0.30	2143	-0.38
6915	0.20	2331	-0.40
7542	0.61	2628	-0.60
8401	0.33	2807	-0.24
(1570, 4180)	0.67	4944	-0.57
(2589, 1112)	0.59	5829	-0.26
(2643, 1953)	0.43	6226	-0.23
(2840, 1953)	0.43	(1032, 3856)	-0.49
(2998, 4180)	0.25	(1862, 1032)	-0.49
(2998, 8376)	0.61	(2143, 7889)	-0.57
(3524, 1288)	0.10	(3080, 7889)	-0.57
(3524, 921)	1.01	(3379, 5590)	-0.84
(5187, 1953)	0.43	(3400, 1478)	-0.82
(5202, 1953)	0.43	(3493, 2143)	-0.12
(5265, 6320)	0.44	(3856, 2143)	-0.18
(6301, 8401)	0.42	(3856, 7889)	-0.08
(6711, 1112)	0.10	(4684, 7889)	-0.95
(9386, 1812)	-0.34		
(9386, 5829)	-0.11		
$R(Sig) = 0.5$			

Table 10.9: Signature of ECR vs LCR by RLL

ECR biomarker and LCR biomarkers of the signature are displayed separately with their corresponding weights. The signature reliability score $R(Sig)$ is also computed. The full list of reference peak abscissas included in the signature is: 819, 921, 1032, 1112,1210, 1213, 1288, 1478, 1570, 1812, 1862, 1953, 2143, 2331, 2558, 2589, 2628, 2643, 2807, 2840, 2998, 3080, 3379, 3400, 3493, 3524, 3856, 4180, 4684, 4944, 5187, 5202, 5265, 5590, 5829, 6226, 6301, 6320, 6711, 6894, 6915, 7542, 7889, 8376, 8401, 9386.

CRC		CTR	
biomarker	weight	biomarker	weight
1802	0.11	1439	-0.44
4155	0.27	1657	-0.79
(3183, 4155)	0.52	(805, 1439)	-0.50
(3697, 4155)	0.11	(2359, 1439)	-0.24
$R(Sig) = 0.9$			

Table 10.10: Signature of CRC vs CTR by RLL
CRC biomarker and CTR biomarkers of the signature are displayed separately with their corresponding weights. The signature reliability score $R(Sig)$ is also computed. The full list of reference peak abscissas included in the signature is: 805, 1439, 1802, 1657, 2359, 3183, 3697, 4155.

we computed the optimized signature and the discriminating power of this signature based classifier.

Signature based discrimination between cancerous and control group (CRC vs CTR) reached perfect performance level 100% for $4k = 8$. For cancer stages discrimination tasks ADE vs ECR, ADE vs LCR, ECR vs LCR, OVC04 vs CTR04, and OVC08 vs CTR08, our signature-based classifiers reached their respective performance plateaus for $4k = 36, 28, 28, 40$, and 28.

We display in Figure 10.16 the maximization of ADE detecting power $J^{+}(Sig) = J^{ADE}(Sig)$ by a simulated annealing search with $200 \times 4k = 7200$ temperature cooling steps. The optimal list of biomarkers Sig^{ADE} achieves correct classification of single spectrum replicates with sensitivity 90.7% and specificity 98.7%, yielding an ADE detecting power of 94.7%. The second optimized list of biomarkers Sig^{ECR} achieves an ECR detecting power of 94.7% as well. Figure 10.17 displays the maximization of ECR detecting power $J^{-}(Sig) = J^{ECR}(Sig)$. The optimal ECR detecting power

OVC04		CTR04	
biomarker	weight	biomarker	weight
328	0.31	377	-0.76
494	0.01	649	-0.13
547	0.21	661	-0.32
582	0.85	902	-0.68
1598	0.33	1083	-0.32
1941	0.11	1246	-0.54
2128	0.25	1262	-0.74
2412	0.49	1292	-0.43
(383, 328)	0.31	1468	-0.51
(593, 547)	0.91	1569	-0.21
(607, 242)	0.31	(1934, 678)	-0.58
(825, 494)	0.60	(2139, 661)	-0.13
(855, 3249)	0.07	(2336, 678)	-0.58
(915, 383)	0.74	(2987, 661)	-0.13
(1598, 3249)	0.33	(9288, 678)	-0.58
(1856, 547)	0.47		
(2455, 3249)	0.54		
(6883, 494)	0.90		
$R(Sig) = 0.7$			

Table 10.11: Signature of OVC04 vs CTR04 by RLL
OVC04 biomarker and CTR04 biomarkers of the signature are displayed separately with their corresponding weights. The signature reliability score $R(Sig)$ is also computed. The full list of reference peak abscissas included in the signature is: 242, 328, 377, 383, 494, 547, 582, 593, 607, 649, 661, 678, 825, 855, 902, 915, 1083, 1246, 1262, 1292, 1468, 1569, 1598, 1856, 1934, 1941, 2128, 2139, 2336, 2412, 2455, 2987, 3249, 6883, 9288.

OVC08		OVC08	
biomarker	weight	biomarker	weight
435	0.09	667	-0.04
442	0.71	681	-0.46
828	0.08	555	-0.89
647	0.33	1115	-0.11
10539	0.07	695	-0.73
668	0.37	4001	-0.09
1336	0.45	650	-0.48
(647, 435)	0.07	(650, 667)	-0.03
(442, 435)	0.09	(15564, 667)	-0.03
(828, 442)	0.07		
(1146, 828)	0.09		
(5283, 647)	0.07		
(762, 435)	0.07		
(1336, 435)	0.07		
$R(Sig) = 0.8$			

Table 10.12: Signature of OVC08 vs CTR08 by RLL

OVC08 biomarker and CTR08 biomarkers of the signature are displayed separately with their corresponding weights. The signature reliability score $R(Sig)$ is also computed. The full list of reference peak abscissas included in the signature is: 435, 442, 555, 647, 650, 667, 668, 681, 695, 762, 828, 1115, 1146, 1336, 4001, 5283 10539, 15564.

is achieved at around 4300 steps. Another example of such optimization step can be found in Figure 10.18 and 10.19, where the maximization of $OVC04$ detecting power $J^{OVC04}(Sig)$ and $J^{CTR04}(Sig)$ are displayed.

Tables 10.13, 10.14, 10.15, 10.16, 10.17, and 10.18 exhibit the optimal signature obtained for CRC vs CTR, ADE vs ECR, ADE vs LCR, ECR vs LCR, OVC04 vs CTR04, and OVC08 vs CTR08 through 100 such simulated annealing each and their reliability scores underneath.

ADE biomarker	ECR biomarker
1397	1261
1845	2773
1851	2782
2475	4168
6377	6283
(1368, 1188)	8813
(1368, 1192)	(918, 1470)
(1372, 1192)	(1470, 8630)
(1565, 1146)	(3145, 3985)
(1579, 1146)	(3610, 4168)
(2475, 1579)	(3985, 4282)
	(6283, 3985)
	(6283, 4168)
	(7254, 8630)
$R(Sig) = 0.6$	

Table 10.13: Signature of ADE vs ECR by MDP
ADE biomarker and ECR biomarkers of the signature are displayed separately with their corresponding weights. The signature reliability score $R(Sig)$ is also computed. The full list of reference peak abscissas included in the signature is: 918, 1146, 1188, 1192, 1261, 1368, 1372, 1397, 1470, 1565, 1579, 1845, 1851,2475, 2773, 2782, 3145, 3610, 3985, 4168, 4282, 6283, 6377, 7254, 8630, 8813.

