MASS SPECTROMETRY BASED PROTEOMIC BIOMARKER
SELECTION AND SAMPLE PREDICTION

by

JUNG HUN OH

Presented to the Faculty of the Graduate School of
The University of Texas at Arlington in Partial Fulfillment
of the Requirements
for the Degree of

DOCTOR OF PHILOSOPHY

THE UNIVERSITY OF TEXAS AT ARLINGTON
August 2008
To my wife Young Bun, my daughter Yuna,
my parents, my sisters and my wife’s family
ACKNOWLEDGEMENTS

I would like to express my deepest gratitude to my advisor, Dr. Jean Gao, for her excellent guidance, patience and support to reach my goal. Through my doctoral work, she encouraged me to develop independent thinking and overall abilities in research. She continually stimulated my analytical thinking and greatly assisted me to develop new efficient algorithms.

I want to thank my exceptional doctoral committee, Dr. Nikola Stojanovic, Dr. Pawel Michalak, Dr. Leonidas Fegaras and Dr. Roger Walker, for their continual encouragement and advice. In comprehensive exam and defence, their suggestions and comments helped me improve my research and dissertation.

I wish to extend my warmest thanks to collaborators, Dr. Kevin Rosenblatt and Prem Gurnani, for allowing me to use their data sets and helping me to develop my background in mass spectrometry. Our good collaboration enabled us to publish several papers.

I am grateful to Biocomputing and Vision Lab members and my friends who were always willing to help and give their best suggestions through the good times and bad times.

Finally, I would like to thank my wife Young Bun and daughter Yuna for the constant love, support and encouragement. Especially, Young Bun is a good collaborator with
whom I can discuss about my research. We are very happy to complete our PhD degree together. I am also very grateful to my parents, sisters and my wife’s family for their endless supports. This dissertation would not have been possible without their helps.

June 17, 2008
ABSTRACT

MASS SPECTROMETRY BASED PROTEOMIC BIOMARKER SELECTION AND SAMPLE PREDICTION

Jung Hun Oh, Ph.D.

The University of Texas at Arlington, 2008

Supervising Professor: Dr. Jean Gao

High-resolution MALDI-TOF (matrix-assisted laser desorption/ionization time-of-flight) mass spectrometry has recently shown promise as a screening tool for detecting discriminatory peptide/protein patterns. The major computational obstacle in finding such patterns is the large number of mass/charge peaks (features, biomarkers, data points) in a spectrum. To tackle this problem, we have developed methods for data preprocessing and biomarker selection. The preprocessing consists of binning, baseline correction, and normalization. An algorithm, Extended Markov Blanket (EMB), is developed for biomarker detection, which combines redundant feature removal and discriminant feature selection. The biomarker selection couples with support vector machine (SVM) to achieve sample prediction from high-resolution proteomic profiles.

Disease progresses in several stages. Therefore, there exist biomarkers corresponding to each stage. To deal with such a multi-class problem, we propose a classification and a feature selection method. The proposed classification method consists of two schemes: error-correcting output coding (ECOC) and pairwise cou-
pling (PWC). In prediction for a test sample, aggregated results of both schemes are considered. In PWC scheme, important features for each pair of classes are found by using extended Markov blanket (EMB) feature selection.

To identify the molecular formulae of the biomarkers, we develop a de novo peptide sequencing method. De novo peptide sequencing that determines the amino acid sequence of a peptide via tandem mass spectrometry (MS/MS) has been increasingly used nowadays in proteomics for protein identification. Current de novo methods generally employ a graph theory, which usually produces a large number of candidate sequences and causes heavy computational cost while trying to determine a sequence with less ambiguity. We present a novel de novo sequencing algorithm that greatly reduces the number of candidate sequences. By utilizing certain properties of b- and y-ion series in MS/MS spectrum, we propose a reliable two-way parallel searching algorithm to filter out the peptide candidates that are further pruned by an intensity evidence based screening criterion.

LDA is a traditional statistical scheme for feature reduction which has been widely used in a diversity of application areas. In a case where the dimensionality exceeds the sample size, however, the classical LDA faces a problem known as singularity. Since the dimensionality of the mass spectrometry data is considerably huge, the singularity problem necessarily happens. Another drawback of the classical LDA is its linear property with which LDA fails for nonlinear problems. To solve the problem, nonlinear based LDA methods have been proposed. However, they suffer from high cost in running. We propose a new fast kernel discriminant analysis (FKDA).
TABLE OF CONTENTS

ACKNOWLEDGEMENTS .................................................. iv
ABSTRACT ................................................................. vi
LIST OF FIGURES ......................................................... xi
LIST OF TABLES .......................................................... xiii

Chapter

1. PROTEOMIC BIOMARKER DETECTION BASED ON EXTENDED
   MARKOV BLANKET .................................................. 1
   1.1 Introduction ..................................................... 1
   1.2 Materials ......................................................... 5
       1.2.1 Serum Samples ............................................. 5
       1.2.2 Sample Preparation on Chips ......................... 5
   1.3 Background ....................................................... 6
       1.3.1 Markov Blanket (MB) Feature Selection ............ 6
       1.3.2 Support Vector Machines (SVMs) ...................... 8
       1.3.3 SVM-Recursive Feature Elimination (SVM-RFE) ..... 9
   1.4 Preprocessing ................................................... 9
       1.4.1 Binning .................................................... 11
       1.4.2 Baseline Correction .................................... 12
       1.4.3 Normalization ............................................ 13
   1.5 Methodology ..................................................... 13
       1.5.1 Extended Markov Blanket ............................... 13
       1.5.2 Feature Pruning by Backward Feature Selection ... 16
   1.6 Experimental Results .......................................... 17
1.6.1 Preprocessing Results .............................................. 17
1.6.2 Biomarker Selection and Classification Validation ............ 18
1.6.3 Comparison with Other Algorithms .............................. 23
1.7 Conclusions and Discussions ........................................ 24
2. DIAGNOSIS OF EARLY RELAPSE IN OVARIAN CANCER USING
SELDI-TOF MASS SPECTROMETRY DATA .............................. 26
  2.1 Introduction ......................................................... 26
  2.2 SVM-Markov Blanket/Recursive Feature Elimination ............ 27
    2.2.1 Preprocessing ............................................... 27
    2.2.2 Scoring function ........................................... 29
  2.3 Experimental Results .............................................. 29
  2.4 Conclusion ......................................................... 31
3. MULTI-STAGE DISEASE CLASSIFICATION BASED ON
BI-CLASSIFICATION STRATEGY ........................................ 36
  3.1 Introduction ......................................................... 36
  3.2 Methods ........................................................... 39
    3.2.1 Data Preprocessing ........................................... 39
    3.2.2 A Redesigned ECOC Scheme for Multi-Class Classification .. 40
    3.2.3 Pairwise Coupling (PWC) Scheme ............................. 44
    3.2.4 Extended Markov Blanket Feature Selection .................. 45
    3.2.5 Feature Pruning by Backward Feature Selection ............ 48
    3.2.6 Retraining ................................................... 49
  3.3 Experiments ......................................................... 49
  3.4 Conclusion ......................................................... 56
4. TWO-WAY SEARCH FOR BIOMARKER IDENTIFICATION ............ 57
  4.1 Introduction ......................................................... 57
# LIST OF FIGURES

<table>
<thead>
<tr>
<th>Figure</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.1 An example that shows a spectrum after binning</td>
<td>10</td>
</tr>
<tr>
<td>1.2 MALDI-TOF mass spectrum in the range between 1k and 10 kDa</td>
<td>11</td>
</tr>
<tr>
<td>1.3 Diagram for Extended Markov Blanket algorithm</td>
<td>14</td>
</tr>
<tr>
<td>1.4 Relative weights of the top 60 ranked biomarkers</td>
<td>19</td>
</tr>
<tr>
<td>1.5 Performance measurements of Extended Markov blanket algorithm</td>
<td>21</td>
</tr>
<tr>
<td>1.6 Average intensity of platinum-resistant/sensitive samples</td>
<td>22</td>
</tr>
<tr>
<td>2.1 Example that shows the change of the number of peaks</td>
<td>33</td>
</tr>
<tr>
<td>2.2 Average measurements changing the size of feature subset</td>
<td>34</td>
</tr>
<tr>
<td>2.3 Frequency of the 58 candidate peaks and t-test values</td>
<td>35</td>
</tr>
<tr>
<td>3.1 Framework of the proposed algorithm</td>
<td>39</td>
</tr>
<tr>
<td>3.2 Flowchart of the algorithm ($p = 20$ in this study)</td>
<td>48</td>
</tr>
<tr>
<td>3.3 Comparison of overall classification accuracies</td>
<td>51</td>
</tr>
<tr>
<td>3.4 The top 20 peaks out of 300 peaks and grouping of peaks</td>
<td>52</td>
</tr>
<tr>
<td>3.5 Frequency for the optimal feature sets</td>
<td>53</td>
</tr>
<tr>
<td>3.6 Average intensities for the top 20 peaks</td>
<td>55</td>
</tr>
<tr>
<td>4.1 Structure of a peptide consisting of four amino acids</td>
<td>58</td>
</tr>
<tr>
<td>4.2 Hypothetical MS/MS spectrum and amino acid sequences</td>
<td>63</td>
</tr>
<tr>
<td>4.3 Measured mass-charge ratios and normal distribution</td>
<td>67</td>
</tr>
<tr>
<td>4.4 Two-way searching algorithm</td>
<td>69</td>
</tr>
<tr>
<td>4.5 Example of peptide sequence YIPGTK in the graph extension</td>
<td>75</td>
</tr>
<tr>
<td>5.1 Mass spectrum of a sample over the mass range 1k-10k Da</td>
<td>80</td>
</tr>
</tbody>
</table>
5.2 Error rates in test samples of each class ......................... 89
5.3 Averaged elapsed time in a run to calculate $\alpha$ ................... 91
## LIST OF TABLES

<table>
<thead>
<tr>
<th>Table</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.1 Top ranked 60 features</td>
<td>18</td>
</tr>
<tr>
<td>1.2 Feature removal by backward feature selection</td>
<td>20</td>
</tr>
<tr>
<td>1.3 Performance comparison with other feature selection algorithms</td>
<td>23</td>
</tr>
<tr>
<td>1.4 Comparison of the individually top 60 features</td>
<td>24</td>
</tr>
<tr>
<td>2.1 Serum sample information</td>
<td>29</td>
</tr>
<tr>
<td>2.2 Change of the number of peaks</td>
<td>30</td>
</tr>
<tr>
<td>2.3 Measurements comparison when $k = 1$ and $k = 2$</td>
<td>32</td>
</tr>
<tr>
<td>3.1 The means and standard deviations (in parenthesis) of accuracies</td>
<td>50</td>
</tr>
<tr>
<td>4.1 Upper tail percentage points for Anderson-Darling statistic $A^*$</td>
<td>65</td>
</tr>
<tr>
<td>4.2 Normality test for distribution of measured mass-charge ratios</td>
<td>66</td>
</tr>
<tr>
<td>4.3 Experimental results of peptide sequencing with $\beta = 35%$</td>
<td>73</td>
</tr>
<tr>
<td>4.4 The number of candidates and rankings</td>
<td>74</td>
</tr>
<tr>
<td>5.1 The mean and standard deviation (in parenthesis) of accuracies</td>
<td>88</td>
</tr>
</tbody>
</table>
CHAPTER 1

PROTEOMIC BIOMARKER DETECTION BASED ON EXTENDED
MARKOV BLANKET

1.1 Introduction

To study the heterogeneity of diverse protein molecules during disease process, proteomics has been recognized as a major technology to simultaneously screen for multiple biomarkers from patient specimen [1], [2]. Due to the perfusion of blood to each organ and tissue, serum proteomic pattern may reflect the abnormality or pathologic state of various diseases [3]. Patient serum protein profiling has been reported as a promising tool to achieve early disease detection, in which the procedure is simple, inexpensive, and minimally invasive [4], [5].

Among different serum profiling technologies, low resolution surface-enhanced laser desorption/ionization time-of-flight (SELDI-TOF) mass spectrometry has been used to detect protein patterns of normal, premalignant and malignant cells found in breast [2], prostate [6], [7], ovarian [8], and lung cancers [9]. High-mass protein products are identified by the low-resolution SELDI-TOF. However, low-molecular weight (LMW) end of the proteomic spectrum (proteins and peptides below 40,000 Dalton) contains an abundance of potential biomarkers [10], [11] and can not be detected by such technology. Therefore, high-resolution mass spectrometry is desired to detect the LMW molecules for better protein pattern identification.

Toward the goal of LMW biomarker detection, high-resolution MALDI-TOF (matrix-assisted laser desorption/ionization time-of-flight) mass spectrometry is capable of collecting data over a broad mass range (100 to < 300,000 Dalton) in
a single acquisition and has less measurement error (mass shift). This technique results in higher accuracy because of the significantly large number of ion peaks. High-resolution MALDI-TOF mass spectrometry has increasingly been used to early disease diagnosis, monitoring disease progression and therapeutic effects of drugs. To discover and identify unique biomarker patterns hidden in the complex and high-dimensional mass spectra, robust computation algorithms have to be developed.

Numerous pattern recognition algorithms have been developed for low-resolution SELDI-TOF data analysis. Early research efforts used genetic algorithm (GA) and self-organizing map (SOM) to identify proteomic patterns to discriminate ovarian cancer samples from normal [6]. Candidate subsets containing 5 to 20 of 15,200 m/z (mass/charge) values were randomly selected. After examining the ability of subsets to distinguish samples, the m/z values contained in the highest rated sets were reshuffled to form new subset candidates. This process was performed iteratively until the discriminative feature set emerges. GA/k-nearest neighbors (GA/KNN) used in microarray expression analysis was also applied to the identification of ovarian cancer [12]. The top ten most discriminative m/z values below 500 were selected. Since such low m/z values are likely to contain ionized organic acid matrix from surface coatings and are generally discarded as noise, further experiments were conducted to omit such m/z values. Methods based on decision trees were successfully used in several studies such as prostate [7], [13] and ovarian cancer [14]. For the analysis of prostate cancer, the area under the ROC curve (AUC) was used as the early filtering, i.e., peaks with an AUC < 0.62 were excluded from further data analysis. For ovarian cancer analysis, peak clustering was performed using the Biomarker Wizard software (Ciphergen Biosystems). Artificial neural network (ANN) known as a powerful tool for the analysis of complex data containing a high level of background noise was employed for molecular ion identification from SELDI-TOF mass spectrometry data.
[15]. It was showed that this technique facilitates rapid identification of validated biomarkers.

For high-resolution mass spectrometry (MS) data, traditional machine learning algorithm may break down with high-dimensional input. Very limited research has been carried out to explore the computational challenges. Among current research endeavors, majority of the work has been focused on classifier design and there is a lot to be investigated on feature dimensionality reduction due to the high dimension nature of MALDI-TOF data. Ressom et al. designed a computational method that combines particle swarm optimization with SVM (Support Vector Machine) to distinguish liver cancer patients from healthy individuals in SELDI-QqTOF spectra [16]. Particle swarm optimization and ant colony optimization are interesting swarm intelligence techniques which have been successfully applied to a number of optimization problems. A comparative study of several well-known classification algorithms, such as linear discriminant analysis (LDA), k-nearest neighbor (KNN), random forest (RF), bagging, boosting, and support vector machine (SVM), has been carried out for ovarian cancer diagnosis [17]. MALDI-TOF mass spectrometry was used to obtain the dataset. It was demonstrated that RF approach leads to an overall higher accuracy rate as well as a more stable assessment for classification errors.

In this paper, we use high-resolution MALDI-TOF mass spectrometry for biomarker profiling to predict recurrent ovarian cancer. Ovarian cancer is commonly diagnosed at stage III or IV with a low five-year survival rate [18]. Primary therapy of ovarian cancer includes surgical cytoreduction followed by chemotherapy with a platinum agent. Three-quarters of the patients will eventually suffer a relapse within a few months or several years later [19]. If cancer does not recur and disease remits for six months or more since completion of primary chemotherapy, the cancer is considered “platinum-sensitive.” On the other hand, if cancer relapses
within less than six months of completing the primary therapy, the cancer is considered “platinum-resistant” and is diagnosed as “early relapse” [20]. Patients with platinum-sensitive disease are usually treated again with the primary chemotherapy used before, while patients with recurrent platinum-resistance cancer are usually not responsive to standard therapy. Therefore many new secondary-chemotherapy drugs have been developed in recent years for re-treatment of these cases. Unfortunately, recurrent ovarian cancer is relatively difficult to be diagnosed. And there is currently no reliable technique for predicting early relapse in ovarian cancer. Hence, new methods such as MALDI-TOF are urgently needed to help physicians and gynecological oncologists to predict the early relapse and to give targeted therapy to patients at high risk of recurrent ovarian cancer.

One computational challenging in analyzing MALDI-TOF data is the high-dimension of ion peaks (a.k.a. features). To handle the large number of peaks in high-resolution MALDI-TOF data [21], we have developed a multi-step feature selection algorithm. Our framework starts with a preprocessing step composed of binning, baseline correction, and normalization. Then a new feature subset selection algorithm, \textit{Extended Markov Blanket Filtering}, is presented. Markov blanket is an information-theory-based method for feature subset selection. It eliminates features having little or no information beyond what is subsumed by the remaining features [22], [23]. Instead of exhaustive search, heuristic Markov blanket is efficient and theoretically optimal. Our method finds two feature subsets relevant to each feature: low and high correlated feature subsets. The high correlated feature subset is used in Markov blanket and contributes to the removal of redundant features. On the other hand, the low correlated feature subset is utilized to assign a feature a weight to represent the extent that the feature contributes sample classification. The per-
formance of our method is demonstrated by comparison with other common-used feature selection methods.

1.2 Materials

1.2.1 Serum Samples

Human sera were collected from 113 patients undergoing gynecologic oncology surgery at the University of Texas Southwestern Medical Center from year 2000 to present using an institutional review board (IRB) approved protocol. Every serum was aliquoted and frozen at -80°C.

1.2.2 Sample Preparation on Chips

Serum samples were processed by using ProXPRESSION™ Biomarker Enrichment Kits (Perkin Elmer). Briefly, this system uses a Cibachron blue (CB) dye affinity-chromatography-based technology that is designed to capture high-abundance carrier proteins (such as albumin) in blood and then enrich for the peptide and protein fragments bound to the carrier proteins. CB filtration plates were purchased from PerkinElmer. ZipPlates™ and the vacuum manifold were purchased from Millipore. Millipore also provided custom-fitting adapters for direct spotting of samples on single-use MALDIchip™ Target plates (PerkinElmer). Serum samples were processed with CB filtration plates. CB filtration plates were washed 3 times with 400 µL of Sample Binding Buffer (SBB). A 15-µL portion of each serum sample was diluted with 150 µL of SBB. Each diluted sample was then loaded on a column and 7.5 mmHg of vacuum was applied. The column was washed with 400 µL of SBB, and the biomarkers were eluted with 200 µL of Sample Elution Buffer (Perkin Elmer) using the vacuum manifold. The elution step was repeated once more, and the two-elution volumes were co-eluted in the same 96 well-format, 1 ml capacity plate.
Eluted biomarkers were concentrated and desalted on ZipPlate and applied directly to MALDIchip targets by vacuum elution. The entire elution volume from each CB well was loaded into 1 well of a ZipPlate, and the sample bound by applying vacuum (12.5 mmHg). The biomarkers were eluted directly by vacuum onto disposable MALDIchip targets. The ZipPlates were lifted from the MALDIchip Target plate, leaving small droplets on the plate, indicating successful sample transfer. Samples are allowed to air dry at room temperature, which leads to the formation of matrix crystals. Mass spectra were acquired using a prOTOF® 2000 matrix-assisted laser desorption/ionization orthogonal time-of-flight (MALDI O-TOF) MS interfaced with TOFWorks® software (PerkinElmer/SCIEX). Because of the orthogonal design, a single external mass calibrant was used to achieve better than 10-ppm mass accuracy over an entire sample plate (up to 96 samples). In this study, a 2-point external calibration of the prOTOF instrument was performed before acquiring the spectra in a batch mode from 96 samples.

1.3 Background

1.3.1 Markov Blanket (MB) Feature Selection

Markov blanket filtering is an instance of backward feature elimination algorithm [22], [23]. Let $\mathbf{F}$ be a set of features with size $r$ defined as $\mathbf{F} = (F_1, \ldots, F_r)$ and $\mathbf{M} \subseteq \mathbf{F}$ be a set of features which does not contain $F_i$. Feature set $\mathbf{M}$ is called Markov blanket for $F_i$ if $F_i$ is conditionally independent of $\mathbf{F} - \mathbf{M} - \{F_i\}$ given $\mathbf{M}$. Therefore, the information contained in feature $F_i$ can be covered by its Markov blanket. However, since the full size Markov blanket may not be available, an approximate one that subsumes the feature information has to be sought. One Markov blanket $\mathbf{M}_i$ for $F_i$ can be defined as the one having $m$ highest Pearson correlations.
with $F_i$. In general, to reduce computational overhead and to avoid fragmenting the training samples, small value $m$ is used.

To evaluate the closeness between $F_i$ and its Markov blanket $M_i$, the following expected cross-entropy is estimated:

$$
\triangle(F_i|M_i) = \sum_{f_{M_i}, f_i} P(M_i = f_{M_i}, F_i = f_i) \times D(P(c|M_i = f_{M_i}, F_i = f_i) \| P(c|M_i = f_{M_i}))
$$

(1.1)

where $f_{M_i}$ and $f_i$ are feature values to $M_i$ and $F_i$, respectively, $c$ is class label, and $D(\cdot \| \cdot)$ represents cross-entropy. For any distributions $\mu$ and $\sigma$, the cross-entropy of $\mu$ to $\sigma$ is $D(\mu\|\sigma) = \sum_{x \in \Omega} \mu(x) \log \frac{\mu(x)}{\sigma(x)}$ that measures the extent of the difference which is made by using $\sigma$ instead of $\mu$. In Eq. (1.1), $\triangle(F_i|M_i) = 0$ means that $M_i$ is a perfect Markov blanket for $F_i$, therefore $F_i$ does not provide any information about class labels beyond that subsumed by its Markov blanket $M_i$. However, since this case is less likely to happen, we look for a set $M_i$ such that $\triangle(F_i|M_i)$ is small. The lower $\triangle(F_i|M_i)$ means that the approximate Markov blanket of $F_i$ is strongly correlated to $F_i$. The feature $F_i$ with the lowest $\triangle(F_i|M_i)$ value in the remaining features is considered to be the most redundant, and should be eliminated first.