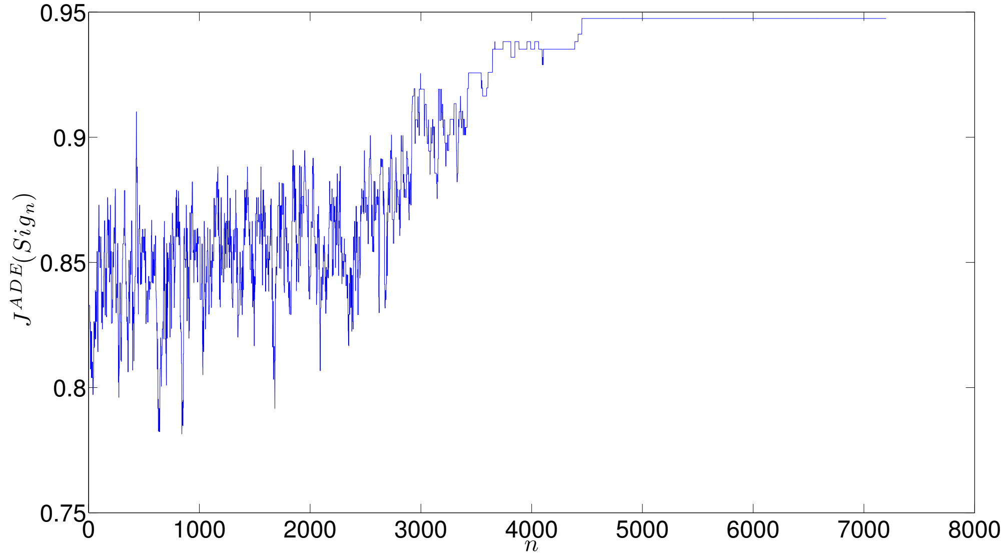


Figure 10.16: ADE vs ECR: simulated annealing search for Sig^{ADE}

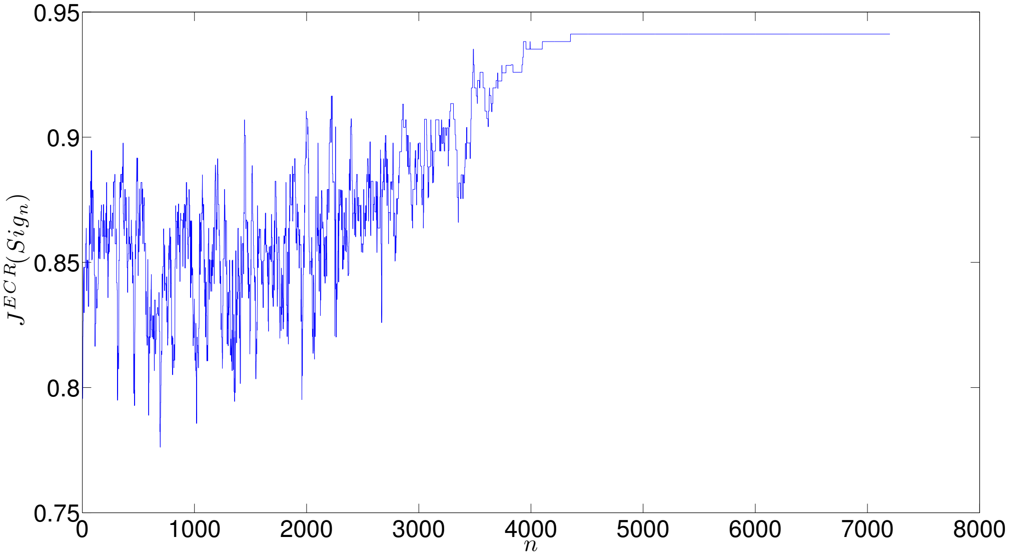


Figure 10.17: ADE vs ECR: simulated annealing search for Sig^{ECR}

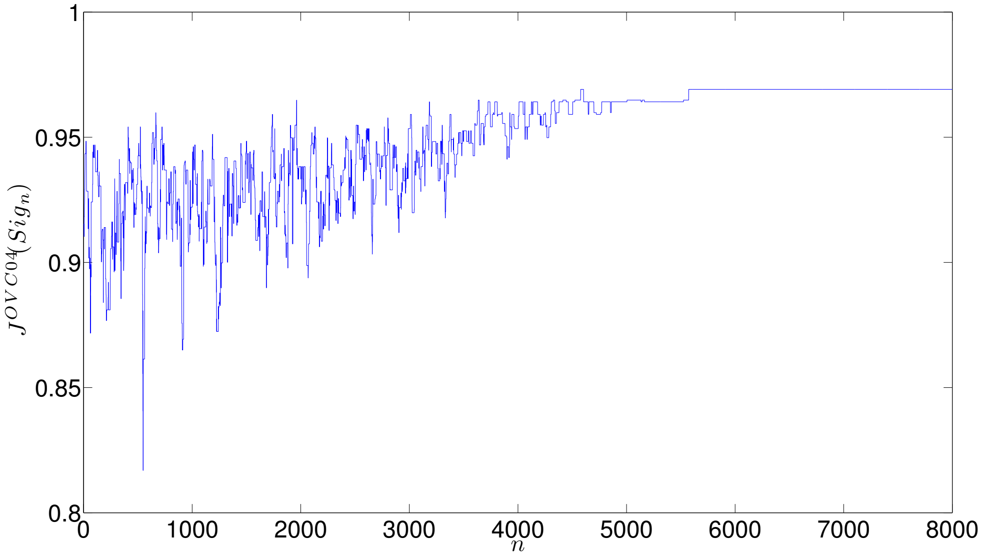


Figure 10.18: OVC04 vs CTR04: simulated annealing search for Sig^{OVC04}

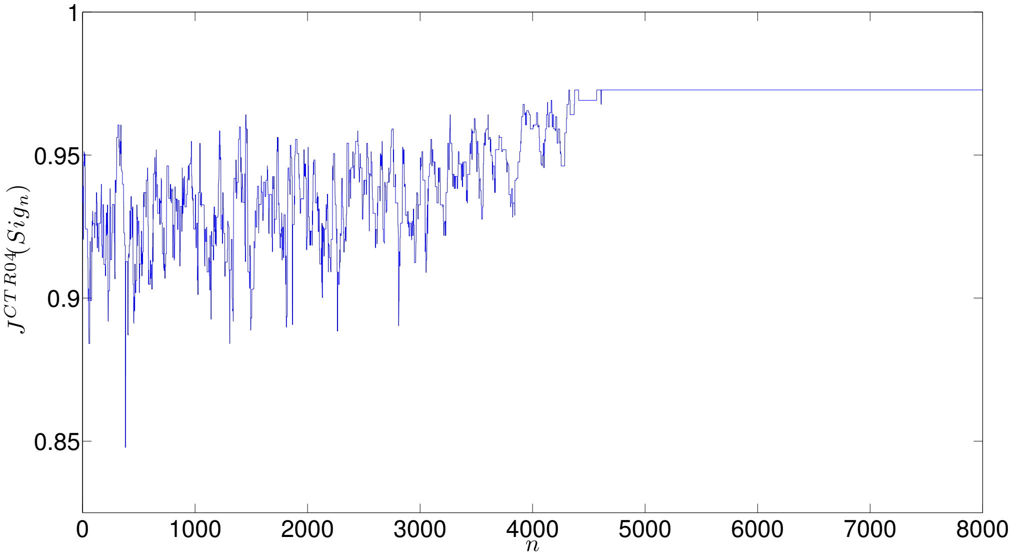


Figure 10.19: OVC04 vs CTR04: simulated annealing search for Sig^{CTR04}

ADE biomarker	LCR biomarker
1143	891
1146	3844
1192	4168
(1116, 1146)	8630
(2001, 1146)	8813
(4180, 1146)	(819, 8630)
(5187, 1146)	(2651, 3145)
(6377, 1146)	(3420, 8630)
(7542, 1146)	(4081, 7587)
	(4269, 7254)
	(7254, 8630)
	(8630, 3145)
$R(Sig) = 0.7$	

Table 10.14: Signature of ADE vs LCR by MDP
ADE biomarker and LCR biomarkers of the signature are displayed separately with their corresponding weights. The signature reliability score $R(Sig)$ is also computed. The full list of reference peak abscissas included in the signature is: 891, 1116, 1143, 1146, 1192, 2001, 2651, 3145, 3420, 3844, 4081, 4168, 4180, 4269, 5187, 6377, 7254, 7542, 7587, 8630, 8813.

ECR biomarker	LCR biomarker
921	1210
1112	2143
1953	2807
6320	7889
7524	(1032, 3856)
8376	(1210, 2143)
(921, 7542)	(3080, 7889)
(1570, 4180)	(3856, 2807)
(2998, 4180)	(3856, 7889)
(2998, 8376)	(4684, 7889)
(6301, 8401)	
(6320, 7542)	
$R(Sig) = 0.6$	

Table 10.15: Signature of ECR vs LCR by MDP
ECR biomarker and LCR biomarkers of the signature are displayed separately with their corresponding weights. The signature reliability score $R(Sig)$ is also computed. The full list of reference peak abscissas included in the signature is: 921, 1032, 1112, 1210, 1570, 1953, 2143, 2807, 2998, 3080, 4684, 3856, 4180, 6301, 6320, 7524, 7542, 7889, 8376, 8401.

CRC	CTR
1802	1439
4155	1657
(3183, 4155)	(805, 1439)
(3697, 4155)	(2359, 1439)
$R(Sig) = 0.9$	

Table 10.16: Signature of CRC vs CTR by MDP
CRC biomarker and CTR biomarkers of the signature are displayed separately with their corresponding weights. The signature reliability score $R(Sig)$ is also computed. The full list of reference peak abscissas included in the signature is: 805, 1802, 1439, 1657, 2359, 3183, 3697, 4155.

OVC04	CTR04
276	377
328	649
494	661
583	1083
1576	1144
1941	1246
2128	1292
2412	1468
3249	1540
(242, 1576)	1569
(383, 328)	(594, 678)
(593, 547)	(661, 649)
(607, 242)	(678, 661)
(855, 3249)	(784, 1144)
(1076, 494)	(824, 661)
(1444, 3249)	(970, 661)
(1598, 3249)	(1098, 1144)
(3249, 494)	(2139, 661)
(6883, 494)	(2336, 678)
	(2987, 661)
$R(Sig) = 0.7$	

Table 10.17: Signature of OVC04 vs CTR04 by MDP
OVC04 biomarker and CTR04 biomarkers of the signature are displayed separately with their corresponding weights. The signature reliability score $R(Sig)$ is also computed. The full list of reference peak abscissas included in the signature is: 242, 276, 328, 377, 383, 494, 547, 583, 593, 594, 607, 649, 661, 678, 784, 824, 855, 970, 1076, 1083, 1098, 1144, 1246, 1292, 1144, 1444, 1468, 1540, 1569, 1576, 1598, 1941, 2128, 2139, 2336, 2412, 2987, 3249, 6883.

OVC08	CTR08
435	555
442	650
647	681
668	695
828	1115
1336	(650, 667)
10539	(681, 667)
(442, 435)	(695, 667)
(646, 435)	(877, 650)
(762, 435)	(4001, 667)
(1146, 828)	(13329, 667)
(1336, 435)	(15564, 667)
(5283, 647)	
$R(Sig) = 0.8$	

Table 10.18: Signature of OVC08 vs CTR08 by MDP
OVC08 biomarker and CTR08 biomarkers of the signature are displayed separately with their corresponding weights. The signature reliability score $R(Sig)$ is also computed. The full list of reference peak abscissas included in the signature is: 435, 442, 555, 646, 647, 650, 667, 668, 681, 695, 762, 828, 877, 1115, 1146, 1336, 4001, 5283, 10539, 13329, 15564.

10.2.5 Group Classification

With a signature obtained by RL, any new subject is classified by the optimal classifier which generates this signature.

With a signature obtained by MDP, each subject has an G^+ score s^+ and an G^- score s^-. The signature-based classifier linearly separate to classify all the (s^+, s^-) points. In each of the Figures 10.20, 10.21, 10.22, 10.23, 10.24, and 10.25, the 67, 64, 77, 119, 216, and 253 patients composing ADE and ECR, ADE and LCR, ECR and LCR, CRC and CTR, OVC04 and CTR04, and OVC08 and CTR08 are displayed as planar points (s^+, s^-), which are separated into two groups by a line.

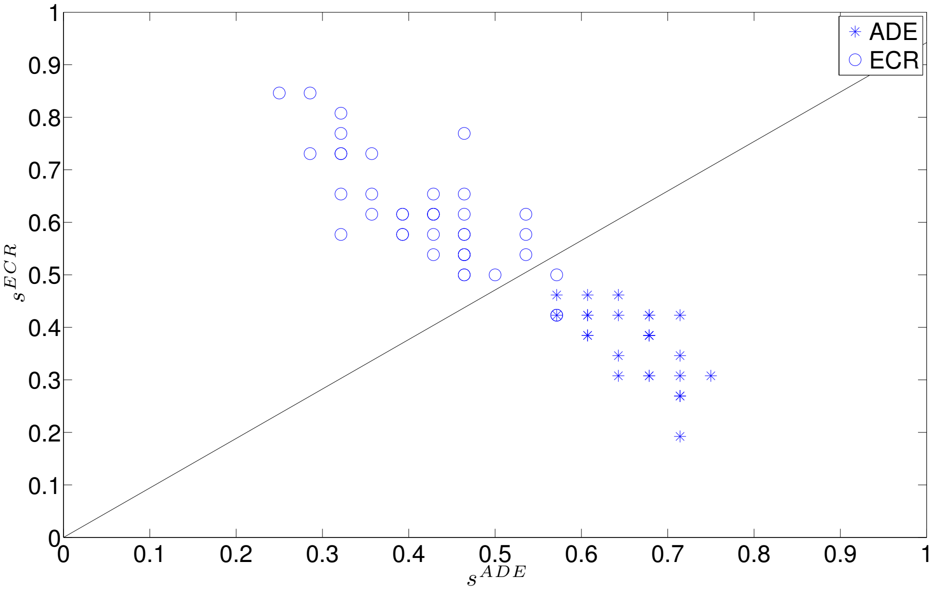


Figure 10.20: ADE vs ECR: graphical display of subjects

We have implemented the leave-two-out evaluation technique in section 7.4 to both of our signature-based classification methods to compute realistic estimates for

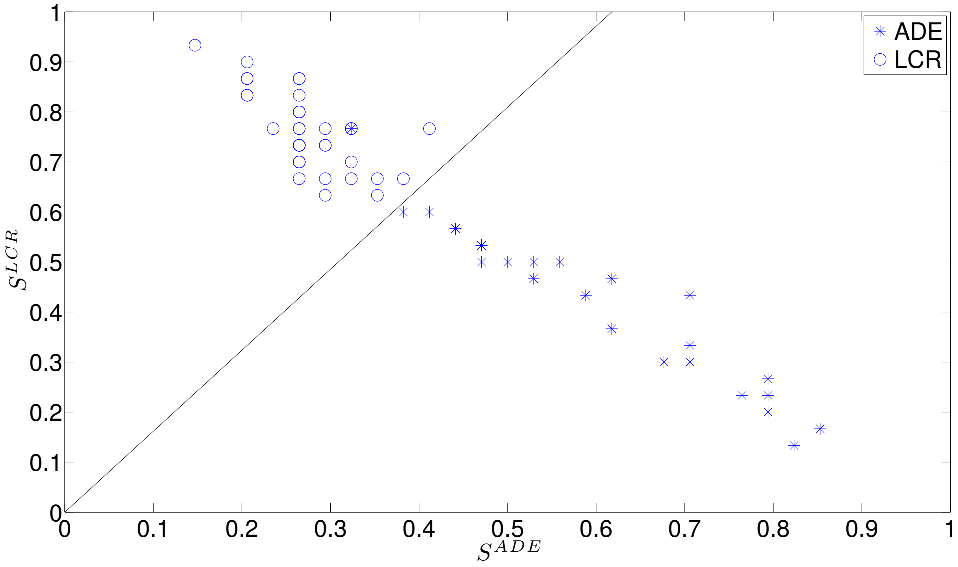


Figure 10.21: ADE vs LCR: graphical display of subjects

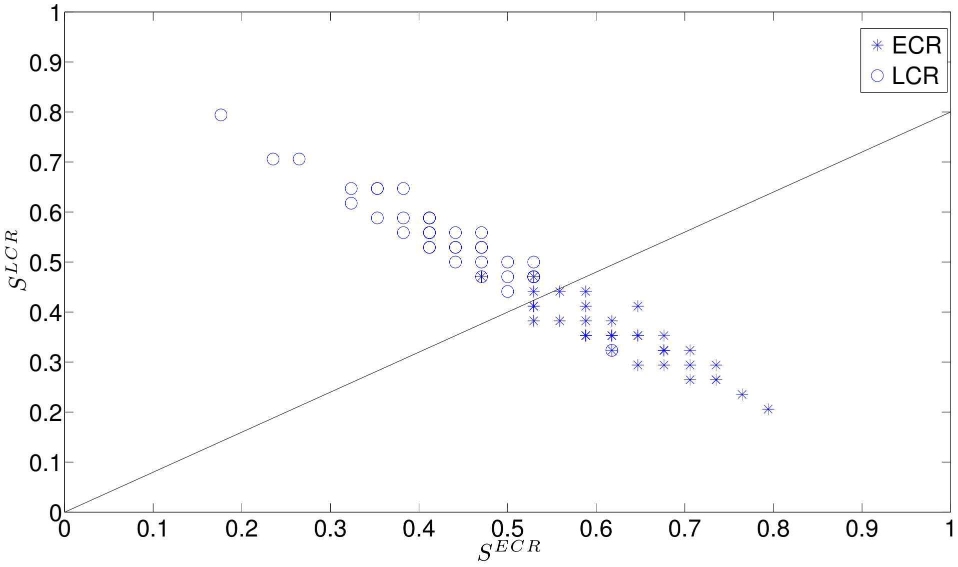


Figure 10.22: ECR vs LCR: graphical display of subjects

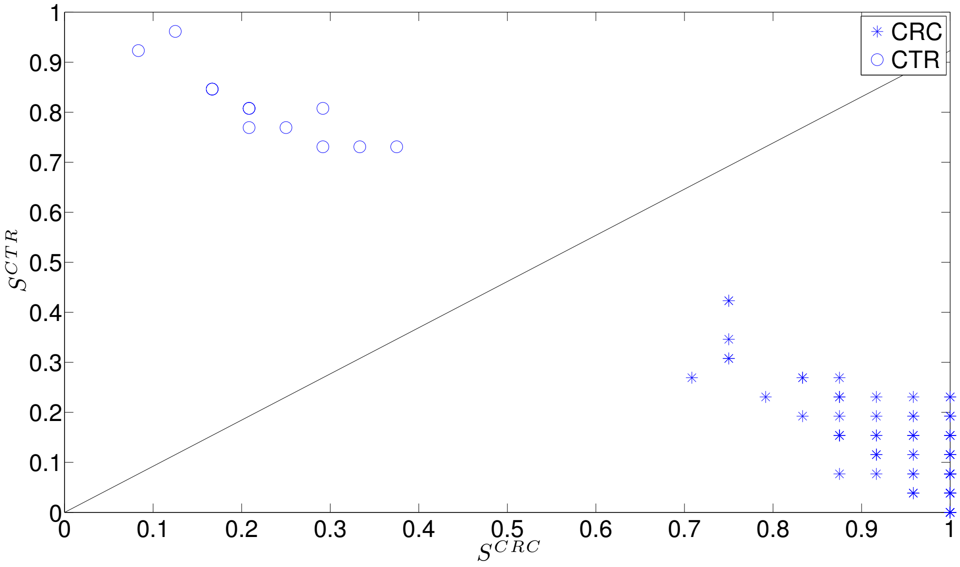


Figure 10.23: CRC vs CTR: graphical display of subjects

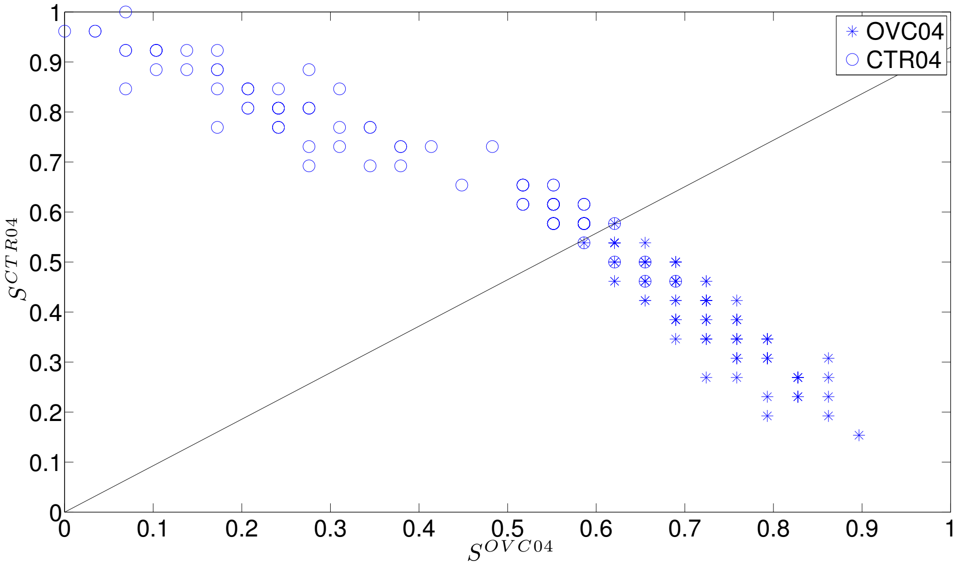


Figure 10.24: OVC04 vs OVC04: graphical display of subjects

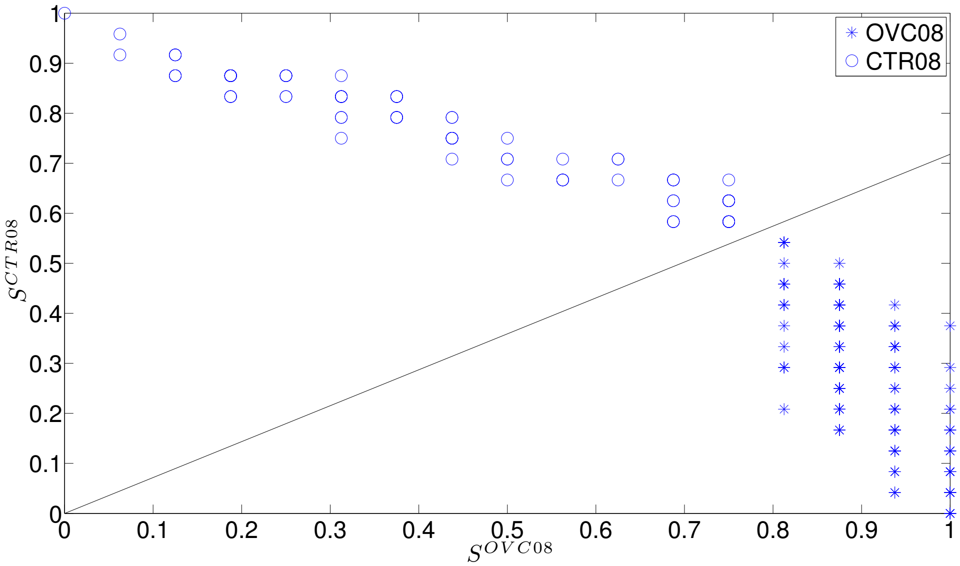


Figure 10.25: OVC08 vs OVC08: graphical display of subjects

their generalization on new dataset, which are summarized in Table 10.19. Applying classical bootstrapping methods [29], we have also computed a 95% confidence interval for sensitivity and specificity respectively.

10.2.6 Feasibility of Constructing Gibbs Based Classifier

In Chapter 5, we have set up a methodology to fit Gibbs distributions to mass spectra groups. In this section, we want to

1. Check the feasibility of fitting Gibbs distributions to groups of mass spectra acquired from cancer patients with a decent quality of fit,

2. Compute the optimal Gibbs based classifier $\mathring{g}$ associated to the pair π^+ and π^- of Gibbs distributions fitted to two groups G^+ and G^- of cancer mass spectra

	Sig by RLL		Sig by MDP	
Task	Sens.	Spec.	Sens.	Spec.
ADE vs ECR	93%	100%	88%	95%
	87-99%	96-100%	80-96%	89-100%
ADE vs LCR	96%	95%	89 %	99%
	90-100%	89-100%	81-97%	93-100%
ECR vs LCR	95%	81%	85%	87%
	89-100%	71-91%	76-94%	80-94%
CRC vs CTR	100%	100%	100%	100
	100-100%	100-100%	100-100%	100-100%
OVC04 vs CTR04	94%	92%	97%	92%
	88-100%	88-96%	94-100%	87-97%
OVC08 vs CTR08	100%	100%	100%	100%
	100-100%	100-100%	100-100%	100-100%

Table 10.19: Cross validation of signature based classifiers
This table lists the results of leave-two-out cross validation of Signature discovery by MDP and by RLL on six discrimination tasks. For one signature discovery algorithm on one discrimination task, both sensitivity and specificity are reported with their confidence interval underneath.

data,

3. Estimate the performance of the Gibbs based classifier $\mathring{g}$ by our leave two-out technique (see Chapter 7.4) and compare it to the performance of our signature based classifiers,

4. Compare the signature deduced from $\mathring{g}$ to the best signatures we discovered earlier.

As an example, we take the discrimination task between two groups of colorectal cancer data $G^+ = ECR$ versus $G^- = LCR$. The two groups contain respectively 80 and 74 mass spectra.

Table 10.9 lists the signature discovered for this discrimination task by our RLL algorithm. In Section 10.2.4.1, we had explored a biomarker target pool (including double-peaks biomarkers) of size 64. The sensitivity and specificity of this signature based classifier are 95% and 81%.

We then proceeded to fit two Gibbs models π^+ and π^- to the data sets $G^+ = ECR$ and $G^- = LCR$, namely

$$\pi^+(\boldsymbol{x}) = \frac{1}{Z^+} e^{-\boldsymbol{\theta}^{+*}\boldsymbol{U}^+(\boldsymbol{x})}, \quad \pi^-(\boldsymbol{x}) = \frac{1}{Z^-} e^{-\boldsymbol{\theta}^{-*}\boldsymbol{U}^-(\boldsymbol{x})}.$$

We chose 20 top ranked G^+ markers and 20 top ranked G^- markers according to activation frequency ratios.