To decide the Markov blanket of each feature, intensity values after the preprocessing task are used in the calculation of Pearson correlation coefficient. (Details for the preprocessing will be provided in Section 1.4.) For computational convenience, the discretized binary values are used in the calculation of the expected cross-entropy $\triangle(F_i|M_i)$ [24]. In discretization of feature $A$ value, suppose that there is a given set $S$ of all samples and a partition boundary $T$ by which $S$ is partitioned into two subsets $S_1$ and $S_2$. Let $P(c_i, S_j)$ be the proportions of samples in subset $S_j$ that belong to class $c_i$. Then the class information entropy of the partition is defined by:

$$
E(A, T, S) = \frac{|S_1|}{|S|} Ent(S_1) + \frac{|S_2|}{|S|} Ent(S_2),
$$

(1.2)
where $\text{Ent}(S_j) = - \sum_{i=1}^{k} P(c_i, S_j) \log(P(c_i, S_j))$ is the class entropy for subset $S_j$, $k$ is the number of classes and $|\cdot|$ represents the number of samples in the subset. The cut point $T_A$ for which $E(A, T, S)$ is minimal among all the candidate cut points is selected for a binary discretization for feature $A$ [25]. Finally, the expected cross-entropy $\triangle(F_i|M_i)$ can be calculated as:

$$\triangle(F_i|M_i) = \sum_{j=1}^{n} P_j(M_i, F_i) \times \sum_{l=1}^{k} P_j(c_l|M_i, F_i) \log\left(\frac{P_j(c_l|M_i, F_i)}{P_j(c_l|M_i)}\right), \quad (1.3)$$

where $n$ is the number of samples [26].

### 1.3.2 Support Vector Machines (SVMs)

SVMs are kernel based learning algorithms to solve two-class classification problems [27]. An optimal hyperplane is sought to separate a given set of binary labeled training data by maximizing the margin between the two classes. To do so, SVMs map the training data $x$ into a higher dimensional space via a mapping function $\Phi(x)$ and construct a decision function as:

$$f(x) = \langle w, \Phi(x) \rangle + b, \quad (1.4)$$

where $w$ is a weight vector and $b$ is a scalar.

Suppose there are $n$ training samples $\{(x_i, y_i), 1 \leq i \leq n\}$ where $x_i$ is the $i^{th}$ training sample consisting of an $r$-dimensional feature vector and $y_i \in \{-1, 1\}$ is the class label of $x_i$. The problem of finding the optimal hyperplane can be formulated as the following quadratic programming problem

$$\min_{\alpha_i} \frac{1}{2} \sum_{i=1}^{n} \sum_{j=1}^{n} \alpha_i \alpha_j y_i y_j K(x_i, x_j) - \sum_{i=1}^{n} \alpha_i, \quad (1.5)$$
subject to
\[ 0 \leq \alpha_i \leq C, \quad \sum_{i=1}^{n} \alpha_i y_i = 0, \] (1.6)
where \( K(x_i, x_j) \) is a kernel function, \( \alpha \) is a Lagrange multiplier and \( C \) is a user defined soft-margin constant. In linear SVM, we compute the weight vector as:
\[ w = \sum_{i=1}^{n} \alpha_i^* y_i x_i, \] (1.7)
where \( \alpha_i^* \) is solution to Eq. (1.5).

### 1.3.3 SVM-Recursive Feature Elimination (SVM-RFE)

SVM-RFE proposed by Guyon et al. is a backward feature elimination algorithm based on SVM [28]. At each iteration, weights for all existing features are obtained by using Eq. (1.7) and a feature corresponding to a smallest absolute weight is eliminated. This procedure continues until only one feature remains so that in the end all features are ranked.

### 1.4 Preprocessing

All samples used in this study were analyzed by MALDI-TOF mass spectrometry which is as accurate as 10-ppm (parts per million) mass accuracy with a single external mass calibrant. We extracted \( m/z \) values in the range between 1k and 10k Da (Dalton) from each spectrum of 726,343 \( m/z \) peaks. Since such a huge dimensionality causes considerable cost burden in feature selection and classification problem, most literatures using MALDI-TOF data perform a preprocessing work with the raw spectra data. We have designed a series of preprocessing steps to reduce input signal dimensionality and to remove noise of the raw spectra.
Figure 1.1. An example that shows a spectrum after binning. A few peaks that seem to be monoisotope and its isotopes are converted into a new peak in the same m/z range. Red line indicates the baseline. (a) raw spectrum; (b) binned spectrum.
1.4.1 Binning

In the first step of data preprocessing, binning is performed to divide the \( m/z \) axis into intervals of desired length. Prior to the task, we need to decide a starting \( m/z \) where the dividing begins. Moving from the lowest \( m/z \) of the spectrum to the right along the \( m/z \) axis, a point at which intensity changes from decreasing to increasing becomes the starting \( m/z \). Let the starting point be \( SP \). From the \( SP \), we proceed to the right with a step of 100 Da and search for the minimum
intensity within a given small $m/z$ clearance from $SP + 100$ $m/z$. Let the position of the minimum intensity be $EP$ and the range between $SP$ and $EP$ be a window. Then, the window is divided by 100 yielding 100 bins so that the size of each bin is approximately 1 Da and the window consists of 100 bins. In each bin, peaks whose intensities are less than 10% of the maximum intensity of the bin, which tend to be noise, are removed. A mean value of all the remaining intensities in each bin is placed at the middle position of the bin indicating the representative intensity. In the similar way, we continue to proceed to the direction of increasing $m/z$ values to find the next window starting from the $EP$ of the first window, which becomes the new $SP$ of the second window. This procedure is repeated until the end of the spectrum. The preprocessing step is individually performed for each spectrum. Without loss of generality, the first spectrum after preprocessing serves as the reference, by which all the bin and window positions of other spectra will be aligned with. Fig. 1.1 demonstrates a zoomed-in local segment of the spectrum before and after binning.

1.4.2 Baseline Correction

Baseline in mass spectrometry data is caused by the chemical noise in the matrix. For baseline correction, a window is divided into 10 groups with each group consisting of 10 bins. A minimum intensity and its $m/z$ ratio are found in each group. Therefore, there exist 10 pairs of intensity and $m/z$ ratio in each window. We estimate the baseline by fitting a fourth order polynomial to the 10 pairs of intensity and $m/z$ ratio. The baseline correction is performed for each window. Finally, the overall regressed baseline is subtracted from the binned spectrum.
1.4.3 Normalization

Furthermore, we choose to normalize the baseline corrected spectrum because the amount of proteins in blood sample changes from patient to patient and even to the same patient with samples drawn at different times. Each baseline corrected $m/z$ value is normalized by the total ion current of the spectrum. The total ion current is the summed intensity over all $m/z$ values. Due to the very small normalized intensity value, all the intensities are multiplied by 10,000 for computational convenience. The consecutive steps of the preprocessing are shown in Fig. 1.2. It can be seen that the significant peaks are retained with smoothed and de-noised effects.

1.5 Methodology

1.5.1 Extended Markov Blanket

In this section, we will present a new feature selection algorithm called extended Markov blanket. Our algorithm considers reducing redundant features while selecting the most discriminant ones. To be more specific, for a feature $F_i$ remained after preprocessing, two feature subsets are considered: the high correlated feature subset (HCFS) and the low correlated feature subset (LCFS) composed of the least correlated $d$ features for $F_i$. HCFS feature subset is used to remove redundant features as in the classical Markov blanket feature selection. Our contribution comes from utilizing LCFS to estimate the classification capability of each feature during the Markov blanket process. This is derived from the fact that mutually low correlated features, in general, lead to good classification performance.

We will now describe how the extended Markov blanket algorithm works out for feature selection. We choose 10-fold cross validation for performance evaluation where all samples are randomly split into 10 exclusive folds. For each of the ten
Figure 1.3. Diagram for Extended Markov Blanket algorithm. In the first Markov blanket run of the first iteration, the expected cross-entropy $\Delta(f_2|\{f_6, f_8\})$ for feature $f_2$ is the smallest value. Therefore, $f_2$ is removed from both LCFS and HCFS for all features. LCFS and HCFS of survival features are then rebuilt.
experiments, typically a single fold is retained as a validation data, and the remaining nine folds are used as train data. In our implementation, we divide the train data into sub-train and sub-test data in the ratio of 80%:20%. For each feature \( F_i \), we perform Markov blanket algorithm with its HCFS obtained from the train data to compute the expected cross-entropy value, \( \Delta(F_i|M_i) \). On the other hand, using LCFS and feature \( F_i \), we run the linear SVM algorithm where the sub-train and sub-test data are used for training and testing, respectively, to obtain an roughly estimated accuracy, denoted as \( \beta \). Then, we compute the normalized weight for each feature in the LCFS using the following function:

\[
  z_k = \begin{cases} 
  \frac{|w_k|}{\sum_{j=1}^{d+1}|w_j|} \times \beta \times \delta & \text{if } 0.5 < \beta, \\
  \left(1 - \frac{|w_k|}{\sum_{j=1}^{d+1}|w_j|}\right) \times (0.5 - \beta) \times \delta & \text{otherwise}
  \end{cases}
\]  

\( z_k \) is the normalized weight for feature \( F_i \) in the LCFS, \( |w_k| \) is the absolute SVM weight obtained using Eq. (1.7), and \( d \) is the size of LCFS chosen as 10, 20, or 30. After computing \( |w_k| \) for all features in the LCFS, each \( |w_k| \) is normalized by the summed absolute SVM weights of all the features in the LCFS and the weight of feature \( F_i \).

In SVM-RFE (SVM-Recursive Feature Elimination) method by Guyon et al., features with the smallest absolute SVM weights are eliminated at each iteration for feature selection [28]. Adversely, we assume that features with large absolute SVM weights are important in terms of classification power. Not surprisingly to observe, although the normalized SVM weight of certain feature in Eq. (1.8) may be high in an
LCFS, the classification accuracy of the linear SVM performed with the limited LCFS may be low due to the partial discriminant capacity of one LCFS. Therefore, not only the normalized SVM weight for each feature but also the accuracy with its LCFS are important factors to estimate feature weight. This leads to the multiplication of the normalized SVM weight by the accuracy obtained after the SVM. If the accuracy is greater than 50%, the proposed weight becomes positive by multiplying 1 as a value of $\delta$. Otherwise, we treat it as a penalty using $-1$ as the $\delta$.

After computing the proposed weights and expected cross-entropy $\triangle(F_i|M_i)$ values for all features as introduced above, certain feature with the smallest $\triangle(F_i|M_i)$ value is removed according to Markov blanket rule. Then, the HCFS and LCFS for all but the removed feature are rebuilt. As an example shown in Fig. 1.3, feature $F_2$ with value as $f_2$ was the one with the smallest cross-entropy and was removed first. Consequently feature $f_1$ that had $f_2$ and $f_6$ as members of HCFS now is updated as features $f_6$ and $f_8$. In fact, only those subsets that contain the removed feature will be affected and modified. Again, we compute the $\triangle(F_i|M_i)$ values and the proposed weights for survival features. The feature weights will be accumulated during the Markov blanket feature removal process. The whole procedure keeps going on until the predefined feature number is reached.

1.5.2 Feature Pruning by Backward Feature Selection

The aforementioned feature removal was repeated 100 times for 10-fold cross validations. We compute a mean weight value for each feature after the 100 iterations by dividing each feature’s accumulated weight by the occurrence counting $O_i$ of feature $F_i$ in all LCFS (Fig. 1.3). All the features are then ranked according to the mean weights. Then, we perform linear SVM using the first top ranked feature then the top two and so forth up to the top 60 features. From a feature subset with the
best accuracy, we start the standard backward feature elimination algorithm to find a compact feature subset. The backward feature selection approach removes one feature at a time from the current feature set. Each time a feature without which the accuracy is improved will be excluded. The backward feature selection method continues until one feature remains. The overall steps of Extended Markov Blanket can be seen from Fig. 1.3.

1.6 Experimental Results

We implemented the preprocessing part using Matlab and the proposed algorithm using JAVA based on LIBSVM [29]. Serum samples were collected from 113 ovarian cancer patients who had undergone a surgical operation: 48 for platinum-resistant and 65 for platinum-sensitive.

1.6.1 Preprocessing Results

After the binning preprocessing for the high-resolution MALDI-TOF serum profiles, the number of features, i.e., mass-to-charge ratio peaks, was reduced from 726,343 to 8,900. Despite the considerable reduction of dimensionality, the size of 8,900 features still makes considerable computation burden for afterward feature selection and classification problem. In order to further filter the high-dimensional number of features, we adopted a two-sample t-test to assess the degree of separation between two classes for single features. With the significance level of 5% at each of the \( m/z \) values in all spectra after preprocessing, the number of features was reduced to 1,338. Then, Extend Markov Blanket was applied to the reduced 1,338 features.
Table 1.1. Top ranked 60 features. Weight indicates normalized feature weight. Accuracy, sensitivity and specificity show the performance when different number of top ranked features are used.

<table>
<thead>
<tr>
<th>No</th>
<th>M/Z</th>
<th>Weight</th>
<th>Accuracy</th>
<th>Sensitivity</th>
<th>Specificity</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>1954.8493</td>
<td>100.00</td>
<td>0.6480</td>
<td>0.1625</td>
<td>0.9717</td>
</tr>
<tr>
<td>2</td>
<td>1992.8479</td>
<td>95.36</td>
<td>0.6475</td>
<td>0.1650</td>
<td>0.9692</td>
</tr>
<tr>
<td>3</td>
<td>1575.5422</td>
<td>90.21</td>
<td>0.6491</td>
<td>0.3100</td>
<td>0.9450</td>
</tr>
<tr>
<td>4</td>
<td>1690.6383</td>
<td>89.69</td>
<td>0.7205</td>
<td>0.4212</td>
<td>0.9200</td>
</tr>
<tr>
<td>5</td>
<td>1577.5430</td>
<td>85.05</td>
<td>0.7145</td>
<td>0.4212</td>
<td>0.9100</td>
</tr>
<tr>
<td>6</td>
<td>2573.1582</td>
<td>79.90</td>
<td>0.7285</td>
<td>0.5500</td>
<td>0.8475</td>
</tr>
<tr>
<td>7</td>
<td>2106.8983</td>
<td>76.29</td>
<td>0.7245</td>
<td>0.5388</td>
<td>0.8483</td>
</tr>
<tr>
<td>8</td>
<td>1777.7205</td>
<td>71.65</td>
<td>0.7035</td>
<td>0.5075</td>
<td>0.8342</td>
</tr>
<tr>
<td>9</td>
<td>1554.5333</td>
<td>67.53</td>
<td>0.7025</td>
<td>0.5163</td>
<td>0.8267</td>
</tr>
<tr>
<td>10</td>
<td>1553.5329</td>
<td>66.49</td>
<td>0.6880</td>
<td>0.5013</td>
<td>0.8125</td>
</tr>
<tr>
<td>11</td>
<td>2252.0007</td>
<td>65.46</td>
<td>0.6875</td>
<td>0.5113</td>
<td>0.8050</td>
</tr>
<tr>
<td>12</td>
<td>2196.9008</td>
<td>64.43</td>
<td>0.6880</td>
<td>0.5325</td>
<td>0.7917</td>
</tr>
<tr>
<td>13</td>
<td>1616.5681</td>
<td>63.40</td>
<td>0.6785</td>
<td>0.5250</td>
<td>0.7808</td>
</tr>
<tr>
<td>14</td>
<td>1463.5031</td>
<td>60.31</td>
<td>0.6745</td>
<td>0.5225</td>
<td>0.7758</td>
</tr>
<tr>
<td>15</td>
<td>2572.1574</td>
<td>59.79</td>
<td>0.6765</td>
<td>0.5237</td>
<td>0.7783</td>
</tr>
<tr>
<td>16</td>
<td>1008.3059</td>
<td>59.28</td>
<td>0.6720</td>
<td>0.5413</td>
<td>0.7592</td>
</tr>
<tr>
<td>17</td>
<td>2050.8725</td>
<td>58.25</td>
<td>0.6630</td>
<td>0.5100</td>
<td>0.7650</td>
</tr>
<tr>
<td>18</td>
<td>1131.3002</td>
<td>56.70</td>
<td>0.6675</td>
<td>0.5238</td>
<td>0.7633</td>
</tr>
<tr>
<td>19</td>
<td>2322.0129</td>
<td>55.15</td>
<td>0.6700</td>
<td>0.5363</td>
<td>0.7592</td>
</tr>
<tr>
<td>20</td>
<td>1793.7357</td>
<td>48.97</td>
<td>0.6615</td>
<td>0.5338</td>
<td>0.7467</td>
</tr>
<tr>
<td>21</td>
<td>1688.5664</td>
<td>47.42</td>
<td>0.6810</td>
<td>0.5275</td>
<td>0.7833</td>
</tr>
<tr>
<td>22</td>
<td>2147.9404</td>
<td>45.88</td>
<td>0.6690</td>
<td>0.5100</td>
<td>0.7550</td>
</tr>
<tr>
<td>23</td>
<td>4084.9432</td>
<td>45.88</td>
<td>0.6530</td>
<td>0.5113</td>
<td>0.7475</td>
</tr>
<tr>
<td>24</td>
<td>2148.9415</td>
<td>45.36</td>
<td>0.6440</td>
<td>0.5138</td>
<td>0.7308</td>
</tr>
<tr>
<td>25</td>
<td>1915.7949</td>
<td>43.81</td>
<td>0.6390</td>
<td>0.5125</td>
<td>0.7233</td>
</tr>
<tr>
<td>26</td>
<td>8284.1325</td>
<td>43.81</td>
<td>0.6395</td>
<td>0.5225</td>
<td>0.7175</td>
</tr>
<tr>
<td>27</td>
<td>2275.0035</td>
<td>43.30</td>
<td>0.6345</td>
<td>0.5250</td>
<td>0.7075</td>
</tr>
<tr>
<td>28</td>
<td>2962.3993</td>
<td>42.78</td>
<td>0.6600</td>
<td>0.5600</td>
<td>0.7267</td>
</tr>
<tr>
<td>29</td>
<td>3147.4299</td>
<td>41.75</td>
<td>0.6725</td>
<td>0.5950</td>
<td>0.7242</td>
</tr>
<tr>
<td>30</td>
<td>2195.9898</td>
<td>41.75</td>
<td>0.6905</td>
<td>0.5862</td>
<td>0.7250</td>
</tr>
</tbody>
</table>

1.6.2 Biomarker Selection and Classification Validation

The size of LCFS used in extended Markov blanket is 10. Using 20 and 30 features yielded similar results. For Markov blanket, size of 2 was used. Table 1.1 lists the top ranked 60 features, their relative weights, and the corresponding \( m/z \) values. Classification performance using different number of top ranked features is also listed. In this study, accuracy refers to the total number of correctly classified platinum-sensitive and platinum-resistant samples. Sensitivity is defined as the percentage of platinum-resistant samples that are correctly classified, while specificity refers to the percentage of platinum-sensitive samples that are correctly classified.
We can see the best performance is obtained when the top 58 ranked features were used with 81.95% accuracy, 80.13% sensitivity and 83.17% specificity. Fig. 1.4 shows the visual graph of different feature weights.

To further improve the sample prediction performance and obtain a compact biomarker set, the backward feature elimination algorithm was performed. The algorithm starts with the current selected 58 features and removes one at a time without which the best accuracy using the remaining features is obtained. In Table 1.2, the bottom row shows the performance when the first feature, which happens to be feature with $m/z$ of 1575.5422 and rank of 3, was removed. From the table we can see, the next removed feature is the one ranked 17. The order of removed features and relevant performances are displayed from bottom to top in Table 1.2. From the table row, we can see Extended Markov blanket obtained overall best performance of 96.75% accuracy, 96.63% sensitivity and 96.83% specificity with 20 features.

The sample classification can be visualized in Fig. 1.5 when different number of features was used in the backward feature pruning process. As shown in Fig. 1.5, with four features, extended Markov blanket achieved 78.7% accuracy, 79.37% sensitivity
Table 1.2. Feature removal by backward feature selection. The order of removed features is shown from bottom up.