For the model π^+, the singletons are the 20 G^+ markers. We computed the mutual information for every pair of markers within the 20 G^+ markers and used 0.04 as a cutoff to select size-2 cliques. This generated 12 size-2 cliques for π^+.

For π^-, the singletons are the 20 G^- markers and with a similar clique selection technique, we obtained 17 size-2 cliques.

We implemented the fast estimation algorithm MPLE (see Section 4.2.2) to estimate the parameter vectors $\boldsymbol{\theta}^+$ and $\boldsymbol{\theta}^-$ of our two Gibbs models. These estimated parameters are listed in Table 10.20.

We then evaluated the quality of fit of these two Gibbs models by the technique described in Section 5.3. As shown in Figure 10.26 and 10.27, the quantiles of the log-likelihoods on the true datasets are 68% and 32%, which indicate a reasonable quality of fit for our two estimated Gibbs models.

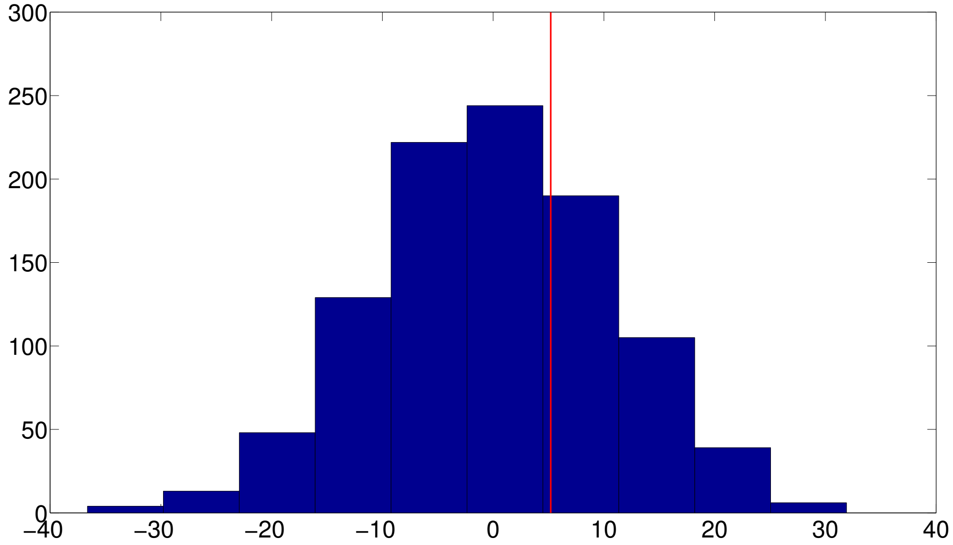


Figure 10.26: Quality of fit of π^+
The red line indicates the quantile of the log likelihood on the true dataset.

Recall that the optimal classifier between two Gibbs models π_1 and π_2 where

$$\pi_1(\boldsymbol{x}) = \frac{e^{-\boldsymbol{\theta}_1^* \boldsymbol{U}(\boldsymbol{x})}}{Z_1}, \ \pi_2(\boldsymbol{x}) = \frac{e^{-\boldsymbol{\theta}_2^* \boldsymbol{U}(\boldsymbol{x})}}{Z_2},$$

ECR		LCR	
Cliques (Markers)	Coeffs of $\boldsymbol{\theta}^{+}$	Cliques (Markers)	Coeffs of $\boldsymbol{\theta}^{-}$
1953	2.63	7889	1.21
7542	-0.79	2807	1.09
8376	1.29	2143	1.73
1112	4.11	3856	-0.29
4180	0.67	1210	1.78
6320	2.36	1032	3.96
921	1.69	1029	1.81
2589	0.04	819	2.83
6894	0.89	1213	1.63
6915	0.27	6226	-0.11
8032	0.25	2331	3.89
1570	0.20	1812	2.11
2558	1.18	4944	-0.96
8401	3.80	5829	1.47
5033	-0.75	1089	-0.67
6339	0.72	2628	-1.21
1288	0.91	4684	-0.80
6301	1.54	2338	3.89
2007	0.00	817	3.77
2998	-0.73	2182	1.63
(8376, 8401)	-3.56	(819, 817)	-6.88
(6320, 6301)	-2.37	(2331, 2338)	-4.85
(6894, 6915)	-2.08	(1210, 1213)	-3.64
(6320, 6339)	-1.74	(1032, 1029)	-3.03
(1112, 5033)	-2.19	(1213, 4944)	2.03
(1112, 2007)	-1.64	(2143, 3856)	-1.66
(2558, 8401)	-1.68	(2331, 2182)	-1.88
(1953, 6301)	-1.82	(1213, 2182)	-1.23
(8376, 4180)	-1.12	(2143, 2338)	-1.47
(1570, 2558)	1.66	(1029, 1812)	-1.10
(4180, 2589)	-1.24	(1213, 1812)	-0.92
(921, 5033)	-1.69	(1029, 2182)	-1.11
		(819, 1213)	1.49
		(1029, 5829)	-1.26
		(1812, 2182)	-0.86
		(1213, 817)	0.95
		(2807, 1032)	-1.37

Table 10.20: Estimated Gibbs Models separately fitted to the datata sets ECR and LCR

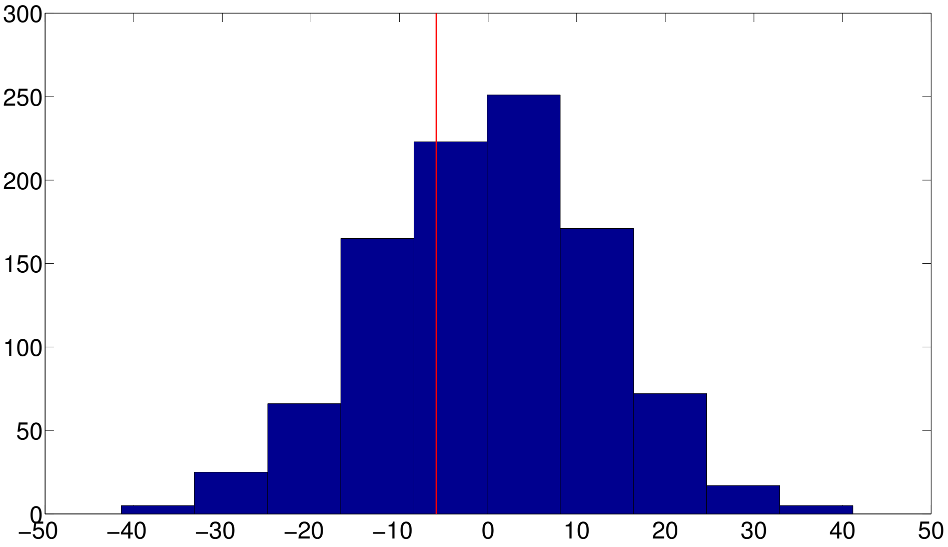


Figure 10.27: Quality of fit of π^-

The red line indicates the quantiles of the log likelihood on the true dataset.

is

$$\mathring{g}(\boldsymbol{x}) = \begin{cases} +1, & \text{if } (\boldsymbol{\theta}_2 - \boldsymbol{\theta}_1)^* \boldsymbol{U}(\boldsymbol{x}) - \log Z_1 + \log Z_2 > 0, \\ -1, & \text{otherwise.} \end{cases} \tag{10.1}$$

$\boldsymbol{U}(\boldsymbol{x})$ is a common factor of the two Gibbs models. In π^+ and π^-, $\boldsymbol{U}^+(\boldsymbol{x})$ and $\boldsymbol{U}^-(\boldsymbol{x})$ correspond to different set of biomarkers. Therefore, as indicated in Section 6.3, we generated the union UCL of the sets of cliques corresponding to all the biomarkers selected for two Gibbs models and constructed a single vector valued function $\boldsymbol{U}(\boldsymbol{x})$ of $\boldsymbol{x}$, gathering the binary activities $1_C(\boldsymbol{x})$ for all cliques $C \in UCL$. Two new Gibbs models $\tilde{\pi}^+$ and $\tilde{\pi}^-$ were then constructed, namely

$$\tilde{\pi}^+(\boldsymbol{x}) = \frac{1}{Z^+} e^{-\tilde{\boldsymbol{\theta}}^{+*} \boldsymbol{U}(\boldsymbol{x})}, \quad \tilde{\pi}^-(\boldsymbol{x}) = \frac{1}{Z^-} e^{-\tilde{\boldsymbol{\theta}}^{-*} \boldsymbol{U}(\boldsymbol{x})},$$

where the coordinates of $\tilde{\boldsymbol{\theta}}^+$ associated to any cliques which are originally in π^+ are still equivalent to the corresponding coordinates of $\boldsymbol{\theta}^{+}$ and the other coordinates are set to zero; $\tilde{\boldsymbol{\theta}}^{-}$ is similarly constructed.

We then derived the optimal classifier $\mathring{g}$ based on the two new Gibbs models $\tilde{\pi}^{+}$ and $\tilde{\pi}^{-}$, by the formula (10.1) just recalled above.

To compute Z^{+} and Z^{-}, since the size of the configuration space $\{0,1\}^{20}$ is only 1,048,576 we could use the explicit formulas

$$Z^{+} = \sum_{\boldsymbol{x}\in\{0,1\}^{20}} e^{-\boldsymbol{\theta}^{+*}\boldsymbol{U}^{+}(\boldsymbol{x})}, \quad Z^{-} = \sum_{\boldsymbol{x}\in\{0,1\}^{20}} e^{-\boldsymbol{\theta}^{-*}\boldsymbol{U}^{-}(\boldsymbol{x})},$$

which gave us the values $Z^{+} = 1.33 \times 10^{5}$ and $Z^{-} = 4.02 \times 10^{4}$. In the classifier $\mathring{g}$, the constant term is then given by $\log Z^{+} - \log Z^{-} = 1.19$.

We have computed the performance of the classifier $\mathring{g}$ and compared it to our signature based classifiers. The sensitivity and specificity of $\mathring{g}$ which we obtained by leave-two-out cross validation are 45% and 68%. This performance is clearly much worse than the performance of our signature based classifier (sensitivity 95% and specificity 81%). In addition, comparing Table 1 and Table 2, we find that the two sets of biomarkers are very different since they only have 18 common biomarkers. Of course this is not too surprising since the two target pools were not identical to begin with. Therefore, the two signatures are not comparable.

Now we can answer the four questions we raised at the beginning of this section.

1. We have successfully fitted Gibbs models to the two groups ECR and LCR of colorectal cancer mass spectra data, with good quality of fits. Previously, we had fitted a Gibbs model to LCR based on another list of features. Therefore fitting

Gibbs models by MPLE techniques to true cancer mass spectra datasets based on a list of reasonable selected features is quite feasible, and MPLE yields Gibbs models with good quality of fit.

This justifies in particular the underlying assumption we have made throughout this thesis that we could safely assume that after pre-processing and binary coding of mass spectra, we could assume that the binary coded mass spectra in each group had been generated by unknown Gibbs models. This was a crucial assumption since it told us what was the family of classifiers we should be exploring.