<table>
<thead>
<tr>
<th>No</th>
<th>M/Z</th>
<th>Accuracy</th>
<th>Sensitivity</th>
<th>Specificity</th>
</tr>
</thead>
<tbody>
<tr>
<td>16</td>
<td>1008.3059</td>
<td>0.5830</td>
<td>0.0625</td>
<td>0.9300</td>
</tr>
<tr>
<td>35</td>
<td>3217.4611</td>
<td>0.7080</td>
<td>0.6025</td>
<td>0.7783</td>
</tr>
<tr>
<td>41</td>
<td>2871.2989</td>
<td>0.7310</td>
<td>0.7113</td>
<td>0.7442</td>
</tr>
<tr>
<td>44</td>
<td>4046.9342</td>
<td>0.7870</td>
<td>0.7938</td>
<td>0.7825</td>
</tr>
<tr>
<td>55</td>
<td>1914.7942</td>
<td>0.8320</td>
<td>0.8388</td>
<td>0.8275</td>
</tr>
<tr>
<td>48</td>
<td>1740.6856</td>
<td>0.8530</td>
<td>0.8713</td>
<td>0.8408</td>
</tr>
<tr>
<td>19</td>
<td>2322.0129</td>
<td>0.8535</td>
<td>0.8363</td>
<td>0.8650</td>
</tr>
<tr>
<td>14</td>
<td>1463.5031</td>
<td>0.8700</td>
<td>0.8150</td>
<td>0.9067</td>
</tr>
<tr>
<td>1</td>
<td>1994.8493</td>
<td>0.8720</td>
<td>0.8188</td>
<td>0.9075</td>
</tr>
<tr>
<td>53</td>
<td>1009.3066</td>
<td>0.8880</td>
<td>0.8500</td>
<td>0.9133</td>
</tr>
<tr>
<td>57</td>
<td>8878.4549</td>
<td>0.9065</td>
<td>0.8725</td>
<td>0.9292</td>
</tr>
<tr>
<td>33</td>
<td>5001.5661</td>
<td>0.9100</td>
<td>0.9025</td>
<td>0.9150</td>
</tr>
<tr>
<td>49</td>
<td>2149.9425</td>
<td>0.8990</td>
<td>0.8825</td>
<td>0.9100</td>
</tr>
<tr>
<td>37</td>
<td>2297.0062</td>
<td>0.9180</td>
<td>0.9163</td>
<td>0.9192</td>
</tr>
<tr>
<td>34</td>
<td>1440.4985</td>
<td>0.9480</td>
<td>0.9438</td>
<td>0.9508</td>
</tr>
<tr>
<td>15</td>
<td>2572.1574</td>
<td>0.9560</td>
<td>0.9663</td>
<td>0.9492</td>
</tr>
<tr>
<td>30</td>
<td>2195.9898</td>
<td>0.9555</td>
<td>0.9625</td>
<td>0.9508</td>
</tr>
<tr>
<td>36</td>
<td>1576.5426</td>
<td>0.9580</td>
<td>0.9588</td>
<td>0.9575</td>
</tr>
<tr>
<td>39</td>
<td>3232.4618</td>
<td>0.9660</td>
<td>0.9688</td>
<td>0.9642</td>
</tr>
<tr>
<td>51</td>
<td>3597.6778</td>
<td>0.9675</td>
<td>0.9663</td>
<td>0.9683</td>
</tr>
<tr>
<td>29</td>
<td>3147.4299</td>
<td>0.9635</td>
<td>0.9763</td>
<td>0.9550</td>
</tr>
<tr>
<td>25</td>
<td>1915.7949</td>
<td>0.9570</td>
<td>0.9675</td>
<td>0.9500</td>
</tr>
<tr>
<td>52</td>
<td>1913.7935</td>
<td>0.9505</td>
<td>0.9575</td>
<td>0.9458</td>
</tr>
<tr>
<td>2</td>
<td>1992.8479</td>
<td>0.9430</td>
<td>0.9538</td>
<td>0.9358</td>
</tr>
<tr>
<td>44</td>
<td>5085.3824</td>
<td>0.9530</td>
<td>0.9563</td>
<td>0.9508</td>
</tr>
<tr>
<td>7</td>
<td>2106.8989</td>
<td>0.9500</td>
<td>0.9613</td>
<td>0.9425</td>
</tr>
<tr>
<td>32</td>
<td>1206.4049</td>
<td>0.9460</td>
<td>0.9350</td>
<td>0.9533</td>
</tr>
<tr>
<td>42</td>
<td>8282.1306</td>
<td>0.9525</td>
<td>0.9550</td>
<td>0.9508</td>
</tr>
<tr>
<td>4</td>
<td>1690.6383</td>
<td>0.9580</td>
<td>0.9575</td>
<td>0.9583</td>
</tr>
<tr>
<td>8</td>
<td>1777.7205</td>
<td>0.9530</td>
<td>0.9538</td>
<td>0.9525</td>
</tr>
<tr>
<td>38</td>
<td>2842.2901</td>
<td>0.9660</td>
<td>0.9713</td>
<td>0.9625</td>
</tr>
<tr>
<td>24</td>
<td>2148.9415</td>
<td>0.9555</td>
<td>0.9525</td>
<td>0.9575</td>
</tr>
<tr>
<td>26</td>
<td>8284.1325</td>
<td>0.9565</td>
<td>0.9563</td>
<td>0.9567</td>
</tr>
<tr>
<td>18</td>
<td>1131.3802</td>
<td>0.9610</td>
<td>0.9788</td>
<td>0.9492</td>
</tr>
<tr>
<td>58</td>
<td>1741.6865</td>
<td>0.9625</td>
<td>0.9663</td>
<td>0.9600</td>
</tr>
<tr>
<td>46</td>
<td>1742.6875</td>
<td>0.9625</td>
<td>0.9550</td>
<td>0.9675</td>
</tr>
<tr>
<td>56</td>
<td>1057.3399</td>
<td>0.9550</td>
<td>0.9575</td>
<td>0.9533</td>
</tr>
<tr>
<td>28</td>
<td>2962.3593</td>
<td>0.9620</td>
<td>0.9575</td>
<td>0.9650</td>
</tr>
<tr>
<td>50</td>
<td>4772.4132</td>
<td>0.9510</td>
<td>0.9600</td>
<td>0.9450</td>
</tr>
<tr>
<td>12</td>
<td>2196.9088</td>
<td>0.9465</td>
<td>0.9500</td>
<td>0.9442</td>
</tr>
<tr>
<td>43</td>
<td>4224.0366</td>
<td>0.9575</td>
<td>0.9688</td>
<td>0.9500</td>
</tr>
<tr>
<td>22</td>
<td>2147.9404</td>
<td>0.9470</td>
<td>0.9625</td>
<td>0.9367</td>
</tr>
<tr>
<td>10</td>
<td>1553.5329</td>
<td>0.9400</td>
<td>0.9500</td>
<td>0.9333</td>
</tr>
<tr>
<td>47</td>
<td>8283.1316</td>
<td>0.9410</td>
<td>0.9425</td>
<td>0.9400</td>
</tr>
<tr>
<td>20</td>
<td>1793.7357</td>
<td>0.9415</td>
<td>0.9488</td>
<td>0.9367</td>
</tr>
<tr>
<td>27</td>
<td>2275.0035</td>
<td>0.9230</td>
<td>0.9200</td>
<td>0.9250</td>
</tr>
<tr>
<td>40</td>
<td>2298.0063</td>
<td>0.9275</td>
<td>0.9200</td>
<td>0.9325</td>
</tr>
<tr>
<td>6</td>
<td>2573.1582</td>
<td>0.9090</td>
<td>0.9013</td>
<td>0.9142</td>
</tr>
<tr>
<td>11</td>
<td>2252.0077</td>
<td>0.9190</td>
<td>0.9088</td>
<td>0.9258</td>
</tr>
<tr>
<td>54</td>
<td>4152.9948</td>
<td>0.9085</td>
<td>0.9125</td>
<td>0.9058</td>
</tr>
<tr>
<td>9</td>
<td>1554.5333</td>
<td>0.8995</td>
<td>0.8963</td>
<td>0.9017</td>
</tr>
<tr>
<td>13</td>
<td>1616.5681</td>
<td>0.8985</td>
<td>0.8950</td>
<td>0.9008</td>
</tr>
<tr>
<td>45</td>
<td>4165.0030</td>
<td>0.8780</td>
<td>0.8775</td>
<td>0.8783</td>
</tr>
<tr>
<td>21</td>
<td>1688.6364</td>
<td>0.8730</td>
<td>0.8713</td>
<td>0.8742</td>
</tr>
<tr>
<td>5</td>
<td>1577.5430</td>
<td>0.8585</td>
<td>0.8513</td>
<td>0.8633</td>
</tr>
<tr>
<td>31</td>
<td>3089.4179</td>
<td>0.8510</td>
<td>0.8375</td>
<td>0.8600</td>
</tr>
<tr>
<td>17</td>
<td>2050.8725</td>
<td>0.8435</td>
<td>0.8263</td>
<td>0.8550</td>
</tr>
</tbody>
</table>
Figure 1.5. Performance measurements of Extended Markov blanket algorithm. The x-axis represents the number of peaks (features) used. 83.2% and 90.65% accuracies were obtained with 5 and 11 features, respectively. With 20 features, the best accuracy of 96.75% was achieved.

and 78.25% specificity. This result, though not the best one out of our method, is better in accuracy measurement than other feature selection algorithms which is to be presented in the next subsection. Another example of performance measurement is 90.65% accuracy, 87.25% sensitivity and 92.92% specificity with 11 features.

To see how different the intensity of $m/z$ ratios between platinum-resistant and platinum-sensitive samples is, we plotted the average intensity graphs and box plots of the two-class samples for each of the 20 features that were used to obtain the best accuracy in the extended Markov blanket. As can be seen in Fig. 1.6, there is a significant difference between platinum-resistant and platinum-sensitive samples for these selected possible protein/peptide markers.
Figure 1.6. Average intensity of platinum-resistant and platinum-sensitive samples. (a) comparison of average intensity of platinum-resistant and platinum-sensitive samples in each of 20 features which were used to obtain the best accuracy in extended Markov blanket algorithm. The values over bars represent $m/z$. (b) box plot of the 20 features in platinum-resistant and platinum-sensitive cases.
Table 1.3. Performance comparison with other feature selection algorithms such as information gain, relief-F and $\chi^2$-statistic. All algorithms used SVM as a classifier with 10-fold cross validation. MB = Markov blanket

<table>
<thead>
<tr>
<th></th>
<th>Extended MB</th>
<th>Information Gain</th>
<th>Relief-F</th>
<th>$\chi^2$-Statistic</th>
</tr>
</thead>
<tbody>
<tr>
<td>Accuracy(%)</td>
<td>96.75</td>
<td>77.10</td>
<td>72.10</td>
<td>75.25</td>
</tr>
<tr>
<td>Sensitivity(%)</td>
<td>96.63</td>
<td>51.87</td>
<td>59.00</td>
<td>71.50</td>
</tr>
<tr>
<td>Specificity(%)</td>
<td>96.83</td>
<td>93.92</td>
<td>80.83</td>
<td>77.75</td>
</tr>
<tr>
<td>No. of peaks</td>
<td>20</td>
<td>21</td>
<td>5</td>
<td>18</td>
</tr>
</tbody>
</table>

1.6.3 Comparison with Other Algorithms

To validate the performance of our algorithm, we carried out comparison experiments with other algorithms: information gain, relief-F and $\chi^2$ feature selection methods. Table 1.3 presented the best results from individual algorithms. These algorithms were implemented based on codes provided by WEKA package [30]. After running the feature selection algorithms, features were ranked according to the scores from each algorithm. Then, we performed the linear SVM using the 10-fold cross validation from the top one to the top 60 ranked features. Next, backward feature elimination algorithm was used to find the final compact feature list as introduced in Section 1.5.2. As can be seen in Table 1.3, our proposed approach gives overall better performance than other algorithms. We can see in terms of sensitivity and specificity, satisfactory performance was obtained using our proposed method while other methods achieved relatively somewhat good specificity, but poor performance in sensitivity.
Table 1.4. Comparison of the individually top 60 features (peaks) found in different algorithms. For each feature found in extended Markov blanket algorithm, features within 3Da in other algorithms are shown. MB = Markov blanket

<table>
<thead>
<tr>
<th>Extended MB</th>
<th>Information Gain</th>
<th>Relief-F</th>
<th>$\chi^2$-Statistic</th>
</tr>
</thead>
<tbody>
<tr>
<td>1008.3059</td>
<td>1008.3059</td>
<td>1008.3059</td>
<td></td>
</tr>
<tr>
<td>1553.5329</td>
<td>1553.5329</td>
<td>1551.5321</td>
<td>1552.5325</td>
</tr>
<tr>
<td>1554.5333</td>
<td></td>
<td>1552.5325</td>
<td>1553.5329</td>
</tr>
<tr>
<td>1740.6856</td>
<td></td>
<td>1743.6884</td>
<td></td>
</tr>
<tr>
<td>1741.6865</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>1742.6875</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>1994.8493</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>2050.8725</td>
<td>2048.8718</td>
<td>2049.8722</td>
<td>2050.8725</td>
</tr>
<tr>
<td>2147.9404</td>
<td>2148.9415</td>
<td></td>
<td></td>
</tr>
<tr>
<td>2148.9415</td>
<td>2148.9415</td>
<td></td>
<td></td>
</tr>
<tr>
<td>2149.9425</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>3147.4299</td>
<td>3148.4301</td>
<td>3148.4301</td>
<td>3149.4303</td>
</tr>
<tr>
<td>3149.4303</td>
<td></td>
<td>3149.4303</td>
<td></td>
</tr>
<tr>
<td>3597.6778</td>
<td>3598.6779</td>
<td></td>
<td></td>
</tr>
<tr>
<td>5001.5661</td>
<td>5002.5666</td>
<td>5002.5666</td>
<td></td>
</tr>
<tr>
<td>5004.5670</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>8282.1306</td>
<td>8280.1286</td>
<td>8280.1286</td>
<td></td>
</tr>
<tr>
<td>8283.1316</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>8284.1325</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

1.7 Conclusions and Discussions

Computational data analysis is critical for biomarker selection from high-resolution mass spectrometry data where the number of candidates can easily go beyond one million. In this paper we presented a preprocessing method and a new feature selection algorithm for high-resolution MALDI-TOF data. Our algorithm was applied to clinical recurrent ovarian cancer study to find biomarkers causing platinum-sensitive and platinum-resistant. Experimental results showed that our proposed algorithm yields better performance in overall sensitivity and specificity.
compared to other feature selection methods. Prediction of early recurrence in ovarian cancer will help oncologists give targeted therapy to ovarian cancer patients who are usually not responsive to standard therapy. Our algorithm shows the sets of different biomarker patterns and the corresponding prediction results. This information will provide insight and assistive information for alternative treatment and drug design.

In future work, we plan to carry out protein sequencing to identify the molecular formulae for the selected biomarkers found in this study. To reduce experimental cost during sequencing, we will compare the selected biomarkers obtained from different algorithms and start with the most common ones. Currently we investigated the top 60 features which were used in each algorithm of this study. Table 1.4 shows features present within 3Da in other algorithms for each feature found by extended Markov blanket. For example, biomarker with $m/z$ of 1553.5329 can be found by all four different feature selection methods. This indicates the reliability of the marker and this peptide should be among the ones to be first sequenced. We also plan to apply our algorithm to other studies such as prostate cancer and preterm delivery.
2.1 Introduction

Ovarian cancer is commonly diagnosed at stage III or IV with a low 5-year survival rate. Primary therapy of ovarian cancer includes surgical cytoreduction followed by chemotherapy with a platinum agent. Ovarian cancer commonly recurs at the rate of 75% within a few months or several years later [19]. If cancer does not recur and disease remits for 6 months or more since completion of primary chemotherapy, the cancer is considered platinum-sensitive. On the other hand, if cancer relapses within less than 6 months of completing primary therapy, or grows during primary therapy, the cancer is considered platinum-resistant. Platinum-sensitive patients are usually treated again with primary chemotherapy used before, while patients with recurrent platinum-resistance cancer are usually not responsive to standard therapy. Therefore many new secondary-chemotherapy drugs have been found in recent years for re-treatment of them. Unfortunately, recurrent ovarian cancer is relatively difficult to be diagnosed. There is currently no reliable technique for predicting early relapse in ovarian cancer. Hence, new methods in this area are urgently needed to help physicians and gynecological oncologists give targeted therapy to patients with recurrent ovarian cancer.

Recently, surface-enhanced laser desorption/ionization time-of-flight (SELDI-TOF) mass spectrometry has been used successfully to detect protein patterns of several cancers for early disease diagnosis. In addition to being a platform for biomarker
discover, SDLDI-TOF system can be applied for toxicology screening and monitoring of disease progression and therapeutic effects of drugs. An advantage of SELDI-TOF over electrospray ionization (ESI) is a higher tolerance for salts that makes this technique better suited to the examination of biological samples such as serum.

In this paper, we propose a new feature subset selection algorithm, SVM-Markov blanket/recursive feature elimination (SVM-MB/RFE) combining SVM-RFE with Markov blanket filtering. Markov blanket is a feature subset selection method which eliminates features having little or no information beyond that subsumed by the remaining features [23]. Major obstacles in analyzing SELDI-TOF mass spectrometry data are a large number of peaks and mass error [21], [31]. We have developed an efficient software in order to overcome such problems, which consists of feature pruning, binning, normalization and feature selection. In this paper, we demonstrate the better ability of our method over other algorithms through the comparison of performance.

2.2 SVM-Markov Blanket/Recursive Feature Elimination

2.2.1 Preprocessing

In the preprocessing task, we employe t-test as a method to assess the degree of separation between two classes. By using the result, we want to filter the huge number of features (m/z ratios) of mass spectrometry data to reduce computational burden in the next stage of our algorithm. The test is performed at each m/z ratio while yielding its t-test statistic value,

\[ t_i = \frac{\mu_i^+ - \mu_i^-}{\sqrt{\frac{(\sigma_i^+)^2}{n^+} + \frac{(\sigma_i^-)^2}{n^-}}} \]  \hspace{1cm} (2.1)
where + and − stand for two class labels; \( \mu^+_i \) and \( \mu^-_i \) are the means of the \( i^{th} \) feature; \( \sigma^+_i \) and \( \sigma^-_i \) are the corresponding standard deviations; \( n^+ \) and \( n^- \) represent the number of samples contained in each class [32]. At each \( m/z \) ratio, the larger the test statistic in absolute value, the stronger the evidence that there is a difference between the two classes so that we can use the \( m/z \) ratio as a candidate feature to classify samples. In this experiment, we used the significance level of 0.05 as a criterion for filtering peaks.

Since one \( m/z \) and its neighbors are likely to come out from the same molecule and to be strongly correlated each other, the binning work is required. Mass difference between peak \( i \) and peak \( i + 1 \) is calculated for each peak using the following formula,

\[
\beta = \frac{m/z(i + 1) - m/z(i)}{m/z(i)}
\]

(2.2)

where \( m/z(i) \) and \( m/z(i + 1) \) are \( m/z \) values of peak \( i \) and \( i + 1 \), respectively. If \( \beta \) is less than a given threshold, two peaks are considered as belonging to the same bin.

In this study, we perform 5-fold cross validation experiments changing the \( \beta \) value and select the one with which we obtain the best accuracy as the \( \beta \) value.

Usually through normalization, we can expect the better performance of classification algorithm. By doing so, values in each \( m/z \) ratio are converted such that the transformed values lie between 0 and 1. Let \( I_i \) denote the raw intensity at the \( i^{th} \) \( m/z \) position and \( I_{min} \) and \( I_{max} \) denote the smallest and largest intensity, respectively. Then, the normalized intensity \( NI_i \) is calculated by

\[
NI_i = \frac{I_i - I_{min}}{I_{max} - I_{min}}.
\]

(2.3)
Table 2.1. Serum sample information.

<table>
<thead>
<tr>
<th></th>
<th>Resistant</th>
<th>Sensitive</th>
</tr>
</thead>
<tbody>
<tr>
<td># training samples</td>
<td>33</td>
<td>45</td>
</tr>
<tr>
<td># test samples</td>
<td>15</td>
<td>20</td>
</tr>
<tr>
<td># total samples</td>
<td>48</td>
<td>65</td>
</tr>
</tbody>
</table>

2.2.2 Scoring function

We propose a new scoring function combining SVM-RFE weight with Markov blanket scoring value which both are instances of backward feature elimination algorithm. By applying a new score combined from the expected cross-entropy value $\Delta(F_i|M_i)$ of Markov blanket and the weight value $w_i^2$ of SVM-RFE, we hope that Markov blanket helps SVM-RFE select more relevant features by removing redundant and irrelevant features. We use the following score to eliminate a feature at every iteration of SVM-MB/RFE,

$$C_i = \frac{\Delta(F_i|M_i)}{\max_j\{\Delta(F_j|M_j)\}} + \frac{(w_i)^2}{\max_j\{(w_j)^2\}} \quad (2.4)$$

where $C_i$ indicates the final score assigned to $F_i$, which lies between 0 and 2. To have the same range for all the features, the weight value and the expected cross-entropy value are divided by their maximum values.

2.3 Experimental Results

We implemented SVM-MB/RFE algorithm using C++ code. A total of 160 serum samples were run on an IMAC30 ProteinChip™ and CM10 ProteinChip™ arrays from Ciphergen Biosystems. Of them, 113 serum samples (48 platinum-
resistant, 65 platinum-sensitive) as an initially preliminary study were analyzed by SELDI-TOF mass spectrometry. The raw m/z and intensity were exported to an excel file using Biomarker Wizard software (Ciphergen Biosystems). Those samples were randomly divided into 33 and 45 as a training set, respectively, from platinum-resistant and platinum-sensitive samples, and the reminder as a test set. Table 2.1 shows the information of serum samples used in this study. In our study, the SVM soft margin parameter was set to $C = 1$.

Each serum sample consists of 27000 m/z ratios. Since m/z values below 500 are likely to reflect the surface coatings and not serum proteins, we removed such m/z values from our samples before the beginning of work [12], [33]. In each sample, the number of peaks whose m/z values are above 500 is 22687. Next, after t-test for feature pruning, the number of peaks was reduced up to 641. Finally, by the binning task which choose the highest peak in each bin, 58 peaks were obtained. We performed 5-fold cross validation changing $\beta$ and chose $\beta = 1.0$ as a criterion for binning because the best performance was obtained when the $\beta$ value was used. Fig. 2.1 and Table 2.2 represent the change of the number of peaks.

Since we are interested in small peak subsets, we investigated the performance of three algorithms such as SVM-MB/RFE, SVM-RFE and Markov blanket with small peak subsets first using the top peak then the top two peaks and so forth up to the top 30 peaks according to the ranked features for each algorithm. This experiment was repeated 50 times, and the means and standard deviations of accuracy,
sensitivity and specificity were evaluated. Since the large size of Markov blanket may cause fragmentation of training samples and the results to degrade, we chose the small value as the size of Markov blanket, \( k = 1 \) and 2. Fig. 2.2 represents the results of experiments showing accuracy (a), sensitivity (b) and specificity (c) when \( k = 1 \), and accuracy (d), sensitivity (e) and specificity (f) when \( k = 2 \). Here, sensitivity is the percent of platinum-resistant samples that are correctly classified as the platinum-resistant. Specificity is the percent of platinum-sensitive samples that are correctly classified as the platinum-sensitive. As can be seen, SVM-MB/RFE outperformed other methods. Accuracy and sensitivity have the similar tendency as the number of peaks increases, while specificity has a valley shape. We investigated that how frequent the candidate peaks of 58 took part in forming 30 peaks subset during 50 runs. 46 peaks were used at least one time. 17 peaks were used in every iteration. Table 2.3 represents measurements comparison when 30 peaks subset was used. Fig. 2.3 shows the frequency along with the t-test value of each peak. Note that although no. 6 and 43 (765.260 and 5683.916 \( m/z \)) have a relatively high t-test value as 3.039 and 3.412, respectively, they participated just 3 and 9 times, respectively. On the other hand, no 13 (985.589 \( m/z \)) was used in the every run although the t-test value is as low as 1.926. And we observed that the accuracy when \( k = 2 \) is better than when \( k = 1 \).

2.4 Conclusion

We proposed a new supervised feature selection method (SVM-MB/RFE) to identify markers for detecting early relapse of ovarian cancer. In the preprocessing task of SVM-MB/RFE, the number of features was reduced up to 58 from 27000 of the raw data. By using a new score ranking combined from the expected cross-entropy value of Markov blanket and the weight value of SVM-RFE, SVM-MB/RFE
Table 2.3. Measurements comparison when $k = 1$ and $k = 2$ in SVM-MB/RFE. The number in parenthesis is the standard deviation.

<table>
<thead>
<tr>
<th>$k$</th>
<th>Measurement</th>
<th>SVM-MB/RFE</th>
<th>SVM-RFE</th>
<th>MB</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Accuracy (%)</td>
<td>83.7(6.0)</td>
<td>81.4(5.9)</td>
<td>79.0(5.7)</td>
</tr>
<tr>
<td>$k = 1$</td>
<td>Sensitivity (%)</td>
<td>71.6(10.8)</td>
<td>68.5(12.3)</td>
<td>62.3(10.5)</td>
</tr>
<tr>
<td></td>
<td>Specificity (%)</td>
<td>92.8(6.9)</td>
<td>91.1(7.4)</td>
<td>91.6(6.0)</td>
</tr>
<tr>
<td></td>
<td>Accuracy (%)</td>
<td>85.8(5.0)</td>
<td>82.6(5.3)</td>
<td>79.2(4.7)</td>
</tr>
<tr>
<td>$k = 2$</td>
<td>Sensitivity (%)</td>
<td>73.5(9.5)</td>
<td>69.3(10.2)</td>
<td>65.9(11.1)</td>
</tr>
<tr>
<td></td>
<td>Specificity (%)</td>
<td>95.1(5.0)</td>
<td>92.5(5.6)</td>
<td>89.2(7.0)</td>
</tr>
</tbody>
</table>

outperformed other methods. We demonstrated that although features have low t-test values, it is worth to see if they can be used as candidate features. In general, the small size of Markov blanket is used to avoid the fragmentation of training set. We compared the performance when $k = 1$ and 2, and showed that the accuracy when $k = 2$ is better than when $k = 1$ in SVM-MB/RFE. This project is in progress. Next, we will classify the full serum samples of ovarian cancer which includes a control dataset by implementing the multiple SVM-MB/RFE. The discovery of accurate biomarkers for identifying early recurrence of ovarian cancer will help ontologists give targeted therapy to ovarian cancer patients.
Figure 2.1. Example that shows the change of the number of peaks in one sample. (a) raw peaks (27000), (b) peaks after t-test (641), (c) peaks after binning (58).
Figure 2.2. Average measurements changing the size of feature subset (a) accuracy when $k = 1$, (b) sensitivity when $k = 1$, (c) specificity when $k = 1$, (d) accuracy when $k = 2$, (e) sensitivity when $k = 2$, (f) specificity when $k = 2$. 
Table 2.3. Frequency of the 58 candidate peaks and t-test values.

<table>
<thead>
<tr>
<th>No</th>
<th>m/z</th>
<th>frequency</th>
<th>t-test value</th>
<th>No</th>
<th>m/z</th>
<th>frequency</th>
<th>t-test value</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>538.115</td>
<td>42</td>
<td>2.657</td>
<td>24</td>
<td>1439.987</td>
<td>9</td>
<td>1.967</td>
</tr>
<tr>
<td>2</td>
<td>599.249</td>
<td>19</td>
<td>2.552</td>
<td>25</td>
<td>1467.177</td>
<td>6</td>
<td>2.196</td>
</tr>
<tr>
<td>3</td>
<td>641.756</td>
<td>18</td>
<td>2.202</td>
<td>26</td>
<td>1495.840</td>
<td>50</td>
<td>2.652</td>
</tr>
<tr>
<td>4</td>
<td>738.831</td>
<td>50</td>
<td>2.334</td>
<td>27</td>
<td>1613.690</td>
<td>3</td>
<td>2.195</td>
</tr>
<tr>
<td>5</td>
<td>756.015</td>
<td>50</td>
<td>2.347</td>
<td>28</td>
<td>1668.505</td>
<td>16</td>
<td>2.331</td>
</tr>
<tr>
<td>6</td>
<td>765.260</td>
<td>3</td>
<td>3.039</td>
<td>29</td>
<td>1732.524</td>
<td>50</td>
<td>2.913</td>
</tr>
<tr>
<td>7</td>
<td>773.979</td>
<td>1</td>
<td>2.275</td>
<td>30</td>
<td>1862.826</td>
<td>49</td>
<td>2.315</td>
</tr>
<tr>
<td>8</td>
<td>786.563</td>
<td>34</td>
<td>2.378</td>
<td>31</td>
<td>1918.018</td>
<td>1</td>
<td>2.645</td>
</tr>
<tr>
<td>9</td>
<td>815.623</td>
<td>1</td>
<td>2.053</td>
<td>32</td>
<td>1950.779</td>
<td>39</td>
<td>2.460</td>
</tr>
<tr>
<td>10</td>
<td>862.975</td>
<td>4</td>
<td>2.147</td>
<td>33</td>
<td>2064.532</td>
<td>41</td>
<td>1.986</td>
</tr>
<tr>
<td>11</td>
<td>909.771</td>
<td>31</td>
<td>2.317</td>
<td>34</td>
<td>2101.403</td>
<td>50</td>
<td>2.578</td>
</tr>
<tr>
<td>12</td>
<td>929.156</td>
<td>8</td>
<td>2.051</td>
<td>35</td>
<td>2198.208</td>
<td>31</td>
<td>2.522</td>
</tr>
<tr>
<td>13</td>
<td>985.579</td>
<td>50</td>
<td>1.962</td>
<td>36</td>
<td>2351.324</td>
<td>35</td>
<td>2.252</td>
</tr>
<tr>
<td>14</td>
<td>1023.450</td>
<td>38</td>
<td>2.417</td>
<td>37</td>
<td>2421.550</td>
<td>50</td>
<td>2.773</td>
</tr>
<tr>
<td>15</td>
<td>1040.963</td>
<td>50</td>
<td>2.068</td>
<td>38</td>
<td>2445.361</td>
<td>50</td>
<td>3.392</td>
</tr>
<tr>
<td>16</td>
<td>1063.064</td>
<td>37</td>
<td>2.057</td>
<td>39</td>
<td>2490.714</td>
<td>32</td>
<td>2.130</td>
</tr>
<tr>
<td>17</td>
<td>1132.880</td>
<td>50</td>
<td>3.029</td>
<td>40</td>
<td>3753.111</td>
<td>45</td>
<td>2.206</td>
</tr>
<tr>
<td>18</td>
<td>1194.392</td>
<td>50</td>
<td>2.342</td>
<td>41</td>
<td>4923.713</td>
<td>44</td>
<td>2.944</td>
</tr>
<tr>
<td>19</td>
<td>1201.287</td>
<td>50</td>
<td>2.209</td>
<td>42</td>
<td>4981.335</td>
<td>50</td>
<td>2.504</td>
</tr>
<tr>
<td>20</td>
<td>1225.759</td>
<td>50</td>
<td>2.050</td>
<td>43</td>
<td>5683.916</td>
<td>9</td>
<td>3.412</td>
</tr>
<tr>
<td>21</td>
<td>1260.511</td>
<td>1</td>
<td>2.045</td>
<td>44</td>
<td>12863.914</td>
<td>49</td>
<td>2.160</td>
</tr>
<tr>
<td>22</td>
<td>1370.018</td>
<td>4</td>
<td>2.175</td>
<td>45</td>
<td>19196.391</td>
<td>50</td>
<td>3.319</td>
</tr>
<tr>
<td>23</td>
<td>1419.366</td>
<td>50</td>
<td>2.370</td>
<td>46</td>
<td>19976.675</td>
<td>50</td>
<td>2.979</td>
</tr>
</tbody>
</table>

Figure 2.3. Frequency of the 58 candidate peaks and t-test values. (a) table that shows how often the 58 candidate peaks take part in forming 30 peaks subset along with their t-test values, (b) comparison graph with regard to the frequency and t-test value.
CHAPTER 3

MULTI-STAGE DISEASE CLASSIFICATION BASED ON
BI-CLASSIFICATION STRATEGY

3.1 Introduction

For high-resolution mass spectrometry (MS) data, traditional machine learning algorithms may break down with high-dimensional input (mass-to-charge ratios, features, biomarkers). Very limited research has been carried out to explore the computational challenges. Yu et al. developed two approaches using different preprocessing and classification methods to analyze high-resolution SELDI-TOF (surface-enhanced laser desorption ionization time-of-flight) MS ovarian data [34], [35]. In the first study, a feature dimension reduction method consisting of a four-step preprocessing was proposed followed by a standard Support Vector Machine (SVM) for classification [34]. In the second paper [35], Bayesian neural network models were used for sample classification. Sequential feature selection was carried out by using bootstrap technique based on the two-sample Kolmogrov-Smirnov test (KS-test). Ressom et al. have designed a computational method that combines particle swarm optimization with support vector machine to distinguish liver cancer patients from healthy individuals in SELDI-QqTOF spectra [16]. Recently, this group proposed to combine ant colony optimization (ACO) with SVM to distinguish hepatocellular carcinoma patients from cirrhosis patients via MALDI-TOF mass spectra [36]. Particle swarm optimization and ant colony optimization are interesting swarm intelligence techniques that have been successfully applied to a number of optimization problems. A comparative study of several well-known classification algorithms, such as
linear discriminant analysis (LDA), $k$-nearest neighbor (KNN), random forest (RF), bagging, boosting, and SVM, has been carried out for ovarian cancer diagnosis [17]. It was demonstrated that RF approach leads to an overall higher accuracy rate as well as a more stable assessment in terms of classification errors. As a summary for current literature on high-resolution MS data analysis, the work, feature selection and classification, has been focused on binary class samples. However, medical data sets typically contain multiple classes. Therefore, feature selection as well as sample prediction for multi-categories are necessary.

Multi-category classification involves assigning an unknown sample into one of the $k$ classes ($k > 2$) [37], [38]. There are two main strategies to tackle the multi-class problems [39]. The first method considers all classes at once by constructing a decision function, which may require high cost and complexity. In the second method, the multi-class problems are broken down into a set of binary classification problems, which is more computationally tractable [40]. There exist several methods to reduce multi-class problems to binary class problems. These include one-against-the-rest, one-against-one, and error-correcting output coding (ECOC). ECOC has shown the generalization capability in multi-sample classification.

ECOC was inspired from communication theory, where misclassifications wrongly guessed by classifiers can be corrected. In the ECOC multi-class classification problem, there are several factors to affect the performance of the algorithm. Investigations have been carried out on the selection of coding matrix [41], [42], [43], [39] and optimization of the decoding function [44], [45], [46]. In the study of coding matrix design, Dietterich and Bakiri provided methods for constructing error-correcting codes in which the coding method varies depending on the number of classes included in the problem [41]. Also the robustness of ECOC in handling small sample size and with respect to the assignment of codewords was also investigated in this work.
Crammer and Singer investigated problem-dependent discrete and continuous codes where they proved that finding an optimal discrete matrix is unfeasible and proposed a relaxation of output codes by using continuous values [42]. Pujol et al. introduced a heuristic coding method based on the mutual information between classes in each column to achieve maximum discrimination [43]. Ie et al. proposed a multi-category classification method based on ECOC for assigning a sequence of amino acids to one of the known protein structures [39]. The coding matrix is directly related to the structural hierarchy such as fold and superfamily detectors. Decoding rule plays an important role in ECOC classification [44], [45], [46]. Smith and Windeatt reviewed existing decoding rules which are based on distance, probability estimation and pattern space transformation, and proposed a likelihood decoding rule [44]. Passerini et al. introduced a new decoding function based on class conditional probabilities that represent the closeness between codewords and the vector formed by the outputs of the classifiers [45]. Kuncheva and Whitaker proposed to use diversity measures rather than the standard minimum Hamming distance to evaluate the quality of an error-correcting code and suggested an evolutionary algorithm to construct the code [46]. Methods combining boosting and ECOC have been studied to take the performance advantages of boosting [47], [48].

In this paper, we develop an ensemble based multi-class learning algorithm by integrating a new ECOC scheme and one-against-one pairwise coupling (PWC) scheme. The motivation is to take advantage of each multi-class classification strategy to achieve the most reliable sample prediction. Our contributions come from defining a performance-based weighting function for binary classifiers (dichotomies) in ECOC and a robust decoding function incorporating individual sample property and the overall dichotomy performance. A unique set of biomarkers to distinguishing each pair of categories (classes) is discovered by a new feature selection method,
Extended Markov Blanket. Fig. 3.1 shows the framework of the proposed biomarker selection and sample category prediction.

The MALDI-TOF data set used in this study is opened by Ressom et al. It consists of hepatocellular carcinoma (HCC) patients, cirrhosis patients, and healthy individuals [36]. Among the three-class data set, Ressom et al. used hepatocellular carcinoma (HCC) and cirrhosis spectra for two category sample classification leaving healthy spectra for peak screening and outlier detection. In this study, however, we used all three-class spectra for multi-category sample classification and biomarker selection for each class.

3.2 Methods

3.2.1 Data Preprocessing

The binned spectra of MALDI-TOF MS for liver cancer study were binned with the size of 100 ppm (parts-per-million) and in total yielding 23,846 bins. Preprocessing by baseline correction and normalization was applied to the binned spectra.
prior to the next stage pattern finding. For baseline correction, we follow the same method, spline approximation, as what was done by Ressom et al. [36]. After that, each spectrum is normalized by dividing the baseline corrected spectrum by its total ion current (the summed intensity over all \( m/z \) values in the baseline corrected spectrum). Because of the small normalized intensity value, all the intensities are multiplied by 10,000 for computational convenience. Ressom et al. generated windows by combining bins after preprocessing. In this study, however, we used the binned data itself to find biomarker candidates in narrow mass regions. To reduce computational burden caused by using peaks in all the bins, we rank peaks by a feature ranking method based on the ratio of between-group to within-group peak differences and select the tractable size of features in our algorithm. The feature ranking method was proposed by Dudoit et al. for feature selection in multi-class problems [49]. For certain \( m/z \) peak \( j \), the ratio is:

\[
\text{BW}(j) = \frac{\sum_i \sum_k I(y_i = k)(\overline{x}_{kj} - \overline{x}_{j})^2}{\sum_i \sum_k I(y_i = k)(x_{ij} - \overline{x}_{kj})^2},
\]  

(3.1)

where \( I(\cdot) \) is the indicator function, \( \overline{x}_{kj} \) denotes the average intensity of peak \( j \) across samples belonging to class \( k \), \( \overline{x}_{j} \) is the average intensity of peak \( j \) across all samples, \( x_{ij} \) is the intensity of sample \( i \) at peak \( j \), and \( y_i \) is the class label for sample \( i \). The larger the ratio, the more likely \( m/z \) peak \( j \) will be relevant to the class separation.

In the next section, we will describe the basic algorithms used in our biomarker selection and classifier design.

3.2.2 A Redesigned ECOC Scheme for Multi-Class Classification

ECOC is a classification method that breaks a \( k \)-class prediction problem into several binary classifications. In the ECOC framework, each class is assigned a unique codeword of length \( h \) composed of 1 and -1 (Fig. 3.1), forming a \( k \times h \) coding matrix.
Typically, the rows of the matrix are codewords assigned to the corresponding class $c_i$ and the columns $D_j$ represent binary classifiers which partition the samples into two subsets labeled according to the coding matrix. Based on the class partitions, the matrix produces $h$ binary classifiers called dichotomies. For a test sample, a code of length $h$ is obtained as a result of the outputs of the $h$ binary functions. This code is compared with each of the $k$ codewords defined in the coding matrix, and the sample is assigned to a class with the closest codeword using certain distance measure. In doing so, the classification process can be seen as a decoding operation.

Good error-correcting codes should meet two main criteria: row and column separations. Row separation represents that each codeword should be well separated from the others. Column separation indicates that the columns should be uncorrelated with each other and is achieved by maximizing the distance between one column and each of the other columns. The minimum Hamming distance is used to measure the quality of an error-correcting code between any pair of codewords.

In the generic ECOC multi-class classification problem, there are several factors affecting the performance of the algorithm. As previously introduced, investigations have been carried out on the designing of coding matrix and option of the decoding function. We develop a new ECOC framework from three aspects: a weighting strategy for different dichotomies, feature dimensionality reduction, and performance-based decoding function.

### 3.2.2.1 Weighting strategy

In standard ECOC framework, the influence of all the dichotomies during classification of an unknown sample is equally treated. However, the importance of each dichotomy is different in terms of generating decision boundaries for the training samples. Therefore, we define a weighting function which is similar to the one used
in boosting algorithms as shown in Eq. (3.12). The weight value of each dichotomy is computed by using the error rate estimated for the dichotomy with the validation data set. Therefore, the weight value represents how confident the dichotomy is. In Eq. (3.12), \( v_i \) and \( e_i \) are the weight value and the error rate of the \( i \)-th dichotomy. In the case where the accuracy of the dichotomy is greater than 50\%, the weight value becomes positive; otherwise, a negative value is returned,

\[
v_i = 0.5 \log \left( \frac{1 - e_i}{e_i} \right).
\]  

(3.2)

Throughout the study, we choose 10-fold cross validation (CV) where all samples are randomly split into 10 exclusive folds. Without loss of generality, SVM is used as the dichotomy function. For each of the ten experiments (iterations), typically nine folds are used as train data, and the remaining one fold is applied as test data. In our implementation, to estimate the accuracy of each dichotomy function, we further divide the nine fold train data into 10 folds, i.e., sub-10-fold. In each iteration of the sub-10-fold, nine folds become sub-train and one fold for sub-test. An averaged error rate resulting from sub-test data is put into Eq. (3.12) to obtain the weight value. This estimation is separately performed for each dichotomy.

3.2.2.2 Feature reduction

In the training of dichotomies, due to the high dimensionality of mass profile even after the preprocessing, irrelevant features might still exist. Therefore, using all features will degrade the ability of dichotomies and increase computational cost. Here, we employ a feature reduction algorithm based on information gain to remove irrelevant features in each dichotomy.
Let $S$ be the set of instances from $k$ classes, $c_1, c_2, \ldots, c_k$ and $P(c_i, S)$ be the fraction of the instances in $S$ that belong to $c_i$. The entropy of the class distribution in $S$ is as follows:

$$I(S) = -\sum_{i=1}^{k} P(c_i, S) \log P(c_i, S).$$

(3.3)

Suppose feature $F_i$ has $m$ distinct values, $f_i^1, f_i^2, \ldots, f_i^m$. Let $S_j$ be the set of instances whose value on attribute $F_i$ is $f_i^j$. Then, the information gain of instance set $S$ based on attribute $F_i$ is calculated as

$$Gain(F_i) = I(S) - I(S|F_i),$$

(3.4)

$$= I(S) - \sum_{j=1}^{m} P(F_i = f_i^j) I(S_j),$$

(3.5)

$$= I(S) - \sum_{j=1}^{m} \frac{|S_j|}{|S|} \times I(S_j).$$

(3.6)

The information gain reflects the reduction in uncertainty about the overall class entropy when certain feature $F_i$ is given. In other words, features with zero information gain indicate the inability to reduce such uncertainty and should be removed during the training of dichotomy function.

### 3.2.2.3 Decoding function

Based on the binary class predictions from the ensemble dichotomies, an output code is generated for the test sample. To assign the final class label to the test sample, a decoding function is required. The decoding function measures the closeness between the output code and the codewords in the coding matrix $M$. A new decoding function reflecting the importance of individual dichotomies is defined as:

$$d_j = \sum_{i=1}^{h} \exp(-v_i x_i y_i^j),$$

(3.7)

where $d_j$ is the distance between the test sample and the $j$-th class, $v_i$ is the importance of the $i$-th dichotomy, $x_i$ is the $i$-th bit of the output code for the test sample.
and $y^j_i$ is the $i$-th bit of the codeword for the $j$-th class in the $M$ matrix. A test sample is assigned to the class which has the minimum distance,

$$c_e = \text{argmin}_{1 \leq j \leq k} d_j.$$  

(3.8)

3.2.3 Pairwise Coupling (PWC) Scheme

Though ECOC multi-class sample prediction performs well in most cases, the major limitation of this framework is that it is hard to be used for subspace feature selection. As a result, biomarkers contributing to certain phenotype category can not be estimated within this framework. Therefore, we incorporate another binary classification scheme, pairwise one-against-one classifier, which will provide us the capability to find discriminant biomarkers to distinguish phenotype differences between one class and another.

Here we are not applying one-against-the-rest binary classifiers. This selection comes from the legitimate biomedical interpretation. As an example, if choosing one-against-the-rest, we need to find biomakers to distinguish cirrhosis patients from the rest which is the combined samples of normal and hepatocellular carcinoma (HCC). Lumping the different stage samples like HCC and normal which may be very different from each other at molecular level could obscure biomarkers that truly distinguish patients from cirrhosis. On the other hand, one-against-one comparisons may point out differences between classes of patients that should not be lumped together, such as normal and HCC. Furthermore, with the number of class $k$ in biomedicine usually less than 10, the computation issue raised by one-against-one binary classification will not be a concern.