2. The actual computation of the best Gibbs classifier $\mathring{g}$ associated to our two estimated Gibbs models was not difficult except for the computation of the constant term $log(Z^{+}) - log(Z^{-})$. Here we used an explicit full computation of the partition functions because the size 2^{20} of the configuration space was moderate. For larger sizes one would have to use estimates of the partition function classically based on generating random binary configurations where all coordinates are independent and take the values 0 or 1 with probabilities 1/2 and 1/2.

3. We have seen that the performance of the Gibbs based classifier generated from two estimated Gibbs models was much worse than for the signature based classifier we obtained by our RLL algorithm. Obviously, the estimation of our two Gibbs models from only 80 and 74 configuration data is extremely risky , since our Gibbs models involved 32 and 37 coefficients respectively. As a rule of thumb for Gibbs model estimation one would require at least 10 configuration data per unknown coefficient, which would mean data sets of sizes larger than 320 and 370 respectively. The fact that the models had a reasonable goodness of fit is only a natural but

minimal requirement which of course does not guarantee the accuracy of parameter estimation by MPLE.

4. We found hardly any common pattern between the classifier derived from the two estimated Gibbs models and the signature based classifier we discovered by RLL algorithm. This is to be expected since our estimated Gibbs models may be wildly off the mark due to the very small size of the data sets.

Because of the really small size of our cancer mass spectra data sets and of the large number of parameters estimated, the Gibbs model estimation is necessarily quite imprecise and should be avoided unless we have much larger data sets of sizes at least 400. Therefore, the construction of optimal classifier from estimated Gibbs models is impractical for our colorectal cancer data sets. Our signature based discovery algorithms and associated classifiers fortunately eliminate the need to estimate Gibbs models, which is a much more efficient way to discover signatures and construct highly discriminating classifiers.

Chapter 11

Comparison with Machine Learning Algorithms

The goal of proteomics profiling is to discriminate clinical groups by recognizing proteomic data patterns. To handle the high dimensionality of mass spectrometry data, and their inherent variability, ”machine learning” algorithms have been popular approaches to facilitate automatic classification between mass spectra.

11.1 Machine Learning Algorithms

Support Vector Machine (SVM), K-Nearest Neighbors (KNN), Decision Tree (DT), and Random Forest (RF) are four popular machine learning algorithms used for mass spectrometry analysis. Comparison of the performance between our classification method and these algorithms has an important meaning in evaluating the improvement made by our signature discovery algorithms.

11.1.1 Support Vector Machine (SVM)

SVM was first introduced by Vapnik [85], and has been extensively used as an effective classification method on large-scale datesets because of its good generalization capacity. Given a set of observations $((\boldsymbol{x}_1, y_1), (\boldsymbol{x}_2, y_2), \cdots, (\boldsymbol{x}_n, y_n))$ of random variable $(\boldsymbol{x}, y)$, where $\boldsymbol{x} \in \mathcal{X}$, $y \in \{-1, 1\}$ labels the class of $\boldsymbol{x}$ (class -1 or class 1), SVM finds a hyperplane that can separate these data points into two classes. Consider a linear function $f(x) = \boldsymbol{w} \cdot \boldsymbol{x} + b$, which gives a decision rule to estimate y_i:

$$\hat{y}_i = \begin{cases} 1, \text{ if } \boldsymbol{w} \cdot \boldsymbol{x}_i + b \geq 1, \\ \\ -1, \text{ if } \boldsymbol{w} \cdot \boldsymbol{x}_i + b \leq -1. \end{cases}$$

The distance between the two hyperplanes $f(x) = 1$ and $f(x) = -1$ is $1/\|\boldsymbol{w}\|$, called margin.

Hyperplane $f(x) = 0$ with a larger margin will generalize better on new data. SVM looks for the best hyperplane with maximum margin for separable training sets, which is equivalent to solving the following quadratic optimization problem

$$\begin{array}{ll} \text{minimize} & \frac{1}{2}\|\boldsymbol{w}\|^2 \\ \text{Subject to} & y_i(\boldsymbol{w} \cdot \boldsymbol{x}_i + b) \geq 1, \\ & i = 1, 2, \cdots, n. \end{array}$$

Using Lagrangian multipliers, which are written as $\boldsymbol{\alpha} = (\alpha_1, \cdots, \alpha_n) \geq 0$, the problem can be transformed into a dual problem

$$\begin{aligned}
\text{minimize} \quad & \textstyle\sum_{i=1}^{n} \alpha_i - \frac{1}{2}\alpha_i\alpha_j y_i y_j < \boldsymbol{x}_i, \boldsymbol{x}_j > \\
\text{Subject to} \quad & y_i(\boldsymbol{w} \cdot \boldsymbol{x}_i + b) \geq 1, \\
& i = 1, 2, \cdots, n, \boldsymbol{\alpha} \geq 0.
\end{aligned}$$

For non-separable training set, the problem can be transformed into a Soft Margin-Dual Lagrangian problem

$$\begin{aligned}
\text{minimize} \quad & \textstyle\sum_{i=1}^{n} \alpha_i - \frac{1}{2}\alpha_i\alpha_j y_i y_j \boldsymbol{x}_i \cdot \boldsymbol{x}_j \\
\text{Subject to} \quad & \textstyle\sum_{i=1}^{n} \alpha_i y_i = 0, \\
& i = 1, 2, \cdots, n, 0 \leq \alpha_i \leq C.
\end{aligned}$$

Here, C is the trade-off parameter, which balances a large margin and a small error penalty. Larger C adds more weights to the margin, therefore, alleviates the error penalty.

After obtained $\boldsymbol{\alpha}$ from above optimization, $\boldsymbol{w}$ is a linear combination of the data vectors $\boldsymbol{x}_i$

$$\boldsymbol{w} = \sum_{i=1}^{n} y_i \alpha_i \boldsymbol{x}_i,$$

and b can be computed from the Kuhn-Tucker Theorem condition

$$\alpha_i(y_i(\boldsymbol{w} \cdot \boldsymbol{x}_i + b) - 1 + \xi_i) = 0$$

$$(C - \alpha_i)\xi_i = 0, i = 1, 2, \cdots n,$$

where $\xi_i = 0$ for separable training set.

To generalize this approach, we define the inner product $\boldsymbol{x}_1 \cdot \boldsymbol{x}_2$ by positive definite kernels $K(x_1, x_2)$, which enables non-linear classification by mapping inputs into high-dimensional feature spaces. The most commonly used kernel is the Gaussian kernel

$$K(x_1, x_2) = e^{-\|x_1 - x_2\|^2/2\sigma^2}.$$

Therefore, there are two parameters C and σ in such an SVM algorithm.

11.1.2 K-Nearest Neighbors (KNN)

KNN [58] is a non-parametric lazy learning algorithm. On a binary labeled dataset $D = ((\boldsymbol{x}_1, y_1), (\boldsymbol{x}_2, y_2), \cdots, (\boldsymbol{x}_n, y_n)), \boldsymbol{x} \in \mathcal{X}, y \in \{-1, 1\}$, it classifies a new point $\boldsymbol{x}$ according to a majority vote of the k-nearest points in the training dataset. Namely, KNN first define the distance $d(\boldsymbol{x}_i, \boldsymbol{x}_j)$ of two points in a space, then the algorithm chooses the label y of a new point $\boldsymbol{x}$ by the formula

$$y = \arg\max_{y} P(y \mid \boldsymbol{x}, D),$$

where $P(y \mid \boldsymbol{x}, D) =$ fraction of points $\boldsymbol{x}_i$ in $N_k(\boldsymbol{x})$ such that $y_i = y$ and $N_k(\boldsymbol{x})$ contains k nearest points of $\boldsymbol{x}$.

The most commonly used distance is Euclidean distance. Rescaling the data features is often needed before implementing KNN because we don't want a single feature to dominate the distance between data

11.1.3 Decision Tree (DT)

DT was first introduced by Breiman [18]. It is a supervised learning technique to generate classifiers and has shown good performance on problems where dimensionality is larger than the size of the training set size and/or a large majority of input variables are irrelevant. Given a binary labeled dataset $D = ((\boldsymbol{x}_1, y_1), (\boldsymbol{x}_2, y_2), \cdots, (\boldsymbol{x}_n, y_n)), \boldsymbol{x}_i \in \mathcal{X}, y_i \in \{-1, 1\}$, the DT algorithm aims to generate a flow-chart like binary tree (such as Figure 11.1) and to minimize the error in each leaf of the tree.

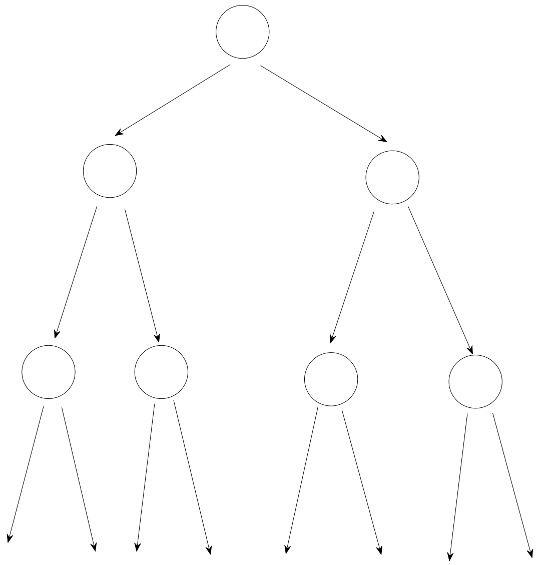


Figure 11.1: Flow-chart of a decision tree

Define the error function E_R on a space $R \subseteq \mathcal{X}$ be the fraction of points $\boldsymbol{x}_i \in R$ misclassified by a majority vote in $R \subseteq \mathcal{X}$. Define also two symbols

$$R_{k+1}(j, s) = \{\boldsymbol{x}_i \in R_k : x_{ij} > s\}, R'_{k+1}(j, s) = \{x_i \in R_k : x_{ij} \leq s\},$$

where x_{ij} is the jth feature of $\boldsymbol{x}_i$.

DT algorithm works as follows:

1. At the root node of the tree which splits the whole space $R_1 = \mathcal{X}$, choose

$j = j_{n_1}$ and $s = s_1$ to minimize

$$E_{R_2(j,s)} + E_{R'_2(j,s)}.$$

Two leaves of the tree are grown as in Figure 11.2.

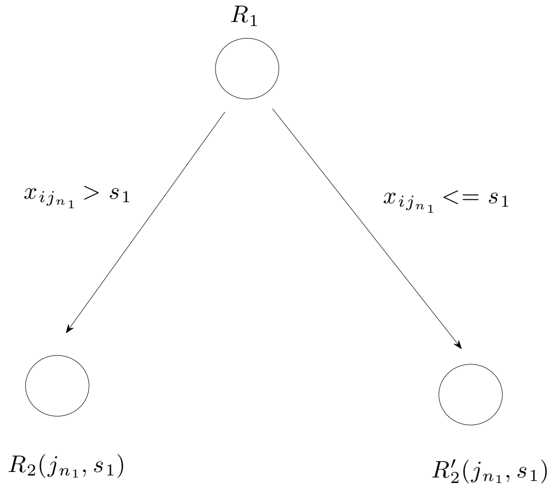


Figure 11.2: Two leaves on the root node

2. Let $R_2 = R_2(j_{n_1}, s_1)$, choose $j = j_{n_2}$ and s_2 to minimize

$$E_{R_3(j,s)} + E_{R'_3(j,s)}.$$

   This grows two leaves on the node as in Figure 11.3

3. Similarly, generate two branches to split the space $R'_2(j_{n_2}, s_2)$.

4. Repeat this process to grow leaves on each subsequent node until at some node, the space contains points only of one class.

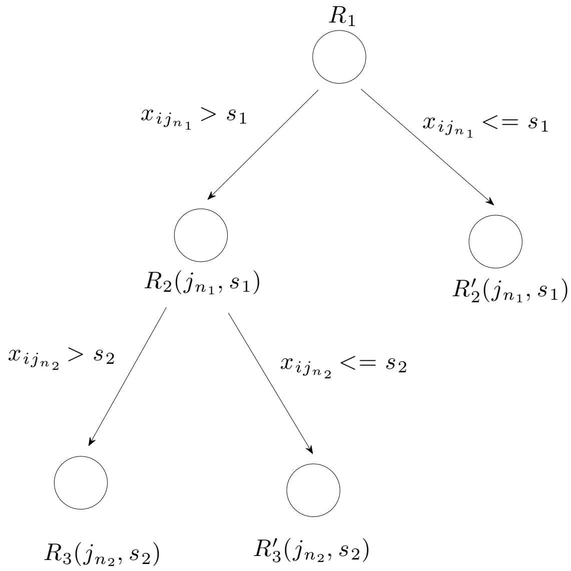


Figure 11.3: Two leaves on the second node

A decision tree provides a decision flow-chart. After a tree is grown, we can follow the flow-chart to classify a new observation $\boldsymbol{x}$. If, for example, we have grown a decision tree as in Figure 11.4. The class of observations in the three end leaves are +1, -1, and -1. A new observation $\boldsymbol{x}$ which has its j_{n_1}th feature larger than s_1 and its j_{n_2}th feature larger than s_2 will be classified into class +1; $\boldsymbol{x}$ with j_{n_1}th feature larger than s_1 and j_{n_2}th feature smaller than s_2 will be classified into class -1; $\boldsymbol{x}$ with j_{n_1}th feature smaller than s_1 will be classified into class -1 as well.

11.1.4 Random Forest (RF)

RF [17] is a method that combines two powerful ideas in machine learning techniques: bagging (bootstrap aggregating) and random feature selection. Given a dataset D,

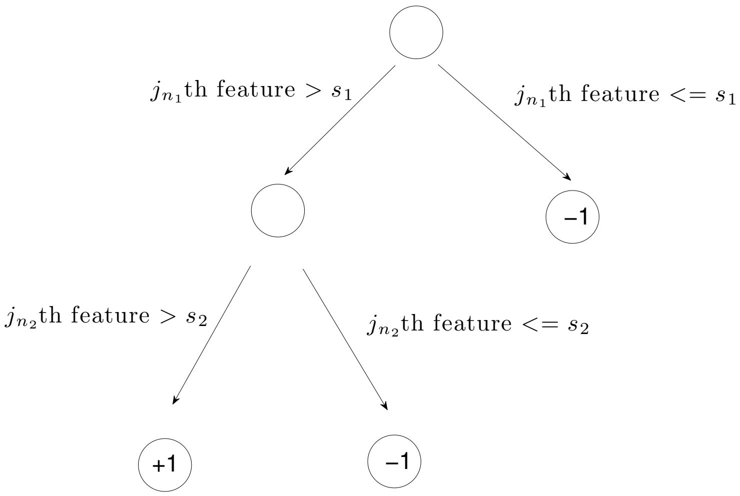

j_{n_1}th feature $> s_1$
j_{n_1}th feature $<= s_1$
-1
j_{n_2}th feature $> s_2$
j_{n_2}th feature $<= s_2$
$+1$
-1

Figure 11.4: A complete decision tree

one constructs multiple trees. For each tree $T_i(i = 1, 2, \cdots b)$, one

1. Selects a bootstrap sample set D_i from D;

2. Selects a random subset of features and restrict each sample in D_i on these features to construct a new sample set D_i';

3. Grows a tree T_i on D_i' as in Section 11.1.3.

For a new observation $\boldsymbol{x}$, every decision tree T_i gives a decision. One will take the majority vote of the decisions across all $T_i, i = 1, 2, \cdots, b$ to make a final decision on $\boldsymbol{x}$.

11.2 Performance Comparisons

Machine learning algorithms are always combined with feature selection techniques to perform classification on mass spectrometry data. t-statistic is a commonly used feature selection method as discussed in [12]. We use the t-statistic to rank all the reference peak abscissas and to select a small subset of features with large t-statistic values. The intensity information of each reference peak abscissa is required by the t-statistic, so for each spectrum, we define the intensity of reference peak abscissa x as the sum of the intensities of all the peaks that are within the error window of x.

We have computed the leave-two-out performance of the above machine learning algorithms for the classification of our pre-processed MALDI spectra. Table 11.1 and 11.2 compare the performance of machine learning algorithms SVM, KNN, DT and RF with our two signature discovery algorithms on the four tasks of MALDI spectra. For each task, Table 11.1 and 11.2 list the sensitivity and specificity of every algorithm. We used 20 reference peak abscissas as features for each machine learning algorithm because after testing several number of features, we found that there was no significant improvement in performance with larger or smaller numbers of features for these algorithms in our cases (Table 11.3). For Gaussian kernel-based SVM, we report the best results obtained by selecting the optimal trade-off parameter C and kernel parameter σ through maximizing the discrimination power on all the samples.

Overall, none of the four machine learning algorithms has better performance than our methods on the four tasks. Note also that the performance differences between the four machine learning algorithms and our signature discovery algorithms

Method	ADE vs ECR		ADE vs LCR		ECR vs LCR		CRC vs CTR	
	Sens.	Spec.	Sens.	Spec.	Sens.	Spec.	Sens.	Spec.
RLL	93%	100%	96%	95%	95%	81%	100%	100%
SVM	0%	100%	0%	100%	4%	81%	99%	99%
KNN	54%	24%	0%	100%	30%	40%	99%	98%
DT	53%	34%	27%	73%	26%	61%	99%	86%
RF	34%	48%	5%	87%	3%	83%	99%	85%

Table 11.1: Comparison of leave-two-out performance of RLL signature discovery with machine learning algorithms
20 reference peak abscissas were used as features for each machine learning algorithm. For each of the discrimination task, this table compares the leave-two-out performance of our RLL signature discovery reported in Table 10.19 with that of machine learning algorithms. Overall, our algorithm performs better than all the four machine learning algorithms on the four tasks.

Method	ADE vs ECR		ADE vs LCR		ECR vs LCR		CRC vs CTR	
	Sens.	Spec.	Sens.	Spec.	Sens.	Spec.	Sens.	Spec.
MDP	88%	95%	89 %	99 %	85%	87 %	100%	100%
SVM	0%	100%	0%	100%	4%	81%	99%	99%
KNN	54%	24%	0%	100%	30 %	40%	99 %	98 %
DT	53%	34%	27%	73%	26 %	61%	99%	86%
RF	34%	48%	5 %	87%	3%	83%	99%	85%

Table 11.2: Comparison of leave-two-out performance of MDP signature discovery with machine learning algorithms
20 reference peak abscissas were used as features for each machine learning algorithm. For each of the discrimination task, this table compares the leave-two-out performance of our signature discovery by MDP reported in Table 10.19 with that of machine learning algorithms. Overall, our algorithm performs better than all the four machine learning algorithms on the four tasks.

are quite large for the three tasks ADE vs ECR, ADE vs LCR, and ECR vs LCR. Namely, ADE, ECR, and LCR are not well separated by these four machine learning algorithms, but can be highly differentiated by our signature-based classifiers.

To take account of the possibility that some of the machine learning algorithms achieve better performance with different numbers of features, we have also computed their leave-two-out performance on 6 distinct numbers of features (N = 15, 20, 25, 30, 35, 40). We found that there was no significant performance variations with this range of values for the number of features. As a brief example, we only report their performance on 15 and 40 features in Table 11.3.

	ADE vs ECR		ADE vs LCR		ECR vs LCR		CRC vs CTR	
15 features	Sens.	Spec.	Sens.	Spec.	Sens.	Spec.	Sens.	Spec.
SVM	0%	100%	0%	100%	0%	86 %	99%	100%
KNN	33%	36%	16%	83%	23%	58%	99%	80 %
DT	39%	42%	22 %	72%	30 %	46%	100%	21%
RF	16%	59%	13 %	84%	5 %	67%	99 %	57%
40 features	Sens.	Spec.	Sens.	Spec.	Sens.	Spec.	Sens.	Spec.
SVM	0%	100%	0%	93%	0%	100 %	99%	7%
KNN	47%	26%	0 %	93 %	11%	67 %	99%	90%
DT	57%	29%	28%	65%	37%	44%	99%	8%
RF	33%	49%	5 %	91%	3 %	86%	99 %	63%

Table 11.3: Leave-two-out performance of machine learning algorithms with different numbers of features
The leave-two-out performance of the four machine learning algorithms are computed with 6 distinct numbers of features (N = 15, 20, 25, 30, 35, 40). Since there was no significant performance variations with this range of values for the number of features, this table only reports their performance on 15 and 40 features.

The two SELDI datasets have been discussed in a number of previous publications using multiple machine learning algorithms, among which, KNN, SVM, DT, and

RF combined with t-test feature selection are popular. We summarize the results reported by these publications.

For OVC04 vs CTR04, the authors of [5] selected 20 features by t-test and reported sensitivity/specificity as 83.3%/55.6% for the KNN classifier. Using the top 15 features selected by t-statistic, the classification of [26] reported an average of 97.3% and 96.6% discrimination power on two groups by RF and SVM. [38] selected 10 features by SVM and achieved a sensitivity/specificity of 92.8%/95%.

For OVC08 vs CTR08, publication [5] and [38] used respectively 20 and 10 peaks selected by t-test and achieved respectively 98.4%/100% and 100%/99.6% sensitivity/specificity.

Our signature discovery by RLL and MDP have shown respectively 94%/92% and 97%/92% sensitivity/specificity on OVC04 vs CTR04 with confidence intervals 88-100% /88-96% and 94-100%/ 87-97%. These results are equivalent or better than those in previous publications. Besides, our signature-based classifiers have also perfectly separated OVC08 from CTR08 as was achieved by previous authors using classical machine learning algorithms.

Chapter 12

Conclusion and Future Research

Mass spectrometry is a very promising approach for biomarker-based early cancer detection. A natural goal of mass spectrometry analysis is to isolate biologically relevant biomarkers to construct classifiers for early clinical diagnosis, to monitor disease progression or to evaluate response to treatment, in order to improve the design of therapeutic strategies. To further help incorporating these biomarkers into clinical protocols, a key step is to discover robust and interpretable "biomarkers signature profiles" for each cancer stage. Although machine learning algorithms, such as decision trees, support vector machines, artificial neural networks and ect., have been generally accepted as very efficient methods to discriminate between groups of mass spectra, their "black-box" results often lack interpretability. Our main focus was to investigate the underlying patterns in mass spectrometry datasets and to design more efficient and interpretable signature discovery algorithms.