The number of all possible binary classifiers using one-against-one classification is $k(k - 1)/2$. A common way to determine a final class for the test sample is
voting. However in many cases, it is essential that multi-class classification should be a confidence measure, such as posterior probability [50]. In this study, we use a probability measure based on pairwise coupling (PWC) scheme to leverage binary prediction results. The probability for a test sample belonging to class $i$ is calculated from combining $k(k - 1)/2$ two-class probabilities [51]:

$$p_i = \frac{1}{\sum_{j=1, j \neq i}^{k} \frac{1}{\mu_{ij}} - (k - 2)},$$

(3.9)

where $\mu_{ij} = P(y = i | y = i \text{ or } j, x)$ corresponding to the posterior probability from the binary classifier.

Since standard SVM does not provide a way to measure the posterior probabilities, this is obtained using a parametric sigmoid model proposed by Platt [52]. For a SVM binary classification, the posterior probability is obtained as:

$$P(y = 1 | x) = \frac{1}{1 + \exp(A \times f(x) + B)},$$

(3.10)

where $y = 1$ is the class label for binary class sample $x$. The parameters $A$ and $B$ are determined by the maximum likelihood estimation from the training set.

A test sample in PWC scheme is assigned to a class which has the maximum probability,

$$c_p = \arg\max_{1 \leq i \leq k} p_i.$$  

(3.11)

### 3.2.4 Extended Markov Blanket Feature Selection

Here, we describe the extended Markov blanket algorithm mentioned in 1.5.1 again. To discover the discriminant biomarkers among multi-classes, we propose a new wrapper-based feature selection algorithm called extended Markov blanket (EMB). This feature selection process is embedded in the pairwise coupling (PWC) multi-class classification scheme. The original Markov blanket is a filter-based feature selection method. Our algorithm considers reducing redundant features while
selecting the most discriminant ones. With the feature subset selected by extended Markov blanket, we run a linear SVM to calculate probabilities for the pair of classes.

To be more specific, for feature $F_i$ remained after preprocessing, two feature subsets are considered: the high correlated feature subset (HCFS) and the low correlated feature subset (LCFS) composed of the least correlated $d$ features for $F_i$. HCFS feature subset is used to remove redundant features as in the classical Markov blanket feature selection. Our contribution comes from utilizing LCFS to estimate the classification capability of each feature during the Markov blanket process. This is derived from the fact that mutually low correlated features, in general, lead to good classification performance.

We will now describe how the extended Markov blanket algorithm works out for feature selection. 10-fold cross validation data split is the same as previously introduced in section 3.2.2.1. For each feature $F_i$, we perform Markov blanket algorithm with its HCFS obtained from the train data to compute the expected cross-entropy value, $\Delta(F_i|M_i)$. On the other hand, using LCFS and feature $F_i$, we run the linear SVM algorithm where the sub-train and sub-test data are used for training and testing, respectively, to obtain a roughly estimated accuracy, denoted as $\beta$. Then, we compute the normalized weight for each feature in the LCFS using the following function:

$$W_k = \begin{cases} 
\frac{|w_k|}{\sum_{j=1}^{d+1} |w_j|} \times \beta \times \delta, & \text{for } \gamma \leq \beta, \\
\left(1 - \frac{|w_k|}{\sum_{j=1}^{d+1} |w_j|}\right) \times (\gamma - \beta) \times \delta, & \text{for } \gamma > \beta,
\end{cases}$$

(3.12)

where

$$\delta = \begin{cases} 
1, & \text{for } \gamma \leq \beta \\
-1, & \text{for } \gamma > \beta
\end{cases}$$

(3.13)

and $\beta$ is the accuracy from sub-test samples in the linear SVM, $k$ is the index for features in LCFS including feature $F_i$, $|w_k|$ is the absolute SVM weight obtained
using Eq. (1.7), $d$ is the size of LCFS chosen as 10, 20, or 30, and $\gamma$ is a heuristic performance threshold typically specified as 0.5. After computing $|w_k|$ for all features in the LCFS, each $|w_k|$ is normalized by the summed absolute SVM weights of all the features in the LCFS and that of feature $F_i$.

In contrast to SVM-RFE where features with the smallest absolute weights are removed, we assume that features with large absolute SVM weights are important in terms of classification power. Not surprisingly to observe, although the normalized SVM weight of certain feature in Eq. (1.7) may be high in an LCFS, the classification accuracy of the linear SVM performed with the limited LCFS may be low due to the partial discriminant capacity of one LCFS. Therefore, not only the normalized SVM weight for each feature but also the accuracy with its LCFS are important factors to estimate feature weight. This leads to the multiplication of the normalized SVM weight by the accuracy obtained after the SVM. Through a few experiments, we found that $\gamma = 0.5$ assures a good performance. If the accuracy is greater than 50%, the proposed weight becomes positive by multiplying 1 as a value of $\delta$. Otherwise, we treat it as a penalty using $-1$ as $\delta$ value.

After computing the proposed weights and expected cross-entropy $\Delta(F_i|M_i)$ values for all features as introduced above, certain feature with the smallest $\Delta(F_i|M_i)$ value is removed according to Markov blanket rule. Then, the HCFS and LCFS for all but the removed feature are rebuilt. In fact, only those subsets that contain the removed feature will be affected and modified. Again, we compute the $\Delta(F_i|M_i)$ values and the proposed weights for survived features. The feature weights will be accumulated during the Markov blanket feature removal process. The whole procedure keeps going until the predefined feature number is reached.
Figure 3.2. Flowchart of the algorithm ($p = 20$ in this study).

3.2.5 Feature Pruning by Backward Feature Selection

After each iteration of 10-fold CV (20 times of 10-fold CV were applied, totalled 200 iterations), a mean weight value for each feature is calculated using the accumulated weight divided by the occurrence counting in all LCFS. All the features are then ranked according to the mean weights. Then, a backward feature elimination method with the top 60 features is performed by using a linear SVM to find a compact feature subset during each 10-fold CV iteration. The backward feature selection approach removes one feature at a time from the current feature set. Each
time a feature without which the accuracy is improved will be excluded. This process continues until one feature remains. Features with which the best accuracy is obtained form an optimal feature subset for the pair of classes. With the optimal feature subset, one-against-one binary classifier for certain pair of classes is built. This method is performed for exhaustive pairs of classes.

After finishing all the iterations of 10-fold CV training, we count how many times a single feature is included in individual optimal feature subsets for each pair of classes. A final feature set is sorted according to the frequency. The higher the frequency, the more reliable a feature is. Fig. 3.2 shows the overall operation flow of the proposed framework.

3.2.6 Retraining

In section 3.2.4 and 3.2.5, a method to find important features for each pair of classes was presented. The final sample prediction will be determined by outputs of ECOC and PWC schemes. If two resultant class labels are the same \((c_e = c_p)\), the test sample is assigned to the identical class; otherwise \((c_e \neq c_p)\), a retraining will be performed using samples that only belong to classes \(c_e\) and \(c_p\) excluding other class samples. Therefore, the retraining comes to be a binary classification problem. In retraining, we use the feature subset which is found for the two classes \(c_e\) and \(c_p\) by Extended Markov Blanket. As a consequence of the binary classification, the final class is determined for the test sample.

3.3 Experiments

Liver cancer data opened by Ressom et al. consists of 201 spectra, hepatocellular carcinoma (78), cirrhosis (51) and health (72). In this paper, all the samples were used to find biomarkers in such a multi-stage liver cancer. We implemented the
Table 3.1. The means and standard deviations (in parenthesis) of accuracies in liver cancer data set.

<table>
<thead>
<tr>
<th>Methods</th>
<th>Accuracy \ No. of peaks</th>
<th>100</th>
<th>200</th>
<th>300</th>
<th>400</th>
<th>500</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>Proposed Method</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Overall</td>
<td>81.94(1.54)</td>
<td>86.92(1.79)</td>
<td><strong>88.71(1.85)</strong></td>
<td>87.61(1.78)</td>
<td>84.93(1.95)</td>
<td></td>
</tr>
<tr>
<td>Cirrhosis</td>
<td>80.39(2.07)</td>
<td>80.98(3.82)</td>
<td>86.86(1.86)</td>
<td>87.25(3.10)</td>
<td>86.67(2.58)</td>
<td></td>
</tr>
<tr>
<td>HCC</td>
<td>86.54(1.93)</td>
<td>90.26(2.28)</td>
<td>89.62(2.13)</td>
<td>88.08(2.27)</td>
<td>85.13(2.28)</td>
<td></td>
</tr>
<tr>
<td>Health</td>
<td>78.06(2.91)</td>
<td>87.50(2.78)</td>
<td>89.03(2.89)</td>
<td>87.36(3.17)</td>
<td>83.47(3.43)</td>
<td></td>
</tr>
<tr>
<td>Overall</td>
<td>78.75(2.25)</td>
<td>81.34(1.74)</td>
<td>79.90(2.07)</td>
<td>79.45(2.36)</td>
<td>77.16(1.34)</td>
<td></td>
</tr>
<tr>
<td>Cirrhosis</td>
<td>79.22(3.36)</td>
<td>78.24(4.66)</td>
<td>79.61(5.16)</td>
<td>79.22(2.80)</td>
<td>74.51(3.46)</td>
<td></td>
</tr>
<tr>
<td>HCC</td>
<td>83.97(2.58)</td>
<td>83.46(2.05)</td>
<td>81.67(4.10)</td>
<td>82.44(3.63)</td>
<td>81.03(2.08)</td>
<td></td>
</tr>
<tr>
<td>Health</td>
<td>72.78(5.08)</td>
<td>81.25(3.54)</td>
<td>78.19(3.76)</td>
<td>76.39(4.19)</td>
<td>74.66(2.66)</td>
<td></td>
</tr>
<tr>
<td><strong>ECOC</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Overall</td>
<td>78.86(1.63)</td>
<td>81.29(1.22)</td>
<td>80.45(1.83)</td>
<td>83.28(1.37)</td>
<td>82.89(1.56)</td>
<td></td>
</tr>
<tr>
<td>Cirrhosis</td>
<td>71.37(4.15)</td>
<td>76.08(4.21)</td>
<td>74.71(4.18)</td>
<td>76.27(4.28)</td>
<td>78.04(3.04)</td>
<td></td>
</tr>
<tr>
<td>HCC</td>
<td>83.72(3.92)</td>
<td>85.64(2.25)</td>
<td>85.38(2.20)</td>
<td>87.82(1.84)</td>
<td>88.08(1.71)</td>
<td></td>
</tr>
<tr>
<td>Health</td>
<td>78.89(2.25)</td>
<td>80.28(2.68)</td>
<td>79.17(2.85)</td>
<td>83.33(2.85)</td>
<td>80.69(3.30)</td>
<td></td>
</tr>
<tr>
<td>Overall</td>
<td>79.01(0.51)</td>
<td>81.94(0.41)</td>
<td>82.69(0.51)</td>
<td>83.73(0.62)</td>
<td>84.03(0.64)</td>
<td></td>
</tr>
<tr>
<td>Cirrhosis</td>
<td>80.59(1.72)</td>
<td>82.75(1.80)</td>
<td>83.53(1.01)</td>
<td>83.14(1.01)</td>
<td>82.94(0.95)</td>
<td></td>
</tr>
<tr>
<td>HCC</td>
<td>88.33(0.41)</td>
<td>89.87(0.73)</td>
<td>87.31(0.73)</td>
<td>87.31(0.41)</td>
<td>87.31(0.41)</td>
<td></td>
</tr>
<tr>
<td>Health</td>
<td>67.78(1.10)</td>
<td>72.78(4.19)</td>
<td>77.08(1.50)</td>
<td>80.28(1.94)</td>
<td>81.25(2.10)</td>
<td></td>
</tr>
<tr>
<td><strong>Random Forest</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Overall</td>
<td>71.09(3.19)</td>
<td>71.74(3.38)</td>
<td>73.28(2.86)</td>
<td>73.23(2.95)</td>
<td>73.08(3.50)</td>
<td></td>
</tr>
<tr>
<td>Cirrhosis</td>
<td>65.49(5.93)</td>
<td>65.29(3.70)</td>
<td>66.08(5.47)</td>
<td>64.90(4.75)</td>
<td>65.88(4.99)</td>
<td></td>
</tr>
<tr>
<td>HCC</td>
<td>76.54(5.54)</td>
<td>77.69(3.69)</td>
<td>78.85(2.03)</td>
<td>78.33(3.80)</td>
<td>76.54(4.15)</td>
<td></td>
</tr>
<tr>
<td>Health</td>
<td>69.17(3.63)</td>
<td>69.86(6.07)</td>
<td>72.36(4.56)</td>
<td>73.61(3.82)</td>
<td>74.44(4.64)</td>
<td></td>
</tr>
</tbody>
</table>

The proposed algorithm based on LIBSVM [29] and WEKA library [30]. A linear SVM was adopted in retraining and in both ECOC scheme and PWC scheme. For ECOC scheme, a random coding strategy was used in which values of \{+1, -1\} were selected uniformly at random to generate codes.

According to the BW ratios after preprocessing, we ranked the 23,846 binned peaks (features) and performed experiments with the top 100, then the top 200 and so forth up to the top 500 peaks. The performance of our method was compared with other classification algorithms such as standard ECOC, Random Forest, Naive Bayes, and J48. In all experiments, 10-fold cross validation was applied.

Table 3.1 shows the experimental results in terms of accuracy means and standard deviations. The individual accuracy for each class is the ratio of correctly labelled samples over the real ones while the overall accuracy is with respect to the total correctly labelled samples for all the classes. The proposed method shows the
best accuracy of 88.71% when the experiments were done with the top 300 peaks ranked by BW ratios. The corresponding accuracies for cirrhosis, HCC, and health are 86.86%, 89.62%, and 89.03%, respectively. As a comparison, J48 achieved the least satisfactory performance in all the trials. Fig. 3.3 visually represents the overall classification accuracies from different classifiers.

In PWC scheme, for each pair of classes (cirrhosis vs health, cirrhosis vs HCC, and health vs HCC) we count how many times each peak is included in the optimal feature subset after the backward feature elimination (See sections 3.2.4 and 3.2.5). Peaks with high frequencies are more reliable than randomly observed biomarkers. In the situation of best overall accuracy when 300 peaks were chosen, the frequencies of all optimal feature sets after backward feature elimination were counted. Fig. 3.5 drafts the normalized frequency of the top 60 peaks selected by our method in cirrho-
sis vs health, cirrhosis vs HCC, and health vs HCC experiments where the highest frequency was scaled to 100. Among the 60 peaks, the top 20 peaks, sorted by the frequency, were listed in Fig. 3.4(a). Furthermore, we grouped the 20 peaks within 3 Da as shown in Fig. 3.4(b). Note that some peaks were commonly selected: 1796.8241, 1961.7096, 1961.9057 in cirrhosis vs health and cirrhosis vs HCC; 2373.3603 and 2373.5976 in cirrhosis vs HCC and health vs HCC; and 2389.7923 in cirrhosis vs health and health vs HCC. This further proves that not a single biomarker but a group of them contributes to the expressions of different phenotypes. We compared peaks selected by our method with those found in cirrhosis vs HCC experiments by Ressom et al.. We observed that 1865.4808 and 1797.0038 \textit{m/z} corresponding to rank 1 and 2 by our algorithm belong to \textit{m/z} windows 1864.0-1870.2 and 1793.1-

### Table: Peaks Ranked by Frequency

<table>
<thead>
<tr>
<th>Ranking</th>
<th>Cirrhosis vs Health</th>
<th>Cirrhosis vs HCC</th>
<th>Health vs HCC</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>1326.4894</td>
<td>1865.4808</td>
<td>3201.7317</td>
</tr>
<tr>
<td>2</td>
<td>1961.9057</td>
<td>1797.0038</td>
<td>3202.0519</td>
</tr>
<tr>
<td>3</td>
<td>2307.1390</td>
<td>2390.0313</td>
<td>2373.1230</td>
</tr>
<tr>
<td>4</td>
<td>1961.7096</td>
<td>1865.6674</td>
<td>2389.7923</td>
</tr>
<tr>
<td>5</td>
<td>2372.1740</td>
<td>2365.5415</td>
<td>2535.5320</td>
</tr>
<tr>
<td>6</td>
<td>1327.1527</td>
<td>2391.4656</td>
<td>2536.0392</td>
</tr>
<tr>
<td>7</td>
<td>2362.9410</td>
<td>1961.7096</td>
<td>2373.3603</td>
</tr>
<tr>
<td>8</td>
<td>2389.5533</td>
<td>2365.3050</td>
<td>3202.3721</td>
</tr>
<tr>
<td>9</td>
<td>1327.2855</td>
<td>2390.9874</td>
<td>2535.0250</td>
</tr>
<tr>
<td>10</td>
<td>1325.5612</td>
<td>1710.3930</td>
<td>2372.1740</td>
</tr>
<tr>
<td>11</td>
<td>1325.6938</td>
<td>1797.0038</td>
<td>2372.4112</td>
</tr>
<tr>
<td>12</td>
<td>1326.4894</td>
<td>1797.0038</td>
<td>2372.4112</td>
</tr>
<tr>
<td>13</td>
<td>1796.8241</td>
<td>2365.0685</td>
<td>2534.0112</td>
</tr>
<tr>
<td>14</td>
<td>2373.3603</td>
<td>2365.3050</td>
<td>2535.0250</td>
</tr>
<tr>
<td>15</td>
<td>2373.5976</td>
<td>2365.5415</td>
<td>2535.2785</td>
</tr>
<tr>
<td>16</td>
<td>1961.7096</td>
<td>2365.7780</td>
<td>2535.5320</td>
</tr>
<tr>
<td>17</td>
<td>1961.9057</td>
<td>2373.3603</td>
<td>2535.7856</td>
</tr>
<tr>
<td>18</td>
<td>1796.6444</td>
<td>2366.6726</td>
<td>2536.0392</td>
</tr>
<tr>
<td>19</td>
<td>1796.8241</td>
<td>2372.4112</td>
<td>2536.2928</td>
</tr>
<tr>
<td>20</td>
<td>1796.8241</td>
<td>2390.0313</td>
<td>2536.2928</td>
</tr>
</tbody>
</table>

### Figure 3.4

(a) The \textit{m/z} values for the top 20 observed peaks out of 300 peaks obtained by BW ratios. (b) Grouping of peaks within 0.3 Da from (a).
Figure 3.5. Frequency for the optimal feature sets from the top 300 peaks based on BW ratios. (a) Cirrhosis vs Health, (b) Cirrhosis vs HCC, (c) Health vs HCC.
1797.0, respectively, which were selected by ACO-SVM. In particular, 1865.4808 \( m/z \) is top-ranked in our algorithm as well as in both methods of weighting factor and ACO-SVM by Ressom et al.

Fig. 3.6 represents the average intensity for peaks in Fig. 3.4(a). We should note that in each pair of classes more severe stage of the disease shows higher intensity distribution. This may imply more protein secretion relevant to the disease. In particular, there are a few peaks which have significant intensity difference in cirrhosis vs HCC and health vs HCC: peaks ranking 2, 9, 15, and 20 corresponding to 1797.0038, 2390.9874, 2373.5976, and 1961.9057 in cirrhosis vs HCC; peaks ranking 6, 9, 10, 13, and 19 corresponding to 2536.0392, 2535.0250, 2535.7856, 2535.2785, and 2373.5976 in health vs HCC. Note that three peaks 2535.0250, 2535.7856, and 2535.2785 among five high intensity peaks in health vs HCC were grouped as seen in Fig. 3.4(b). Also, peak 2373.5976 was commonly found in cirrhosis vs HCC (ranking 15) and health vs HCC (ranking 19) experiments, with much higher average intensity in HCC. In cirrhosis vs HCC, however, the top-ranked peak 1865.4808 not only in our algorithm but in both methods by Ressom et al. does not show a considerable difference of intensity. This further evinces that the importance of possible biomarkers is not only purely determined by its absolute volume but also by its correlation or coregulation with other biomarkers.

For next stage of biomarker identification, finding exact \( m/z \) values is critical to sequencing candidate selection, as small mass difference may lead to different protein candidates. Since our biomaker candidates fall into single bins whose size is usually less than 1 Dalton as opposed to a large window size spanning several daltons, the proposed approach offers narrow range to choose the sequencing candidates.
Figure 3.6. Average intensities for the top 20 peaks observed by the proposed method (corresponding to peaks listed in Fig. 3.4). (a) Cirrhosis vs Health, (b) Cirrhosis vs HCC, (c) Health vs HCC.
3.4 Conclusion

Disease progresses in several stages. It is important to diagnose the exact current stage of patients in order to provide proper treatments. Therefore, diagnosing of multi-stage diseases and finding of biomarkers corresponding to each stage are imperative. The work presented in this paper echoes the necessity. We proposed a new multi-class sample classification scheme with simultaneous feature selection. The classification framework is formed by the integration of a redesigned error-correcting output code (ECOC) scheme and a pairwise coupling (PWC) scheme with each scheme producing its best prediction. If the two predictions are the same, the identical class label is assigned for a test sample; otherwise, a retraining is carried out only with the two-class samples excluding samples from other classes. Also, we proposed a feature selection method, Extended Markov Blanket (EMB), within the multi-class classification framework. EMB chooses features by considering two aspects of biomarkers: redundancy and relevance. The reliability of features is also taken into consideration based on the appearance frequency. A final optimal feature set is discovered for pairwise categories. Experimental results using multi-stage liver cancer data demonstrated the performance of the proposed work. Comparison study using different multi-classification approaches was also presented. Distinct biomarker patterns were found in different stages of the disease between pair-wise categories.
CHAPTER 4

TWO-WAY SEARCH FOR BIOMARKER IDENTIFICATION

4.1 Introduction

Tandem mass spectrometry (MS/MS) is a mass spectrometry that has more than one analyzer. It has been recognized as one of the most powerful tools in proteomics for protein identification [53], [54], [55], [56]. Prior to an MS/MS experiment, proteins are digested into peptides by ionization process such as electrospray ionization (ESI) or matrix-assisted laser desorption ionization (MALDI). Tandem mass spectrometer usually has two analyzers. The first analyzer selects ions of a particular charged peptide called precursor or parent peptide according to the mass-charge ratio (m/z). The selected peptide ions are fragmented by a process known as collision-induced dissociation (CID). Once fragmented ions pass through the second analyzer, they are detected by an ion detector which is connected with a data system where mass-charge ratios are stored together with their relative abundances to generate the MS/MS spectrum.