Based on any list of reference peak abscissas, we can code each mass spectrum by a binary vector according to the binary status (activated / not activated) of each reference peak abscissa. A reference peak abscissa Ab is said to be "activated" by a mass spectrum M if there exists a strong peak in M that is within an error window of Ab. Gibbs distributions, as used in our work, are efficient stochastic models to study the spatial dependency of coordinates for high dimensional binary vectors viewed as realizations of random Markov random fields. We focus our study on Gibbs models which involve only "cliques" of size 1 and 2. We study three methods, maximum likelihood estimation, maximum pseudolikelihood estimation and marginal fitting estimation, to estimate the parameters of a Gibbs model. We have tested these three methods on typical mass spectrometry datasets obtained by MALDI-TOF techniques from plasma samples of late stage colorectal cancer patients, and compared the quality of fit of the estimated Gibbs models. In our case studies, the parameters estimated from the three methods are comparable, and we numerically proved that the underlying distributions of typical homogeneous groups of mass spectra (coded by a short list of reference peak abscissas) can be approximated by Gibbs models.

Neyman-Pearson test of hypothesis based on log-likelihood are known to be optimal. To discriminate between two Gibbs distributions, we have derived these optimal classifiers, which can be expressed as linear separators on an extended space.

A signature is here a list of highly discriminating biomarkers which have strong discriminating power when combined. A biomarker, in this context, is a strong peak determined by a mass to charge ratio m/z. In our study, we first condense all m/z ratios of strong peaks into a list of reference peak abscissas. We then extract a biomarker target pool from the list of reference peak abscissas according to a ranking based on activation frequencies ratios. We also consider biomarkers defined as pairs of reference peak abscissas and select a list of such double-peak biomarkers again by ranking their activation frequency ratios. The activation frequency ratio is indeed, in this context, a nearly optimal biomarker selection criterion. To discriminate between two distinct homogeneous groups of mass spectra G^+ and G^-, these biomarkers are then divided into two sets: G^+ biomarkers and G^- biomarkers.

We have designed a signature discovery algorithm by robust log likelihood analysis, called here RLL signature discovery for short, in order to search for an optimal classifier without constructing underlying Gibbs distributions. The biomarkers involved in this classifier constitute an optimal signature.

Our second signature discovery algorithm, called MDP signature discovery, is driven by a powerful stochastic optimization tool, simulated annealing. In this algorithm, we search for two lists of biomarkers which maximize the G^+ detecting power and G^- detecting power respectively and construct a specific classifier on the coded spectrometry space.

We have tested the efficiency of our two signature discovery algorithms by intensive simulations. Both of them have reached performance results very close to the optimal discrimination power provided the size of the training set is large enough. The signatures and signature-based classifiers generated by both algorithms converge to the true signature and the optimal classifier when the size of the training set becomes large enough. RLL signature discovery has shown clearly faster convergence speed than MDP signature discovery.

There are two most commonly used mass spectra acquisition technologies, MALDI-TOF and SELDI-TOF. We have studied both types of datasets.

To lower data dimensionality and reduce data acquisition ”noise”, our study starts by mass spectra pre-processing, which is a standard first step, often implemented via commercial interactive software. Pre-processing principles are well known, but implementation details vary considerably, and are often not accessible in commercial software. For better context control, we have developed our own sequence of pipelined pre-processing steps for each raw mass spectrum, implementing peak normalization, peak denoising, baseline removal and peak detection.

After our pre-processing procedure, we compute for each of the peaks a ”reliability score”, implemented by intensive Monte-Carlo simulations to emulate the underlying biomarkers variability. This provides a tool to compare the reliability of discovered signatures. To take account of peak error window and make mass spectra comparable, we construct a list of ”consensus” reference peak abscissas, each of which represents a cluster of peak positions within a horizontal error window. This ”condensation” leads us focus on the ”activation frequencies” of the reference peak abscissas.

We have successfully tested our two signature discovery algorithms on a new experimental set of 238 MALDI-TOF mass spectra acquired from patients at various stages of colorectal cancer and two previously published data sets of SELDI-TOF mass spectra acquired from ovarian cancer patients. The performance levels are computed by leave-two-out cross validation technique. The performances of our new signature discovery algorithms were good in all these concrete cases and compared quite favorably with the performance levels achieved by other machine learning algorithms as well as with published results for the same ovarian cancer dataset. The comparison further shows that our signature discovery approach handles easy discrimination tasks (colorectal cancer versus control) certainly as well as all existing data analysis techniques, but also exhibits clearly better performances for the more delicate discrimination between successive colorectal cancer stages.

A natural major long term goal is to transform our signature discovery algorithms into clinically viable techniques. To get to that stage, both algorithms should be validated on a large number of real datasets. In the future, we intend to calibrate and test our signature discovery techniques on more mass spectrometry datasets acquired from various cancerous patient groups.

Gibbs modeling provides a way to estimate the underlying distribution of mass spectrometry samples and therefore, can be used to simulate virtual mass spectra. In our study, although we have simulated virtual samples for the purpose of studying machine learning performance, we have not attempted to simulate the random noise affecting abscissas and ordinates on mass spectra. Thus, in the process of testing our signature discovery algorithms on these virtual datasets, we have not needed to implement the pre-processing step. However, due to the lack of publicly available large bases of real mass spectrometry datasets, simulating virtual mass spectra datasets is a practical solution for validating mass spectrometry data mining algorithms. In future, we intend to analyze and model the background noise in typical mass spectra in order to simulate more accurate virtual mass spectrometry samples. This will provide a better context to intensively test signature discovery algorithms.

Bibliography

[1] E. Aarts and J. Korst. *Simulated Annealing and Boltzmann Machines.* Wiley, New York, 1989.

[2] B. L. Adam et al. Serum protein fingerprinting coupled with a pattern-matching algorithm distinguishes prostate cancer from benign prostate hyperplasia and healthy men. *Cancer Res.*, 62:3609–3614, 2002.

[3] R. Aebersold and M. Mann. Mass spectrometry-based proteomics. *Nature*, 422:198–207, 2003.

[4] G. Alexe et al. Ovarian cancer detection by logical analysis of proteomic data. *Proteomics*, 4:766–783, 2004.

[5] A. Assareh and M. H. Moradi. Extracting efficient fuzzy if-then rules from mass spectra of blood samples to early diagnosis of ovarian cancer. *Proceedings of the 2007 IEEE Symposium on Computational Intelligence in Bioinformatics and Computational Biology*, 2007.

[6] R. Azencott. *Simulated Annealing: Parallelization Techniques.* Wiley, New York, 1992.

[7] K. A. Baggerly et al. A comprehensive approach to the analysis of matrix-assisted laser desorption/ionization-time of flight proteomics spectra from serum samples. *Proteomics*, 3:1667–1672, 2003.

[8] R. R. Bahadur. Rates of convergence of estimates and test statistics. *Ann. Math. Statist.*, 2:303–324, 1967.

[9] G. Ball et al. An integrated approach utilizing artificial neural networks and SELDI mass spectrometry for the classification of human tumours and rapid identification of potential biomarkers. *Bioinformatics*, 18:395–404, 2002.

[10] S. Becker et al. Surfaced-enhanced laser desorption/Ionization time-of-flight (SELDI-TOF) differentiation of serum protein profiles of BRCA-1 and sporadic breast cancer. *Ann. Surg. Oncol.*, 11:907–914, 2004.

[11] M. Bellew et al. A suite of algorithms for the comprehensive analysis of complex protein mixtures using high-resolution LC-MS. *Bioinformatics*, 22:1902–1909, 2006.

[12] R. H. Berk. Consistency and asymptotic normality of MLEs for exponential models. . *Ann. Math. Statist.*, 43:193–204, 1972.

[13] J. Besag. Statistical analysis of non-lattice data. *Statistician*, 24:179–195, 1975.

[14] G. Bhanot et al. A robust meta-classification strategy for cancer detection from MS data. *Proteomics*, 6:592–604, 2006.

[15] C. Bishop. *Pattern Recongnition and Machine Learning.* Springer, New York, 2006.

[16] A. Bouamrani et al. Mesoporous silica chips for selective enrichment and stabilization of low molecular weight proteome. *Proteomics*, 10:496–505, 2010.

[17] L. Breiman. *Randomforest.* Technical Rerport, Stat. Dept. UCB, 2001.

[18] L. Breiman et al. *Classification and Regression Trees.* Wadsworth International, CA, 1984.

[19] P. Brémaud. *Markov chains: Gibbs fields, Monte Carlo Simulation, and Queues.* Springer-Verlag, New York, 1999.

[20] L. D. Brown. *Fundamentals of Statistical Exponential Families.* Institute of Mathematical Statistics, Hayward, 1986.

[21] S. Chen et al. Wavelet-based procedures for proteomic mass spectrometry data processing. *Computational Statistics & Data Analysis*, 52:211–220, 2007.

[22] T. P. Conrads et al. High-resolution serum proteomic features for ovarian cancer detection. *Endocr. Relat. Cancer*, 11:163–178, 2004.

[23] K. R. Coombes et al. Quality control and peak finding for proteomics data collected from nipple aspirate fluid by surface-enhanced laser desorption and ionization. *Clin. Chem.*, 49:1615–1623, 2003.

[24] K. R. Coombes et al. Improved peak detection and quantification of mass spectrometry data acquired from surface-enhanced laser desorption and ionization by denoising spectra with the undecimated discrete wavelet transform. *Proteomics*, 5:4107–4117, 2005.

[25] A. Cruz-Marcelo et al. Comparison of algorithms for pre-processing of SELDI-TOF mass spectrometry data. *Bioinformatics*, 24:2129–2136, 2008.

[26] S. Datta and L. M. DePadilla. Feature selection and machine learning with mass spectrometry data for distinguishing cancer and non-cancer samples. *Statistical Methodology*, 3:79–92, 2005.

[27] P. Du et al. Improved peak detection in mass spectrum by incorporating continuous wavelet transform-based pattern matching. *Bioinformatics*, 22:2059–2065, 2006.

[28] P. Du et al. Data reduction of isotope-resolved LC-MS spectra. *Bioinformatics*, 23:1394–1400, 2007.

[29] B. Efron and R. Tibshirani. *An Introduction to the Bootstrap.* Chapman and Hall, London, 1993.

[30] A. El-aneed, Anas Cohen and J. Banoub. Mass spectrometry, review of the basics: electrospray, MALDI, and commonly used mass analyzers. *Applied Spectroscopy Reviews*, 44:210–230, 2009.

[31] G. Ge and G. W. Wong. Classification of premalignant pancreatic cancer mass-spectrometry data using decision tree ensembles. *BMC Bioinformatics*, 9:907–914, 2008.

[32] P. Geurts et al. Proteomic mass spectra classication using decision tree based ensemble methods. *Bioinformatics*, 21:3138–3145, 2005.

[33] S. M. Hanash et al. Mining the plasma proteome for cancer biomarkers. *Nature*, 452:571–579, 2008.

[34] P. B. Harrington et al. Bootstrap classification and point-based feature selection from age-staged mouse cerebellum tissues of matrix assisted laser desorption/ionization mass spectra using a fuzzy rule-building expert system. *Analytica Chimica Acta*, 599:219–231, 2007.

[35] M. Hilario et al. Machine learning approaches to lung cancer prediction from mass spectra. *Proteomics*, 3:1716–1719, 2003.

[36] M. Hilario et al. Processing and classification of protein mass spectra. *Mass Spectrometry Reviews*, 25:409–449, 2006.

[37] G. Izmirlian. Application of the random forest classification algorithm to a SELDI-TOF proteomics study in the setting of a cancer prevention trial. *Ann. NY Acad. Sci.*, 1020:154–174, 2004.

[38] K. Jong et al. Feature selection in proteomic pattern data with support vector machines. *Proceedings of the IEEE Symposium on Computational Intelligence in Bioinformatics and Computational Biology*, pages 41–48, 2004.

[39] Y. V. Karpievitch et al. PrepMs: TOF mass data graphical preprocessing tool. *Bioinformatics*, 23:264–265, 2007.

[40] M. Katajamaa and J. Miettinen. MAmine:Toolbox for processing and visualization of mass spectrometry based molecular profile data. *Bioinformatics*, 22:634–636, 2006.

[41] S. Kirkpatrick et al. Optimization by simulated annealing. *Science*, 220:671–680, 1983.

[42] J. Koopmann et al. Serum diagnosis of pancreatic adenocarcinoma using surface-enhanced laser desorption and ionization mass spectrometry. *Clin. Cancer Res.*, 10:860–868, 2004.

[43] K. R. Kozak et al. Identification of biomarkers for ovarian cancer using strong anion-exchange proteinchip: potential use in diagnosis and prognosis. *Proc. Natl. Acad. Sci. USA*, 100:12343–12348, 2003.

[44] L. Lancashire et al. Classification of bacterial species from proteomic data using combinatorial approaches incorporating artificial neural networks, cluster analysis and principal components. *Bioinformatics*, 21:2191–2199, 2005.

[45] E. Lange et al. High-accuracy peak picking of proteomics data using wavelet techiniques. *Pac. Symp. Biocomput., Hawaii, USA*, pages 243–254, 2006.

[46] A. Lebrecht et al. Surface-enhanced laser desorption/ionization time-of-flight mass spectrometry to detect breast cancer markers in tears and serum. *Cancer Genomics & Proteomics*, 6:75–84, 2009.

[47] E. Lehmann. The Fisher, Neyman-Pearson theories of testing hypothesis: One theory or two? *J. Amer. Statist. Assoc.*, 88:1242–1249, 1993.

[48] E. L. Lehmann and G. Casella. *Theory of Point Estimation (2nd ed.).* Springer-Verlag, New York, 1998.

[49] K. C. Leptos et al. Map-Quant: open-source software for large-scale protein quantification. *Proteomics*, 6:1770–1782, 2006.

[50] J. Li et al. Proteomics and bioinformatics approaches for identification of serum biomarkers to detect breast cancer. *Clin. Chem.*, 48:1296–1304, 2002.

[51] L. Li et al. Data mining techniques for cancer detection using serum proteomic profiling. *Artif. Intell. Med.*, 32:71–83, 2004.

[52] X. Li et al. *Bioinformatics and Computational Biology Solutions Using R and Bioconductor*. Springer, New York, 1999.

[53] R. H. Lilien et al. Probabilistic disease classification of expression-dependent proteomic data from mass spectrometry of human. *J. Comp. Biol.*, 10:925–946, 2003.

[54] H. Liu et al. A comparative study on feature selection and classification methods using gene expression profiles and proteomic patterns. *Genome Inform.*, 13:51–60, 2002.

[55] Q. Liu et al. Identification of differentially expressed proteins using MALDI-TOF mass spectra. *Asilomar Conf. on Signals, Systems and Computers*, 2003.

[56] D. I. Malyarenko et al. Enhancement of sensitivity and resolution of surface-enhanced laser desorption/ionization time-of-flight mass spectrometric records for serum peptides using time-series analysis techniques. *Clin.Chem.*, 51:65–74, 2005.

[57] D. Mantini et al. LIMPIC: a computational method for the separation of protein MALDI-TOF-MS signals from noise. *BMC Bioinformatics*, 8:101, 2007.

[58] D. L. Massart et al. *The k-nearest neighbor method. In Chemometrics: A Textbook(Data Handling in Science and Technology, Vol 2).* Elsevier Science, New York, 1988.

[59] J. S. Morris et al. Feature extraction and quantification for mass spectrometry in biomedical applications using the mean spectrum. *Bioinformatics*, 21:1764–1775, 2003.

[60] P. Neville et al. Generalizable mass spectrometry mining used to identify disease state biomarkers from blood serum. *Proteomics*, 3:1710–1715, 2003.

[61] K. Ning et al. PepSOM: an algorithm for peptide identification by tandem mass specrometry based on SOM. *Genome Inform.*, 17:194–205, 2003.

[62] J. L. Norris et al. Processing MALDI mass spectra to improve mass spectral direct tissue analysis. *International Journal of Mass Spectrometry*, 260:212–221, 2007.

[63] J. H. Oh et al. Prostate cancer biomarker discovery using high performance mass spectral serum profiling. *Computer Methods and Programs in Biomedicine*, 96:33–41, 2009.

[64] C. P. Paweletz et al. Proteomic patterns of nipple aspirate fluids obtained by SELDI-TOF: potential for new biomarkers to aid in the diagnosis of breast cancer. *Dis. Markers*, 17:301–307, 2001.

[65] E. F. Petricoin et al. Use of proteomic patterns in serum to identify ovarian cancer. *Lancet*, 359:572–577, 2002.

[66] E. F. Petricoin and L. A. Liotta. SELDI-TOF-based serum proteomic pattern diagnostics for early detection of cancer. *Curr. Opin. in Biotechnol.*, 15:24–30, 2004.

[67] T. C. W. Poon et al. Comprehensive proteomic profiling identifies serum proteomic signatures for detection of hepatocellular carcinoma and its subtypes. *Clin. Chem.*, 49:752–760, 2003.

[68] J. Prados et al. Mining mass-spectra for diagnosis and biomarker discovery of cerebral accidents. *Proteomics*, 4:2320–2332, 2004.

[69] W. Press et al. *Numerical Recipes: The Art of Scientific Computing (3rd ed.).* Cambridge University Press, New York, 2007.

[70] P. V. Purohit and D. M. Rocke. Discriminant models for hight-throughput proteomics mass spectrometer data. *Proteomics*, 3:1699–1703, 2003.

[71] Y. Qu et al. Data reduction using a discrete wavelet transform in discriminant analysis of very high dimensionality data. *Biometrics*, 59:143–151, 2003.

[72] Y. S. Qu et al. Boosted decision tree analysis of surface-enhanced laser desorption/ionization mass spectral serum profiles discriminates prostate cancer from noncancer patients. *Clin. Chem.*, 48:1835–1843, 2002.

[73] A. J. Rai et al. Proteomic approaches to tumor marker discovery. *Arch. Pathol. Lab. Med.*, 126:1518–1526, 2002.

[74] H. W. Ressom et al. Analysis of mass spectral serum profiles for biomarker selection. *Bioinformatics*, 21:4039–4045, 2005.

[75] R. Romero et al. Proteomic analysis of amniotic fluid to identify women with preterm labor and intra-amniotic inflammation/infection: The use of a novel computational method to analyze mass spectrometric profiling. *J. Matern. Fetal. Neonatal. Med.*, 21:367–388, 2008.

[76] F. Rosenblatt. The Perceptron–a perceiving and recognizing automaton. *Report 85-460-1, Cornell Aeronautical Laboratory*, 1957.

[77] Y. Saeys et al. A review of feature selection techniques in bioinformatics. *Bioinformatics*, 23:2507–2517, 2007.

[78] G. A. Satten et al. Standardization and denoising algorithms for mass spectra to classify whole-organism bacterial specimens. *Bioinformatics*, 20:3128–3136, 2004.

[79] C. Shen et al. Comparison of computational algorithms for the classification of liver cancer using SELDI mass spectrometry: a case study. *Cancer Informatics*, 3:339–349, 2007.

[80] H. Shin and M. K. Markey. A machine learning perspective on the development of clinical decision support system utilizing mass spectra of blood samples. *J. Biomed. Informatics*, 39:227–248, 2006.

[81] D. J. Slotta et al. Clustering mass spectrometry data using order statistics. *Proteomics*, 3:1687–1691, 2003.

[82] J. M. Sorace and M. Zhan. A data review and re-assessment of ovarian cancer serum proteomic profiling. *BMC Bioinformatics*, 4:24, 2003.

[83] M. Stone. Cross-validatory choice and assessment of statistical predictions. *J. R. Statist. Soc.*, 36:111–147, 1974.

[84] R. Tibshirani et al. Sample classification from protein mass spectrometry by 'peak probability contrast'. *Bioinformatics*, 20:3034–3044, 2004.

[85] V. Vapnik. *Statistical Learning Theory.* Wiley, New York, 1998.

[86] A. Vlahou et al. Diagnosis of ovarian cancer using decision tree classification of mass spectral data. *J. Biomed. Biotechnol.*, 2003:308–314, 2003.

[87] M. Wagner et al. Protocols for disease classification from mass spectrometry data. *Proteomics*, 3:1692–1698, 2003.

[88] M. Z. Wang et al. Analysis of human serum proteins by liquid phase isoelectric focusing and matrix-assisted laser desorption/ionization-mass spectrometry. *Proteomics*, 3:1661–1666, 2003.

[89] R.-E. F. K.-W. C. C.-J. H. X.-R. Wang and C.-J. Lin. LIBLINEAR: A library for large linear classification. *J. Mach. Learn. Res.*, 9:1871–1874, 2008. Software available at http://www.csie.ntu.edu.tw/ cjlin/liblinear.

[90] W. Wegdam et al. Classification-based comparison of pre-processing methods for interpretation of mass spectrometry generated clinical datasets. *Proteome Science*, 7:19, 2009.

[91] T. Whistler et al. A method for improving SELDID-TOF mass spectrometry data quality. *Proteome Science*, 5:14, 2007.

[92] J. W. H. Wong et al. SpecAlign - processing and alignment of mass spectra datasets. *Bioinformatics*, 21:2088–2090, 2005.

[93] B. L. Wu et al. Comparison of statistical methods for classification of ovarian cancer using mass spectrometry data. *Bioinformatics*, 19:1636–1643, 2003.

[94] C. Yang et al. Comparison of public peak detection algorithms for MALDI mass spectrometry data analysis. *BMC bioinformatics*, 10, 2009.

[95] Y. Yasui et al. A data-analytic strategy for protein biomarker discovery: profiling of high-dimensional proteomic data for cancer detection. *Biostatistics*, 4:449–463, 2003.

[96] Y. Yasui et al. An automated peak identification/calibration procedure for high-dimensional protein measures from mass spectrometers. *J. Biomed. Biotechnol.*, 2003:242–248, 2003.

[97] J. S. Yu et al. Ovarian cancer identification based on dimensionality reduction for high-throughput mass spectrometry data. *Bioinformatics*, 21:2200–2209, 2005.

[98] Z. R. Yurkovetsky et al. Multiple biomarker panels for early detection of ovarian cancer. *Future Oncol.*, 2:733–741, 2006.

[99] B. Zhang et al. Review: proteomics and biomarkers for ovarian cancer diagnosis. *Annals of Clinical & Laboratory Science*, 40:218–225, 2010.

[100] X. G. Zhang et al. Recursive SVM feature selection and sample classification for mass-spectrometry and microarray data. *BMC Bioinformatics*, 7:197, 2006.

[101] H. Zhu et al. Tree-based disease classification using protein data. *Proteomics*, 3:1673–1677, 2003.

[102] W. Zhu et al. Detection of cancer-specific markers amid massive mass spectral data. *Proc. Natl. Acad. Sci. USA*, 100:14666–14671, 2003.