The fragmentation of a precursor peptide bond is determined by the properties of the peptide and the energy of CID. Fig. 4.1 illustrates how a peptide with four amino acids can be cleft into different fragmentations [57], [58]. There are three different types of bonds in a peptide, i.e., CH-CO, CO-NH and NH-CH bonds. Each bond breakage produces two pieces. Therefore, there are six likely types of fragment ions for each amino acid residue: the N-terminal a, b, c fragments and C-terminal x, y, z fragments. The most common cleavage happens at CO-NH bonds by low-energy
Figure 4.1. Structure of a peptide consisting of four amino acids linked by the C-N bonds. Six different types of fragment ions are produced by CID, i.e., a, b, c type fragment ions with N-terminal and x, y, z type fragment ions with C-terminal. CID, which makes b- and y-ions dominant fragment types. For a peptide consisting of n amino acids, the possible number of fragment ions is 6(n − 1).

Though the m/z ratios of fragment ions provide informative information for protein identification, no knowledge about the cleavage positions or charge of fragments is provided. In addition, contamination and inaccuracy of instrument may generate fake m/z ratio peaks. Consequently successful identification of protein structure still remains a challenging task. Current endeavors to unambiguous peptide identification can be generalized into two classes: database search algorithms and de novo sequencing algorithms. In the first, one seeks to determine the peptide sequence by the best match by comparing the experiment spectrum with the theoretical spectrum generated from a candidate peptide list which is obtained from a protein database [56], [58], [59], [60], [61], [62]. The early popular SEQUEST computes the cross correlation value (Xcorr) between the experimental MS/MS spectrum and the hypothetical spectra generated from candidate peptides in the database with the same mass [60]. The candidate peptide producing the highest Xcorr value comes to be the first hit. However, a drawback of this algorithm is that its scoring function is not based on a rigorous probability manner. ProteinProspector is composed of several tools for mining sequence databases in conjunction with mass spectrometry.
experiments [63]. It takes into account the impact of mass measurement accuracy on protein identification experiment. Mascot computes the probability based scoring to obtain the significance of the observed match between the experimental data and mass values calculated from a candidate peptide [64]. SCOPE calculates the probability density function based on a two-step model, i.e., the probability of a particular fragmentation pattern of a peptide and the probability that the observed spectrum is generated by the fragmentation pattern of the peptide [59]. It is assumed that fragments are independent in order to make its complex probability problem computable. ProbID makes use of the Bayesian approach as the basis for the probabilistic score function [56]. It calculates the final posterior probability by considering several contributing factors. Despite the simple approach, the performance of this algorithm is comparable to industry-standard software. Lu and Chen propose a suffix tree based approach to identify peptide sequence [61]. The construction and search of suffix tree are performed within the reasonable time. To rank candidate peptide sequences, a SEQUEST-like scoring function is used. Fu et al. introduce a scoring algorithm by considering the correlative information among fragment ions to improve the peptide identification accuracy [58]. The Kernel Spectral Dot Product (KSDP) extended from SDP is used as a scoring method. The success of all these algorithms depends largely on the completeness of database and the robustness of the scoring metrics, and can not be used for the identification of proteins from unknown genomes.

On the other hand, de novo algorithms rely heavily on the MS/MS spectrum for the determination of peptide sequence, and often do not use a database. SHERENGA constructs an optimal path scoring in the spectrum graph, and automatically learns fragment types and intensity thresholds from test spectra [53]. Lutefisk converts an experimental spectrum into a spectrum graph of their corresponding b type ion masses to make a sequence graph [65]. To identify variants of known proteins in
database, sequence candidates obtained from Lutefisk can be used as input. PEAKS uses the dynamic programming to compute 10000 sequences with the highest scores [55]. It shows not only the confidence level of each output sequence but also the confidence level of each amino acid in the sequence. For each mass, this method first computes the reward and penalty. The reward is given, if there is a peak close to the mass, otherwise penalty. This algorithm tries to find a sequence such that its y and b ions maximize the total rewards at their mass values. Yan et al. propose a novel graph approach to solve the problem of separating b-ions from y-ions, in which two types of edges are considered: a type-1 edge connects two peaks possibly of the same ion types and a type-2 edge connects two peaks possibly of different ion types [66]. This algorithm does not deal with the PTM (Post-Translational Modification) problem. Jarman et al. present a partial peptide identification based on a model of random sequence probability and the evidence defined as the instances of consecutive subsequences [67]. In this paper, the sequence hierarchy is used to represent a family of candidate partial peptides. Recently, Frank and Pevzner presented a new peptide sequencing algorithm using a probabilistic network for a scoring scheme that assigns a relevance score to peptide prefix masses [68]. The probabilistic network represents three different types of relations such as correlations between fragment ions, the positional influence of the cleavage site and the influence of flanking amino acids to the cleavage site. These factors help to improve the accuracy of the peptide sequencing algorithm. Majority of de novo algorithms employs a graph theory through which the experimental spectrum is transformed into a spectrum graph. Each peak in the spectrum is converted into several nodes representing different ion types. Two nodes are connected by an edge if the mass of an amino acid is approximately equal to the difference between the two nodes. For the final resulting directed acyclic graph (DAG), each path from start node to end node corresponds to a candidate sequence.
Regardless of the different mechanisms, a lot of research focus has been put on effective scoring metrics which is doubtlessly essential for unambiguous peptide identification. However, robust selection of peptide candidates can not only improve the computation, but also can eliminate false positives. In this paper, we will present an effective and efficient two-way *de novo* searching algorithm to reduce the number of candidate sequences dramatically by utilizing the properties of MS/MS spectrum and the confidence measurement based on the intensity values of spectrum. To make a spectrum graph, the decision of start and end position is very important. We also introduce the new positioning method. Experimental results demonstrate the performance and efficacy of our novel approach and the positioning method is correct for precursor charge +1 and +2.

This paper is organized as follows. In Section 4.2, we give a detailed description of our algorithm which embodies properties of MS/MS spectrum, relation between the precursor m/z and mass of peptide, normality test, and an elaboration of our two-way searching algorithm. Then in Section 4.3, we test our algorithm on public data, and finally we close our paper by conclusion and future work.

### 4.2 Algorithms

#### 4.2.1 Random peptide sequence denotation

We will first introduce several peptide notations before moving on to the proposed algorithms. Let $\Sigma$ be the alphabet set which consists of 20 amino acids. And each amino acid has a distinctive mass which is represented by $m(a), a \in \Sigma$. By denoting the product of $\Sigma_1$ and $\Sigma_2$ as $\Sigma_1 \times \Sigma_2 = \{ab \mid a \in \Sigma_1 \text{ and } b \in \Sigma_2\}$, the following expressions $\Sigma^1 = \Sigma$, $\Sigma^2 = \Sigma \times \Sigma$, and $\Sigma^n = \Sigma \times \Sigma^{n-1}$ are possible. $\Sigma^n$ means a set that includes all possible sequences with length $n$, whose number of
elements $|\Sigma^n|$ is $20^n$. So the set $\Sigma^+$ consisting of all possible sequences with different lengths made by 20 amino acids can be written as

$$\Sigma^+ = \Sigma^1 \cup \Sigma^2 \cup \ldots \cup \Sigma^n \cup \ldots = \bigcup_{i=1}^{\infty} \Sigma^i. \quad (4.1)$$

Obviously $\Sigma = \{A, C, \ldots, Y\}$, and the power expressions of $\Sigma$ indicating peptides with different lengths are defined recursively as $\Sigma^1 = \{A, C, \ldots, Y\}$, $\Sigma^2 = \{AA, AC, \ldots, AY, CA, CC, \ldots, CY, YA, YC, \ldots, YY\}$, etc.. The union of all possible peptides is $\Sigma^+ = \{A, C, \ldots, Y, AA, AC, \ldots, YY, AAA, AAC, \ldots, YYY, \ldots\}$.

A parent peptide $P = p_1 p_2 \ldots p_n$ is a sequence of amino acids, which consists of $n$ amino acids. And the mass of peptide $P$ is $m(P) = \sum_{i=1}^{n} m(p_i)$ where $P \in \Sigma^n$ and the mass of each amino acid is $m(p_i)$. Let $T$ be an element of $\Sigma^+$, i.e., $T \in \Sigma^+$. If the mass of $T$ is approximately equal to that of the target peptide $P$, i.e., $|m(T) - m(P)| < \epsilon$, $T$ will be one candidate sequence with respect to the parent peptide. $\epsilon$ is the tolerant error the mass spectrometer has.

**4.2.2 Properties of MS/MS spectrum**

The actual mass of peptide $P$ can be written as $18+m(P)$. Number 18 comes from two extra hydrogen atoms and one extra oxygen atom at the C- and N-terminals where the mass of one hydrogen atom is approximately 1 Da (Dalton) and the mass of one oxygen atom is approximately 16 Da. The mass of b-ion with $i$ amino acids, represented by $b_i$, can be computed as

$$b_i = 1 + m(p_1) + \cdots + m(p_i) = 1 + \sum_{j=1}^{i} m(p_j) \quad (4.2)$$
Figure 4.2. Hypothetical MS/MS spectrum and amino acid sequences.

where mass 1 comes from one hydrogen atom attached to the b-ion type fragments. Similarly, the mass of y-ion with \( i \) amino acids, denoted by \( y_i \), can be calculated by

\[
y_i = 19 + m(p_{n-i+1}) + \cdots + m(p_n) = 19 + \sum_{j=n-i+1}^{n} m(p_j)
\]

(4.3)

where mass 19 is due to three hydrogen atoms and one oxygen atom linked to the y-ion type fragments.

The difference of \( m/z \) ratio of two adjacent singly-charged b- or y-ions is the exact mass of one residue. However, in real MS/MS spectra, an ion may be charged with different values \((+1, +2, \ldots)\), which make several different peaks. In addition, there is no information about the fragment ion type (b, y, \ldots) and fragmenting position. For a tandem mass spectrometry, we suppose that each fragment ion has a unique mass-charge ratio and each amino acid is fragmented. Therefore, the \( m/z \) value of an ion is equal to the mass of the ion. Before introducing our Two-way Searching Algorithm in Section 4.2.5, following two properties will be presented:

**Property One:** If the mass-charge ratio of precursor is given, then the positions of start and end nodes for both b- and y-ion series are known. (More details will be
provided in Section 4.2.3).

**Proof:** Start nodes are always at 1 \( m/z \) for b-ion series and 19 \( m/z \) for y-ion series in the spectrum because of the extra attachments as explained above. And the end nodes of b- and y-ion series will happen at \( m(P) + 1 \) \( m/z \) and \( m(P) + 19 \) \( m/z \), respectively by Eq. (4.2) and (4.3). Therefore, the \( m/z \) positions of \( b_1, y_1, b_{n-1}, \) and \( y_{n-1} \) can be expressed as \( b_1 = 1 + m(p_1), y_1 = 19 + m(p_n), b_{n-1} = m(P) + 1 - m(p_n), \) and \( y_{n-1} = m(P) + 19 - m(p_1) \) respectively.

Furthermore, let \( S = \{ (S_i, I_i) \mid 1 \leq i \leq k \} \) be an MS/MS spectrum ordered by \( m/z \) values where \( S_i \) and \( I_i \) denote the position and intensity of the \( i \)-th peak, and \( k \) is the number of peaks. Then the position of \( b_i \) in the spectrum is determined by the \( m/z \) value with the largest intensity \( I_j \) within the tolerant error.

\[
b_i = \arg\max_{S_j} \{ I_j \mid |S_j - b_{i-1} - m(a)| < \epsilon \}
\]  

(4.4)

where \( a \in \Sigma \). The same rule is applied to \( y_i \) position locating.

**Property Two:** There exists a pair-wise relationship between peaks in the b- and y-ion series. That is, the sequence identified in the b-ion series is the same as that identified in the y-ion series in the reversed order.

**Proof:** We can derive the equation \( b_i + y_{n-i} = 20 + m(P) \) from Eq. (4.2) and (4.3). Therefore, \( b_i \) and \( y_i \) can be expressed as \( b_i = 20 + m(P) - y_{n-i} \) and \( y_i = 20 + m(P) - b_{n-i} \), respectively. This means that the sequence of b-ion series is the same as that of y-ion series in the reverse order. Fig. 4.2 shows the two properties of an MS/MS spectrum.
Table 4.1. Upper tail percentage points for Anderson-Darling statistic $A^*$.

<table>
<thead>
<tr>
<th>$\alpha$</th>
<th>0.2</th>
<th>0.15</th>
<th>0.1</th>
<th>0.05</th>
<th>0.025</th>
<th>0.01</th>
<th>0.005</th>
</tr>
</thead>
<tbody>
<tr>
<td>$A^*_\alpha$</td>
<td>0.509</td>
<td>0.561</td>
<td>0.631</td>
<td>0.752</td>
<td>0.873</td>
<td>1.035</td>
<td>1.159</td>
</tr>
</tbody>
</table>

4.2.3 Relation between the precursor m/z and mass of peptide

To simultaneously identify peptide from both the start and end nodes, knowledge of the positions of these nodes is important. Since the end nodes of b- and y-ion series happen at $m(P) + 1$ m/z and $m(P) + 19$ m/z, information about the mass of the target peptide sequence is required which can be obtained based on the precursor information. Let the m/z of precursor be $P^z$, where $z$ is the charge of the ion. When $z = 1$, the positions of end node of b- and y-ion series are $P^1 - 18$ and $P^1$, because of $P^1 - 19 \approx m(P)$.

For $z \geq 2$ the above equation can not be used any more. Through experiment, the following equation is formed based on some heuristic $\delta$ value:

$$P^2 \times 2 - \delta - 19 \approx m(P)$$

where $0.9 \leq \delta \leq 1.0$. Therefore, for $P^2$ the positions of end node of b- and y-ion series are $P^2 \times 2 - \delta - 18$ and $P^2 \times 2 - \delta$, respectively. The position of start node in $P^2$ is the same as that of $P^1$. We used $\delta = 0.95$ in our experiment and showed $\delta$ can be used as the pertinent adjusted value.

4.2.4 Normality Test

One assumption we used in our de novo peptide sequencing is that the measured mass-charge ratio can be modelled as a normal distribution as other researchers have done [59], [63]. We performed a goodness of fit test to confirm whether we can use
Table 4.2. Normality test for distribution of measured mass-charge ratios.

<table>
<thead>
<tr>
<th>$i$</th>
<th>$X_{(i)}$</th>
<th>$(X_{(i)} - \mu)/\sigma$</th>
<th>$F(Z_{(i)})$</th>
<th>$\ln(F(Z_{(i)})) + \ln(1 - F(Z_{(r+1-i)}))$</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>0.006</td>
<td>-1.168</td>
<td>0.121</td>
<td>-5.314</td>
</tr>
<tr>
<td>2</td>
<td>0.009</td>
<td>-1.119</td>
<td>0.132</td>
<td>-12.626</td>
</tr>
<tr>
<td>3</td>
<td>0.028</td>
<td>-0.844</td>
<td>0.199</td>
<td>-17.774</td>
</tr>
<tr>
<td>4</td>
<td>0.037</td>
<td>-0.715</td>
<td>0.237</td>
<td>-16.947</td>
</tr>
<tr>
<td>5</td>
<td>0.043</td>
<td>-0.634</td>
<td>0.263</td>
<td>-19.084</td>
</tr>
<tr>
<td>6</td>
<td>0.089</td>
<td>0.029</td>
<td>0.512</td>
<td>-15.255</td>
</tr>
<tr>
<td>7</td>
<td>0.094</td>
<td>0.110</td>
<td>0.544</td>
<td>-11.885</td>
</tr>
<tr>
<td>8</td>
<td>0.109</td>
<td>0.321</td>
<td>0.626</td>
<td>-11.097</td>
</tr>
<tr>
<td>9</td>
<td>0.161</td>
<td>1.065</td>
<td>0.857</td>
<td>-6.411</td>
</tr>
<tr>
<td>10</td>
<td>0.171</td>
<td>1.210</td>
<td>0.887</td>
<td>-4.958</td>
</tr>
<tr>
<td>11</td>
<td>0.208</td>
<td>1.744</td>
<td>0.959</td>
<td>-3.588</td>
</tr>
</tbody>
</table>

the normal distribution as analysis model for our experiment data set or not. There are several approaches for assessing the underlying distribution of a data set. Among them, Anderson-Darling (AD) test which belongs to a class of distance test is known as a more powerful test than other distance tests. AD test shows a good performance in small samples as well as large samples. When the number of measured mass-charge ratios with respect to the center one within the tolerant error is sparse, AD test is more appropriate because of applying the cumulative distribution function (CDF) of the data set.

To test the normality, we define hypotheses:

$H_0$ : The distribution for the data set is a normal distribution.

$H_1$ : The distribution for the data set is a non-normal distribution.

Let $X$ be a random sample $X = (X_{(1)}, X_{(2)}, \ldots, X_{(r)})$ sorted in the ascending order with sample size $r$. The standardized value is $Z_{(i)} = (X_{(i)} - \mu)/\sigma$ where $\mu$ and $\sigma$ denote mean and variance for the sample data. The AD normality test calculates the following function:
Figure 4.3. Measured mass-charge ratios (solid line bar) and normal distribution (dotted curve).

\[
AD = -\frac{1}{r} \left\{ \sum_{i=1}^{r} (2i - 1) \ln F[Z(i)](1 - F[Z(r+1-i)]) \right\} - r \tag{4.6}
\]

where \( F \) is the standard normal cumulative probability and \( \ln \) is the natural logarithm (base e). Eq. (4.6) is modified by computing

\[
A^* = AD(1 + \frac{0.75}{r} + \frac{2.25}{r^2}). \tag{4.7}
\]

If \( A^* \) exceeds the selected critical values given in Table 4.1, we will reject the null hypothesis at the 100\( \alpha \)% level.

As an example for peptide YLYELAR shown in Fig. 4.3, we normalize the intensities of all peaks such that the highest peak has one and extract those with more than 1 % (0.00208) of maximum intensity (0.208). So we obtain the sample data
Mean and variance of these intensities are $\mu = 0.087$ and $\sigma = 0.069$. We obtained $AD = 0.358$ and $A^* = 0.389$ through the procedure shown in Table 4.2. At level of significance 0.1, $A^* = 0.389 \leq \alpha = 0.631$. Therefore, we can assume normality for distribution of measured mass-charge ratios. Fig. 4.3 represents the distribution of measured mass-charge ratios and normal distribution. In general, if the p-value is 0.1 or more, we can assume normality.

We found most measured mass-charge ratios are self-centered normal distribution. Therefore the application of normal distribution as the fundamental frame in the following scoring function is legitimate.

4.2.5 Two-way searching algorithm

4.2.5.1 Peptide candidate initial filtering by two-way searching

Our new two-way searching algorithm for MS/MS peptide sequencing begins with both start and end position localizations. In our approach, the positions of start and end nodes for b-ion and y-ion are determined in advance in the MS/MS spectrum, i.e., at 1 and $m(P) + 1 \text{ m/z}$ for b-ion series, and at 19 and $m(P) + 19 \text{ m/z}$ for y-ion series as shown in Fig. 4.2. During our two-way parallel searching, these four initial nodes will extend simultaneously by scanning the whole spectrum, where start nodes for b- and y-ion series proceed simultaneously in the forward direction, and end nodes in the backward direction at the same time. This procedure keeps going until some requirements are met. We denote the direction from low m/z to high m/z as the forward direction and from high m/z to low m/z as the reverse direction.
Figure 4.4. Two-way searching algorithm. Sb and Eb represent the start and end nodes of b-ion series, respectively. Sy and Ey represent the start and end nodes of y-ion series, respectively. Amino acid sets $F_b$ and $R_y$ always have the same amino acids. Same to sets $R_b$ and $F_y$. Amino acids with gray color are deleted in the comparison procedure and amino acids with thick boundaries are removed in the further pruning procedure.

Four amino acid sets generated in the process of graph extension are denoted as $F_b$ (forward for b-ion), $R_b$ (reverse for b-ion), $F_y$ (forward for y-ion), and $R_y$ (reverse for y-ion) as shown in Fig. 4.4. At every extension of the graph, the candidate amino acids of $F_b$ are compared with those of $R_y$. The common amino acids are kept and new nodes are added in positions corresponding to the $m/z$ values. And the amino acids which are not in common are eliminated to reduce the computational burden. Likewise, the amino acids of $R_b$ are compared with those of $F_y$ simultaneously.
In stead of exhaustively checking all possible paths, we reduce the number of nodes in the spectrum graph effectively through such a method. Consequently it reduces the number of candidate sequences and computational cost to determine a sequence with the best score. After the successive progress, when the nodes of $F_b$ meet those of $R_b$ within the error range, \( |b_i - b_j| < \epsilon \), \( b_i \in F_b \), and \( b_j \in R_b \), we merge two nodes into one and trace back in the two-way direction while storing the amino acids.

Finally, the two partial amino acids are concatenated into one complete sequence which will be used as one candidate sequence. It is also possible that these processes confront with a distance, which corresponds to one amino acid between $F_b$ and $R_b$ nodes, \( |b_i - b_j - m(a)| < \epsilon \), \( b_i \in F_b \), \( b_j \in R_b \), and \( a \in \Sigma \). If nodes of both sides cross over, the nodes disappear from the graph. The number of steps to obtain amino acid sequences of the same length as the target peptide is \( \lceil n/2 \rceil \).

The same rule is applied to $F_y$ and $R_y$. Fig. 4.4 shows a diagram representing our algorithm.

Since the b- and y-ions may lose a water or ammonia molecule, it is necessary to employ all the related ion types for the series. Ions b, b - H$_2$O, and b - NH$_3$ for b-ion series are considered in the forward and reverse direction, which are the most frequent N-terminal ions, and y, y - H$_2$O, and y - NH$_3$ for y-ion series. In addition, we consider the a-ion which is also a dominant ion. After the initial filtering, we have a reduced amino acid candidate pool. Next we will introduce further pruning of candidate amino acids by a scoring function to determine the best sequence.

4.2.5.2 Scoring function for final candidate screening

By utilizing the piece-wise local region intensity values of MS/MS, we define an evidence based scoring function for screening out the best optimal peptide candidate.
Based on the normality test verification introduced in Section 4.2.4, we can let $I_{bi}$ be the Gaussian sum of all peak intensities close to $b_i$ within a tolerant error $\epsilon$:

$$I_{bi} = \sum_{b_i - \epsilon}^{b_i + \epsilon} \frac{I_j}{I_M} \exp\left(-\frac{(S_j - S_{bi})^2}{2\sigma^2}\right)$$  \hspace{1cm} (4.8)

where $I_M$ is the highest intensity in the whole spectrum and $I_j$ is the individual peak intensity of the local region. In the procedure of normalization, intensities of all peaks are divided by the highest intensity. Standard deviation $\sigma$ represents to which extent the peak positions in the experimental spectrum deviate from the theoretical ones. Without loss of generality based on the normality test, we can assume the peak having the highest intensity in the whole spectrum is most likely to be a real fragment and neighbor peaks close to the peak will form the normal distribution. Standard deviation $\sigma$ can be heuristically determined by

$$\sigma_{\text{min}} = \arg\min_{\sigma} \sum_{b_i - \epsilon}^{b_i + \epsilon} \left(\exp\left(-\frac{(S_j - S_M)^2}{2\sigma^2}\right) - \frac{I_j}{I_M}\right)$$  \hspace{1cm} (4.9)

where $S_M$ is the m/z value corresponding to the highest intensity $I_M$.

Let $x$ be the mass of b-ion, then the masses of a-ion, b - H2O, and b - NH3 are $x - 28$, $x - 18$, and $x - 17$, respectively, i.e., differences $\Delta = \{-28, -18, -17\}$. Given the $b_i$ we define

$$I_{bi - NH3} = \sum_{S_{bi - NH3}} I_j \exp\left(-\frac{(S_j - S_{bi - NH3})^2}{2\sigma^2}\right)$$  \hspace{1cm} (4.10)

where $S_{bi - NH3} = \arg\max_{S_j} \{I_j \mid |b_i - S_j - 17| < \epsilon\}$.

Likewise, we can compute $I_{bi - H2O}$, $I_{bi - a}$, $I_{yi}$, $I_{yi - H2O}$ and $I_{yi - NH3}$. The total intensity of ions related to $b_i$ is

$$I_{Tbi} = I_{bi} + I_{bi - NH3} + I_{bi - H2O} + I_{bi - a}.$$  \hspace{1cm} (4.11)
And $I_{Tyi}$ is expressed as follows

$$I_{Tyi} = I_{yi} + I_{yi-NH_3} + I_{yi-H_2O}.$$  \hspace{1cm} (4.12)

The total intensity of b-ion series in the forward direction is

$$I_{Fb} = \sum_{i=1}^{n/2} I_{Tbi}$$ \hspace{1cm} (4.13)

where $n$ is the number of steps to obtain the peptide sequence. The total intensity of b-ion series in the reverse direction is

$$I_{Rb} = \sum_{i=(n/2)+1}^{n-1} I_{Tbi}.$$  \hspace{1cm} (4.14)

Similarly, we can compute $I_{Ry}$ and $I_{Fy}$. A legitimate intensity scoring function is obtained by summing the four direction total intensities and the top scoring sequence among all candidate sequences becomes the best candidate:

$$Scoring = I_{Fb} + I_{Rb} + I_{Fy} + I_{Ry}.$$ \hspace{1cm} (4.15)

This scoring function is reasonable because many of the noise peaks have a low intensity value. By incorporating abundance difference of b- and y-ions, i.e., y-ions are usually more ample than b-ions, we may apply different weights to the total summing intensity of b-ion series and the total summing intensity of y-ion series. Then, the scoring function can be modified as

$$Scoring = \lambda(I_{Fb} + I_{Rb}) + (1 - \lambda)(I_{Fy} + I_{Ry}).$$ \hspace{1cm} (4.16)

Obviously $\lambda$ is set as less than or equal to 0.5.

Before the scoring function is applied, for further pruning of candidate pool, we define a screening criterion from the calculation of total intensities $I_{Tbi}$ and $I_{Ty(n-i)}$.
of ions in the $i$-th step of graph extension where $|b_j - b_{j-1}| \approx |y_{n-j+1} - y_{n-j}| \approx m(a)$ with $b_{-1} = S_b$, $y_n = E_y$ and $1 \leq j \leq i$. This screening criterion is used at every step of node extension of the spectrum graph. Suppose there exist $q$ candidate amino acids in the $i$-th step after eliminating amino acids which are not in common between $Fb$ and $Ry$. Let $I_i$ be a set whose elements indicate the sum of $I_{Tb_i}$ and $I_{Ty(n-i)}$ where $b_i$ and $y_{n-i}$ are a complementary ion pair.

$$I_i = \{I_{Tb_i}^1 + I_{Ty(n-i)}^1, \ldots, I_{Tb_i}^q + I_{Ty(n-i)}^q\} \quad (4.17)$$

The elements in set $I_i$ are sorted in ascending order, and the first $\beta\%$ amino acids are removed from $Fb$ and $Ry$. The same rule is applied to every step of $Fy$ and $Rb$.

### 4.3 Experimental Results

The two-way peptide sequencing algorithm was implemented by using C++ codes. We employed nine data sets whose ground truths were given, among which

<table>
<thead>
<tr>
<th>Spectrum</th>
<th>$z$</th>
<th>Correct sequence</th>
<th>Two-way searching algorithm</th>
<th>Lutefisk</th>
</tr>
</thead>
<tbody>
<tr>
<td>634.4</td>
<td>1</td>
<td>IFVQK</td>
<td>LFVQK</td>
<td>LFVQK</td>
</tr>
<tr>
<td>678.3</td>
<td>1</td>
<td>YIPGTK</td>
<td>YLPGTK</td>
<td>YLPGTK</td>
</tr>
<tr>
<td>779.4</td>
<td>1</td>
<td>MIFAGIK</td>
<td>MLFAGLK</td>
<td>[244.12]</td>
</tr>
<tr>
<td>927.4</td>
<td>1</td>
<td>YLYEIR</td>
<td>YLYELAR</td>
<td>[276.11]</td>
</tr>
<tr>
<td>584.8</td>
<td>2</td>
<td>TGPNLHLGLFGR</td>
<td>TGPNLHLGLFGR</td>
<td>[409.25]</td>
</tr>
<tr>
<td>689.9</td>
<td>2</td>
<td>HGTVVLTAAGGLKLK</td>
<td>HGTVVTLAAGGLKLK</td>
<td>[194.08]</td>
</tr>
<tr>
<td>728.8</td>
<td>2</td>
<td>TGQAQFSYTDANK</td>
<td>TGQAQFSQPSQPNK</td>
<td>AQGTV[H]K</td>
</tr>
<tr>
<td>792.9</td>
<td>2</td>
<td>KTGAGAPAMQGDAN</td>
<td>KTGAGAPAMQGDAN</td>
<td>[209.58]</td>
</tr>
<tr>
<td>943.0</td>
<td>2</td>
<td>YLEFSDAHHLHLSK</td>
<td>YLEFSLATTLHHLHLSK</td>
<td>[222.56]</td>
</tr>
</tbody>
</table>

Table 4.3. Experimental results of peptide sequencing with $\beta = 35\%$ ($\beta = 20\%$ for $678.3 \ m/z$) in our method. For comparison, Lutefisk was performed. We cannot distinguish between the isobaric amino acid pairs leucine (L) and isoleucine (I), and glutamine (Q) and lysine (K) as with most *de novo* methods where $m(I)=m(L)=113.16$ Da, $m(Q)=128.13$ Da, $m(K)=128.17$ Da. The value shown in brackets represents an approximate mass of the remaining amino acid residues.
Table 4.4. The number of candidates and rankings as the screening ratio changes. The numerator indicates the ranking of the correct sequence our algorithm made, and the denominator represents the total number of candidate sequences. 0% means no screen is adopted.

<table>
<thead>
<tr>
<th>Spectrum</th>
<th>0%</th>
<th>10%</th>
<th>20%</th>
<th>30%</th>
<th>40%</th>
<th>50%</th>
<th>60%</th>
</tr>
</thead>
<tbody>
<tr>
<td>634.4</td>
<td>1/91</td>
<td>1/27</td>
<td>1/21</td>
<td>1/19</td>
<td>1/14</td>
<td>1/12</td>
<td>0/6</td>
</tr>
<tr>
<td>678.3</td>
<td>1/1</td>
<td>1/1</td>
<td>1/1</td>
<td>0/0</td>
<td>0/0</td>
<td>0/0</td>
<td>0/0</td>
</tr>
<tr>
<td>779.4</td>
<td>2/173</td>
<td>2/152</td>
<td>2/93</td>
<td>2/64</td>
<td>2/10</td>
<td>2/6</td>
<td>2/6</td>
</tr>
<tr>
<td>927.4</td>
<td>1/5197</td>
<td>1/819</td>
<td>1/360</td>
<td>1/239</td>
<td>1/195</td>
<td>1/126</td>
<td>1/24</td>
</tr>
</tbody>
</table>

four data sets with precursor charge +1 and five data sets of +2. These data sets were obtained from QSTAR instrument. In this study, we used $\epsilon = 0.3$ and $\lambda = 0.5$.

Table 4.3 shows results of our algorithm with $\beta = 35\%$ ($\beta = 20\%$ for spectrum 678.3 $m/z$) as a screening criterion on $\delta = 0.95$. To show the result which is most optimal to benchmark, instead of listing the most optimal identified peptide, i.e., the one with highest scoring value or with ranking 1, we show the sequence which is within ranking 3 range. We compared results of the proposed method with those of Lutefisk [65]. Lutefisk converts an experimental spectrum into a sequence graph where partial sequences are examined. To reduce candidates, after finding complete sequences, sequences that appear to have been derived from alternating b-type and y-type ions and that are derived mostly from the low-mass fragments are discarded. Finally, Lutefisk yields at most 50 candidates. Users can set the parameter. When we set the parameter as the maximum value, 50, 45, 50 and 50 candidates were made in 634.4, 678.3, 779.4 and 927.4 $m/z$, respectively.

For precursor charge +2, we found that there exists an equation $P^2 \times 2 - \delta - 19 \approx m(P)$ where $0.9 \leq \delta \leq 1.0$. We used $\delta = 0.95$ to determine the position of end node in the graph extension. This value is greater than our tolerant error 0.3. Therefore
Figure 4.5. Example of peptide sequence YIPGTK in the graph extension: (a) The sequence of b-ion series; (b) The sequence of y-ion series. Amino acids with triangle symbols are deleted in the procedure of comparison and amino acids with star symbols are removed from the candidate amino acid pool in the procedure of further pruning. Amino acids represented in black circles are reserved. Thick line presents the path of one candidate sequence.

if we use $P^2 \times 2 - 19 \approx m(P)$ as the position of end node as in other methods which proceed in the forward direction only, our algorithm will not provide good answers. Thus the positions of end node of b- and y-ion series are respectively $P^2 \times 2 - 0.95 - 18$ and $P^2 \times 2 - 0.95$. For example for spectrum 584.8 m/z, the end nodes of b- and y-ion series come to appear in the $584.8 \times 2 - 0.95 - 18 = 1150.65$ m/z and $584.8 \times 2 - 0.95 = 1168.65$ m/z. The position of start node in $P^2$ is the same as that of $P^1$. Our algorithm came up six sequences identical with ground truth.
Table 4.4 shows the number of candidates change as the screening criterion $\beta$ varying from 0% to 60% for the charged +1 sequences. Although $\beta$ changes, the rankings remain unchanged. For spectrum 678.3 $m/z$, since the number of candidates after initial two-way searching was reduced to only one as seen in Fig. 4.5, no screen was applied. With 10% further pruning at every step of graph extension, we can reduce the number of candidates significantly up to 70% and 84% for spectra 634.4 and 927.4 $m/z$, respectively. Therefore, by adopting a proper screening criterion to our algorithm, we can reduce the processing time effectively.

4.4 Conclusion

In this paper, we presented a novel de novo approach called two-way searching algorithm for determining the sequence of peptide. The main contribution of this paper lies in the greatly reduced number of peptide candidates. Based on the property that the same identification of peptide sequence will be resulted from b-ion or y-ion series, we obtained a list of peptide candidates by simultaneously searching from four different start and end positions and filtering out peptide sequences which violate this property. The initially filtered candidate pool is further pruned by incorporating a screening criterion based on the local region intensities of the spectrum. The final optimal best candidate is singled out based on the highest confidence defined as the global intensity from two-way search results. Contributions of this paper also come from the determination of the end nodes for b and y-ion series in the case of the charged +2 precursor. For the future work, we will improve our algorithm by introducing gap edges corresponding to the di- and tri-peptides in a spectrum graph. Also we will modify the scoring algorithm by using a probabilistic model to make the pruning more robust.
CHAPTER 5

FAST KERNEL DISCRIMINANT ANALYSIS FOR DIAGNOSIS OF
ALZHEIMER DISEASE STAGE USING MASS SPECTRA

5.1 Introduction

Alzheimer’s disease (AD) is characterized by progressive memory loss and other impaired abilities to carry out daily activities. AD is the most common form of dementia in people over age 65. The risk increases with age. It is estimated that as many as 4.5 million in the United States suffer from the disease. One of the causes of AD is genetic mutation which leads to accumulation of beta amyloid protein in the brain. Amyloid plaques and neurofibrillary tangles are considered signs of AD. Unfortunately, since the ultimate cause and treatment of AD are not known, it is urgent to identify biomarkers for the disease in order to accelerate drug and therapeutic development.

Matrix-assisted laser desorption/ionization (MALDI) and surface enhanced laser desorption/ionization (SELDI) time-of-flight (TOF) mass spectrometry have increasingly been used to early disease diagnosis, monitoring disease progression and therapeutic effects of drugs [69], [70]. Currently, MALDI TOF/TOF based platform has been used for quantitative proteomics analysis. In our recent study for neurodegenerative diseases such as AD and Parkinson’s disease (PD), a set of candidate biomarkers were detected and validated using LC MALDI TOF/TOF based targeted quantitative proteomics platform [71].

LDA is a traditional statistical scheme for feature reduction which has been widely used in a diversity of application areas such as face recognition [72], [73],
microarray data classification [74] and text classification [75]. In a case where the dimensionality exceeds the sample size, however, the classical LDA faces a problem known as singularity [76]. Since the dimensionality of the mass spectrometry data is considerably huge, the singularity problem necessarily happens. Several methods were proposed to overcome the limitation. Li et al. proposed a new LDA method named generalized linear discriminant analysis (GLDA) which solves the singularity problem and is fast in the calculation of eigenvectors [77]. Ye et al. developed uncorrelated linear discriminant analysis (ULDA) based on generalized singular value decomposition (GSVD) which was tested with microarray datasets [74]. Another drawback of the classical LDA is its linear property with which LDA fails for nonlinear problems [78]. To solve the problem, nonlinear based LDA methods were proposed. The main idea is to map the input space into a high dimensional feature space. In the feature space, the classical LDA is performed. Baudat and Anouar proposed a nonlinear extension of the classical LDA called generalized discriminant analysis (GDA) where experimental data in the feature space is centered by shifting feature vectors by the global centroid vector [79]. Mika et al. developed a nonlinear extension of Fisher’s linear discriminant analysis called kernel Fisher discriminant analysis (KFD). However, KFD was designed only for two class problems [80].

In this paper, we propose a new fast kernel discriminant analysis (FKDA) which is a nonlinear extension of GLDA. To tackle the large number of peaks and noise in high-resolution MALDI-TOF data, we have developed a multi-step feature selection algorithm. We apply FKDA to the classification of AD dataset analyzed by Lopez et al. which consists of three classes, mild cognitive impairment (MCI), AD and normal. However, they analyzed the dataset as a binary class problem combining MCI and AD [10]. In this study, we analyze the Alzheimer dataset as three classes.
5.2 Preprocessing

In this study, we use \( m/z \) values over the mass range 1k-10k Da (Dalton) in which a raw mass spectrum contains over 750,000 \( m/z \) values. To reduce the noise and dimensionality of the raw spectrum, we employ a three-step preprocessing procedure: (1) binning, (2) baseline correction and (3) normalization. Another objective in such preprocessing tasks is to improve the performance of identifying the disease.

In the first step, binning of raw mass spectra is performed. Starting from 1k Da, a bin size of 1 Da is used, which yields about 9000 bins. For the binned spectrum, the \( m/z \) ratio represents the left boundary of an interval. The binning not only makes the spectrum smooth but also aligns \( m/z \) values in all spectra so that it facilitates the analysis of mass spectrometry data. For baseline correction, we find a minimum intensity per every 10 bins, proceeding from low mass to high mass range. And then, the baseline is estimated by fitting a fourth order polynomial to 20 minimum intensities. The regressed baseline is subtracted from the binned spectrum. Furthermore, we choose to normalize the baseline corrected spectrum because the amount of proteins varies depending on serum samples. Each spectrum is normalized by dividing the baseline corrected spectrum by its total ion current (the summed intensity over all \( m/z \) values in the baseline corrected spectrum). Because of the very small normalized intensity value, all the intensities are multiplied by 1000 for computational convenience. As shown in Fig. 5.1 (d), our preprocessing task appears to be satisfactory because almost all significant peaks are retained.

5.3 Method

Classifiers based on discriminative learning try to find a decision boundary that maximizes separation between classes. SVM is a popular learning algorithm to
Figure 5.1. Mass spectrum of a sample in Alzheimer dataset over the mass range 1k-10k Da: (a) raw spectrum, (b) binned spectrum, (c) baseline corrected spectrum, (d) normalized spectrum. In (b), the red line indicates the baseline.

solve two-class classification problems [27], [38], [28], [37]. An optimal hyperplane that separates a given set of binary labeled training data is built by maximizing margin between two classes. LDA is very similar to SVM in the underlying idea. A distinct difference is that LDA looks for the decision boundary using all training samples, while SVM uses only support vectors that are close to the decision boundary. Also SVM was basically designed for two-class classification problems, while LDA has the ability to classify multi-class samples. However, the classical LDA
might fail in nonlinear problems due to its linearity property. Another issue is that in many real datasets where the dimensionality is larger than the number of samples the singularity problem arises. In this study, we propose a novel kernel based discriminant analysis algorithm for classifying multi-class samples that can overcome the singularity problem in the classical LDA with an efficient method.

A given data set of $n$ samples, each of which consists of $l$ features, is represented as $X = [x_1, \cdots, x_n] \in \mathbb{R}^{l \times n}$. Suppose that the data set is clustered to $c$ classes $[N_1, \cdots, N_c]$ where each class $X_i$ has $n_i$ samples, i.e., $X = \bigcup_{i=1}^{c} N_c$ and $n = \sum_{i=1}^{c} n_i$. In nonlinear discriminant analysis, the input data space $\mathbb{R}^l$ is mapped into the higher feature space $F \subset \mathbb{R}^m$ via a nonlinear mapping function $\Phi$. Unfortunately, we cannot explicitly observe the data in the feature space $F$. An alternative method is to use the kernel function that defines the inner product in the feature space. More specifically, if the kernel function $k$ satisfies Mercer’s condition, there exists a mapping $\Phi$ such that

$$k(x, y) = <\Phi(x), \Phi(y)>.$$  \hspace{1cm} (5.1)

The main idea of the nonlinear discriminant analysis is to solve the LDA problem in the feature space by maximizing the following Fisher criterion:

$$J^\phi(W) = \frac{W^T S_b^\phi W}{W^T S_t^\phi W}$$ \hspace{1cm} (5.2)

where $W$ is a linear transformation matrix; $S_b^\phi$ and $S_t^\phi$ are the between-class and total scatter matrices obtained in the feature space. Suppose that $p_i, \mathbf{m}_i^\phi, \mathbf{m}^\phi$ are a prior probability of the $i$th class, the mean vector of the training samples of the $i$th class, and the mean vector across all training samples in the feature space, respectively. Then, $S_b^\phi$ and $S_t^\phi$ are defined as follows:

$$S_b^\phi = \sum_{i=1}^{c} p_i (\mathbf{m}_i^\phi - \mathbf{m}^\phi)(\mathbf{m}_i^\phi - \mathbf{m}^\phi)^T,$$ \hspace{1cm} (5.3)
\[ S^\Phi_t = \frac{1}{n} \sum_{i=1}^{n} (\Phi(x_i) - m^\Phi)(\Phi(x_i) - m^\Phi)^T, \quad (5.4) \]

where \( p_i = \frac{n_i}{n} \), \( m^\Phi_i = \frac{1}{n_i} \sum_{j \in N_i} \Phi(x_j) \), and \( m^\Phi = \frac{1}{n} \sum_{i=1}^{n} \Phi(x_i) \). The scatter matrices \( S^\Phi_b \) and \( S^\Phi_t \) can be decomposed as

\[ S^\Phi_b = M_b M_b^T \quad \text{and} \quad S^\Phi_t = M_t M_t^T \quad (5.5) \]

where

\[
M_b = \begin{bmatrix} \sqrt{p_1} (m^\Phi_1 - m^\Phi) \\ \vdots \\ \sqrt{p_c} (m^\Phi_c - m^\Phi) \end{bmatrix} \in \mathbb{R}^{m \times c}, \\
M_t = \frac{1}{\sqrt{n}} \begin{bmatrix} (\Phi(x_1) - m^\Phi) \\ \vdots \\ (\Phi(x_n) - m^\Phi) \end{bmatrix} \in \mathbb{R}^{m \times n}.
\]

By solving the generalized eigenvalue equation, the optimal discriminant vector \( W \) corresponding to the eigenvector of the equation can be obtained

\[ S^\Phi_b W = S^\Phi_t W \Lambda, \quad W = [W_1, \ldots, W_{c-1}] \in \mathbb{R}^{m \times (c-1)} \quad (5.6) \]

where \((S^\Phi_t)^{-1} S^\Phi_b\) has at most \( c - 1 \) non-zero eigenvalues \((m >> c)\). It is possible to express the eigenvector as a linear combination of the observations in the feature space. Hence we have

\[ W_j = \sum_{i=1}^{n} \alpha_{ij} \Phi(x_i) \quad (5.7) \]

and

\[ W = \begin{bmatrix} \sum_{i=1}^{n} \alpha_{i1} \Phi(x_i) \\ \vdots \\ \sum_{i=1}^{n} \alpha_{i(c-1)} \Phi(x_i) \end{bmatrix} = Q \alpha \quad (5.8) \]

where \( \alpha = [\alpha_1, \ldots, \alpha_n]^T \) is a coefficient vector and \( Q = [\Phi(x_1), \ldots, \Phi(x_n)] \). We can also express \( \alpha \) as

\[ \alpha = [\alpha^1, \ldots, \alpha^{(c-1)}] = \begin{bmatrix} \alpha_{11} & \ldots & \alpha_{1(c-1)} \\ \vdots & \vdots & \vdots \\ \alpha_{n1} & \ldots & \alpha_{n(c-1)} \end{bmatrix} \in \mathbb{R}^{n \times (c-1)}. \quad (5.9) \]
Here, we multiply both sides of the eigenvalue equation in Eq. (5.6) by $Q^T$. Then, Eq. (5.6) becomes

$$Q^T S_b^\phi W = Q^T S_t^\phi W \tilde{\Lambda}. \quad (5.10)$$

Replacing $S_b^\phi$ and $S_t^\phi$ with $M_b M_b^T$ and $M_t M_t^T$, we obtain

$$Q^T M_b M_b^T Q\alpha = Q^T M_t M_t^T Q\alpha \tilde{\Lambda}. \quad (5.11)$$

Let $\tilde{M}_b$ and $\tilde{M}_t$ denote $\tilde{M}_b = Q^T M_b$ and $\tilde{M}_t = Q^T M_t$, respectively. Then, $\tilde{M}_b$ and $\tilde{M}_t$ can be expressed with the kernel function instead of the mapping function:

$$\tilde{M}_b = Q^T M_b \in R^{n \times c} \quad (5.12)$$

$$= \begin{bmatrix} 
\Phi(x_1)^T \\
r \vdots \\
\Phi(x_n)^T 
\end{bmatrix} = \sqrt{p_1} \Phi_1^b \ldots \sqrt{p_c} \Phi_c^b$$

$$\tilde{M}_t = Q^T M_t \in R^{n \times n} \quad (5.13)$$

$$= \begin{bmatrix} 
\sqrt{p_1} K_{b1}^{11} \ldots \sqrt{p_c} K_{b1}^{c1} \\
r \vdots \\
\sqrt{p_1} K_{bn}^{1n} \ldots \sqrt{p_c} K_{bn}^{cn} 
\end{bmatrix}$$

where $\Phi_t^i = \frac{1}{m} \sum_{j \in N_t} \Phi(x_j) - \frac{1}{n} \sum_{i=1}^{n} \Phi(x_i)$ and $K_{b}^{ip} = \frac{1}{m} \sum_{j \in N_t} k(x_j, x_p) - \frac{1}{n} \sum_{i=1}^{n} k(x_i, x_p)$.
where $\Phi_l = \Phi(x_l) - \frac{1}{n} \sum_{i=1}^{n} \Phi(x_i)$ and $K_{lp}^{\text{np}} = k(x_l, x_p) - \frac{1}{n} \sum_{i=1}^{n} k(x_i, x_p)$.

Substituting $\tilde{\mathbf{M}}_b$ and $\tilde{\mathbf{M}}_t$ into Eq. (5.11), we have

$$
\tilde{\mathbf{M}}_b \tilde{\mathbf{M}}_b^T \alpha = \tilde{\mathbf{M}}_t \tilde{\mathbf{M}}_t^T \alpha \Lambda. 
$$

(5.14)

Let $\mathbf{H}_b$ and $\mathbf{H}_t$ denote $\mathbf{H}_b = \tilde{\mathbf{M}}_b \tilde{\mathbf{M}}_b^T$ and $\mathbf{H}_t = \tilde{\mathbf{M}}_t \tilde{\mathbf{M}}_t^T$, respectively. Therefore, Eq. (5.14) becomes

$$
\mathbf{H}_b \alpha = \mathbf{H}_t \alpha \Lambda 
$$

(5.15)

where $\alpha$ comes to the eigenvector of $\mathbf{H}_t^{-1} \mathbf{H}_b$ in the newly defined eigenvalue equation. However, since $\mathbf{H}_t$ is singular, we cannot directly calculate the inverse matrix of it. Instead, we use simply the pseudoinverse. Let $\mathbf{H}_t^+$ denote the pseudoinverse of $\mathbf{H}_t$. To obtain $\alpha$, we should calculate $\mathbf{H}_t^+$ and the eigenvector of $\mathbf{H}_t^+ \mathbf{H}_b$. Both $n \times n$ matrix calculations are required. Here, to save running time and memory required to calculate $\alpha$, we develop an efficient algorithm.

In deed, $\mathbf{H}_t$ and $\mathbf{H}_b$ correspond to the total scatter matrix and between-scatter matrix in the $n$ dimensional space where the data set is represented as follows:

$$
\begin{bmatrix}
  k(x_1, x_1) & \ldots & k(x_1, x_n) \\
  \vdots & \ddots & \vdots \\
  k(x_n, x_1) & \ldots & k(x_n, x_n)
\end{bmatrix}.
$$

(5.16)

That is, each column in the data matrix can be viewed as a data point. Now we can deal with the data set in a very reduced data space, that is, the dimensionality is reduced from $m$ to $n$ ($m >> n$). By singular value decomposition (SVD), $\tilde{\mathbf{M}}_t$ can be expressed as $\tilde{\mathbf{M}}_t = \mathbf{U} \hat{\Lambda} \frac{1}{2} \mathbf{V}^T$ where $\mathbf{U}$ and $\mathbf{V}$ are orthonormal column matrices and $\Lambda$ is a diagonal matrix with positive or zero elements in decreasing order. Then, $\mathbf{H}_t = \tilde{\mathbf{M}}_t \tilde{\mathbf{M}}_t^T = \mathbf{U} \hat{\Lambda} \frac{1}{2} \mathbf{V}^T (\mathbf{U} \hat{\Lambda} \frac{1}{2} \mathbf{V}^T)^T = \mathbf{U} \hat{\Lambda} \frac{1}{2} \mathbf{V}^T \mathbf{V} \hat{\Lambda} \frac{1}{2} \mathbf{U}^T = \mathbf{U} \hat{\Lambda} \mathbf{U}^T$. That is, $\mathbf{U}$ and $\hat{\Lambda}$ are the eigenvector and the eigenvalue of $\mathbf{H}_t$, respectively.
Definition (Moore – Penrose Inverse) Given a matrix \( A \), a matrix which satisfies the following four equations is called Moore – Penrose inverse of \( A \).

1. \((AA^+)^T = AA^+\)
2. \((A^+A)^T = A^+A\)
3. \(AA^+A = A\)
4. \(A^+AA^+ = A^+\)

Assume that the rank of \( H_t \) is \( r \). Hence, \( H_t = \sum_{i=1}^{r} \lambda_i u_i u_i^T \). Here, the Moore-Penrose inverse of \( H_t \) can be expressed as \( H_t^+ = \sum_{i=1}^{r} \frac{1}{\lambda_i} u_i u_i^T = U\tilde{\Lambda}^+ U^T \). Since \( H_t \) is positive semidefinite, \( H_t^{\frac{1}{2}} = U\tilde{\Lambda}^{\frac{1}{2}} U^T \). Therefore, the Moore-Penrose inverse of \( H_t^{\frac{1}{2}} \) is \( H_t^{+\frac{1}{2}} = U\tilde{\Lambda}^{+\frac{1}{2}} U^T \).

Corollary Covariance matrices are always positive semidefinite. If a matrix \( A \) is positive semidefinite and \( A = U\Lambda U^T \) where \( U \) and \( \Lambda \) are the eigenvector and eigenvalue of \( A \), then the following equation holds:

\[ A^{b/a} = U\Lambda^{b/a} U^T \text{ where } b/a > 0. \]

Suppose that \( S_b \) and \( S_t \) are the between scatter matrix and total scatter matrix in the input space. Then, we use \( S_b = S_b S_t^+ S_t \) which was proven in [77]. Likewise, since \( H_b \) and \( H_t \) are the between scatter matrix and total scatter matrix in the \( n \) dimensional space, the following equation holds:

\[ H_b = H_b H_t^+ H_t. \quad (5.17) \]

Similarly, \( H_b = H_t H_t^+ H_b \). Then, Eq. (5.15) becomes

\[ H_b H_t^+ H_t \alpha = H_t \alpha \tilde{\Lambda}. \quad (5.18) \]

Let \( K \) denote \( K = H_t \alpha \). Therefore, \( \alpha \) is

\[ \alpha = H_t^+ K = H_t^{+\frac{1}{2}} (H_t^{+\frac{1}{2}} K) \quad (5.19) \]
because of

\[
J(H_t^+ K) = J(H_t^+ H_t \alpha) \tag{5.20}
\]

\[
= \text{tr}((\alpha^T H_t^+ H_t \alpha)^{-1}(\alpha^T H_t^+ H_t \alpha))
\]

\[
= \text{tr}((\alpha^T H_t^+ H_t \alpha)^{-1}(\alpha^T H_t^+ H_t \alpha))
\]

\[
= \text{tr}((\alpha^T H_t \alpha)^{-1}(\alpha^T H_t \alpha)) = J(\alpha)
\]

and \(H_t^+ H_t = H_t\) by Moore-Penrose inverse.

From Eq. (5.18), we have

\[
H_b H_t^+ K = K \tilde{\Lambda}. \tag{5.21}
\]

Multiplying both sides of Eq. (5.21) by \(H_t^{+\frac{1}{2}}\), we have

\[
H_t^{+\frac{1}{2}} H_b H_t^{+\frac{1}{2}} (H_t^{+\frac{1}{2}} K) = (H_t^{+\frac{1}{2}} K) \tilde{\Lambda}. \tag{5.22}
\]

We know \(H_t^{+\frac{1}{2}} H_b H_t^{+\frac{1}{2}} = (H_t^{+\frac{1}{2}} \tilde{M}_b)(H_t^{+\frac{1}{2}} \tilde{M}_b)^T\) where \(H_t^{+\frac{1}{2}}\) is a symmetric matrix. Thus, \(H_t^{+\frac{1}{2}} K\) can be obtained by solving singular value decomposition (SVD) of \(H_t^{+\frac{1}{2}} \tilde{M}_b \in R^{n \times c}\).

If we express a matrix \(A\) as \(A = U \Lambda^{1/2} V^T\) via SVD, where \(U\) and \(\Lambda\) are orthonormal column matrices and \(\Lambda^{1/2}\) is a diagonal matrix with positive or zero elements in decreasing order, then \(A A^T = U \Lambda^{1/2} V^T (U \Lambda^{1/2} V^T)^T = U \Lambda U^T\). That is, \(U\) and \(\Lambda\) are the eigenvector and eigenvalue of \(A A^T\). Therefore, by SVD of \(A\) we can obtain the eigenvector of \(A A^T\).

Now we can calculate \(\alpha\) in Eq. (5.19) because we know \(H_t^{+\frac{1}{2}} = U \tilde{\Lambda}^{+\frac{1}{2}} U^T\) and \(H_t^{+\frac{1}{2}} \tilde{M}_b\). Though \(H_t^{+\frac{1}{2}}\) requires \(n \times n\) matrix calculation, the proposed method can save the running time and the usage of memory to calculate \(\alpha\) because \(H_t^{+\frac{1}{2}} \tilde{M}_b\) is \(n \times c\) matrix calculation \((n >> c)\). Given a sample \(x \in R^{l \times 1}\) and its mapped data \(\Phi(x)\), the discriminant feature vector \(y\) is obtained by the following transformation

\[
y = W^T \Phi(x) = (Q \alpha)^T \Phi(x) = \alpha^T Q^T \Phi(x) \tag{5.23}
\]
\[ \alpha^T [k(x_1, x), \ldots, k(x_n, x)]^T \in R^{(c-1) \times 1}. \]

For a test sample \( x \), the predicted class is
\[ C(x) = \arg\min_k \sum_{i=1}^{c-1} (W_i^T (\Phi(x) - m_k^\Phi))^2. \] (5.24)

After mapping the mean vector of each class in the feature space and \( \Phi(x) \) into the discriminant space, a class with the smallest distance between the mapped mean vector and the mapped \( \Phi(x) \) in the discriminant space is chosen.

5.4 Experiments

5.4.1 The Dataset

The raw mass spectral data for Alzheimer’s disease was downloaded from http://www.perkinelmer.com/biomarkerdata. The dataset consists of 276 samples, mild cognitive impairment (MCI:32), Alzheimer disease (AD:58) and normal (186), each processed in triplicate. In this study, one out of 3 separate sets was selected at random.

5.4.2 Classification Algorithms

FKDA was implemented in MATLAB 6.5. The performance of our method was compared with other algorithms, LDA, SVM, Random Forest, J48 and \( k \)NN with \( k = 1 \). In each classification algorithm, 10 cross validation (CV) was applied. That is, samples of each class are partitioned randomly into 10 folds consisting of 9 folds for training and one fold for testing. During 10-CV, all folds are used for testing. The 10-CV was repeated 30 times and the resulting measurements were averaged.
Table 5.1. The mean and standard deviation (in parenthesis) of accuracies in Alzheimer disease dataset. The dimensionality after mapping in FKDA and GLDA is $c - 1$ regardless of the number of features used.

<table>
<thead>
<tr>
<th>No. of features</th>
<th>Methods (Gaussian)</th>
<th>Methods (Polynomial)</th>
<th>GLDA</th>
<th>kNN ($k=1$)</th>
<th>RF</th>
<th>J48</th>
<th>SVM</th>
</tr>
</thead>
<tbody>
<tr>
<td>500</td>
<td>82.07(0.92)</td>
<td>80.47(0.69)</td>
<td>74.84(1.89)</td>
<td>66.16(0.95)</td>
<td>73.80(2.18)</td>
<td>65.07(1.78)</td>
<td>75.07(1.29)</td>
</tr>
<tr>
<td>1000</td>
<td>86.01(0.74)</td>
<td>82.86(0.78)</td>
<td>82.30(0.54)</td>
<td>66.63(0.73)</td>
<td>73.22(1.14)</td>
<td>62.25(2.35)</td>
<td>75.14(0.86)</td>
</tr>
<tr>
<td>1500</td>
<td>86.83(0.70)</td>
<td>84.64(0.49)</td>
<td>84.73(0.68)</td>
<td>68.48(0.64)</td>
<td>72.83(1.70)</td>
<td>61.70(1.96)</td>
<td>73.01(1.03)</td>
</tr>
<tr>
<td>2000</td>
<td>87.07(0.50)</td>
<td>84.31(0.48)</td>
<td>84.89(0.62)</td>
<td>67.21(0.75)</td>
<td>72.72(0.71)</td>
<td>61.34(2.35)</td>
<td>73.15(1.10)</td>
</tr>
<tr>
<td>2500</td>
<td>85.74(0.63)</td>
<td>84.31(0.30)</td>
<td>85.18(0.57)</td>
<td>68.19(1.10)</td>
<td>71.96(1.52)</td>
<td>61.30(1.36)</td>
<td>72.61(0.95)</td>
</tr>
<tr>
<td>3000</td>
<td>85.83(0.64)</td>
<td>83.41(0.56)</td>
<td>84.35(0.64)</td>
<td>65.76(0.97)</td>
<td>71.63(1.38)</td>
<td>60.14(2.16)</td>
<td>75.22(0.86)</td>
</tr>
<tr>
<td>3500</td>
<td>85.80(0.57)</td>
<td>83.19(0.60)</td>
<td>84.00(0.71)</td>
<td>65.22(1.29)</td>
<td>71.63(1.82)</td>
<td>59.71(2.75)</td>
<td>73.66(0.65)</td>
</tr>
<tr>
<td>4000</td>
<td>84.69(0.52)</td>
<td>82.93(0.36)</td>
<td>83.66(0.48)</td>
<td>65.04(1.19)</td>
<td>70.14(1.72)</td>
<td>61.81(2.14)</td>
<td>71.96(1.27)</td>
</tr>
<tr>
<td>4500</td>
<td>84.29(0.44)</td>
<td>82.64(0.65)</td>
<td>83.22(0.53)</td>
<td>66.74(1.04)</td>
<td>71.20(1.65)</td>
<td>61.34(1.74)</td>
<td>72.79(0.75)</td>
</tr>
<tr>
<td>5000</td>
<td>83.35(0.68)</td>
<td>81.85(1.00)</td>
<td>82.64(0.55)</td>
<td>66.52(0.82)</td>
<td>69.38(1.65)</td>
<td>60.51(2.79)</td>
<td>70.00(0.96)</td>
</tr>
</tbody>
</table>

5.4.3 Feature Reduction

To reduce computational burden caused by using peaks in all the bins, we ranked peaks by a feature ranking method based on the ratio of between-group to within-group sums of squares and select the tractable size of features in our algorithm. By BW ratio, some of them were used in the experiments. The feature ranking method was proposed by Dudoit et al. for feature selection for multi-class problems [49]. For a peak $j$, the ratio is

$$\text{BW}(j) = \frac{\sum_i \sum_k I(y_i = k)(\bar{x}_{kj} - \bar{x}_{j})^2}{\sum_i \sum_k I(y_i = k)(\bar{x}_{ij} - \bar{x}_{kj})^2}$$  (5.25)

where $I(\cdot)$ is the indicator function. $\bar{x}_{j}$ and $\bar{x}_{kj}$ denote the average intensity of peak $j$ across all samples and across samples belong to class $k$. As the ratio for a peak is large, the peak is more likely relevant to class separation.

5.4.4 Experimental Results

Based on BW ratio, we extracted the top 500, 1000, and so forth up to the top 5000 peaks in all spectra. Note that no matter what the number of peaks is, the
Figure 5.2. Error rates in test samples of each class. In labels, prefix FG, FP and G indicate FKDA with Gaussian kernel, FKDA with the second-order polynomial kernel and GLDA, respectively.

dimensionality after dimension reduction in FKDA and GLDA is $c - 1$. With each selected feature subset, all classification algorithms mentioned in section 5.4.2 were carried out. For FKDA, two popular kernels were employed. One is Gaussian kernel $k(x, y) = \exp(-||x - y||^2/\delta)$ and the other is the second-order polynomial kernel $k(x, y) = (x \cdot y + 1)^2$. Table 5.1 shows the averaged accuracies for all algorithms. As can be seen from the table, when the Gaussian kernel in FKDA was used, the best performance was achieved for all feature subsets. However, GLDA was superior to the polynomial kernel in FKDA. It should be noted that discriminant analysis methods (KFDA and GLDA) obtained the better performance compared to others.

Table 5.1 demonstrates overall accuracies for all the test sets. However, we are also interested in the accuracy for test samples of each class. In all classification methods, the accuracy rate to correctly classify MCI samples is low and the accuracy
rate for normal samples is high. It can be attributed to the fact that the distribution of samples used in this study is rather imbalanced. For example, the number of normal samples (186) is about 6 times as larger as MCI (32). In such a case, traditional classification algorithms are biased to the majority classes. According to the experimental results in this study, FKDA and GLDA are relatively strong in the situation (not shown in this paper). Fig. 5.2 illustrates the error rate for FKDA and GLDA in test samples of each class. FKDA with Gaussian kernel has the smallest error rate. We observed that when the number of features increases, the error rate in MCI also increases while the error rate in normal decreases. It is implied that if all features are used, the accuracy in MCI will dramatically get worse (not shown in this paper).

We compared the elapsed time to calculate $\alpha$ from FKDA and from the eigen-decomposition of $H^\dagger H_b$ in Eq. (5.15). For test, values were generated uniformly between 0 and 1. Fig. 5.3 depicts the average running time after 10 runs to obtain $\alpha$ value when with 1000 fixed features the number of samples increases from 50 to 1000 by 50. As the number of samples increases, the elapsed time gap between two graphs grows larger. Since in our study the calculation of $\alpha$ is performed many times, after entire experiments the difference in the total accumulating running time will be considerably increased.

5.5 Conclusion

We proposed a fast kernel discriminant analysis technique, FKDA which was designed to aim to overcome some limitations of classical LDA: singularity and linearity. In experiments with mass spectral data for Alzheimer’s disease, FKDA with Gaussian kernel outperformed other classification algorithms. Also, FKDA is faster in the calculation of the optimal discriminant vectors and strong in the imbalanced
Figure 5.3. Averaged elapsed time in a run to calculate $\alpha$. The label ‘Fast’ indicates our FKDA method and ‘Slow’ represents the method through the eigen-decomposition of $H_t^+H_b$ in Eq. (5.15).

dataset. Since diseases progress in multistage, the accurate diagnosis of the current stage is very important for treatment. It requires the fast and precise multi-class classification algorithm. FKDA can be applicable to such multi-class classification problems with nonlinearly structured data.
REFERENCES


BIOGRAPHICAL STATEMENT

Jung Hun Oh received his BS and MS degrees in Computer Science from Soongsil University, South Korea, in 1995 and 1997, respectively, and the PhD degree in Department of Computer Science and Engineering at the University of Texas at Arlington in 2008. His research interests include bioinformatics, machine learning, and data mining. His recent work has focused on biomarker selection, multi-class classification, and quantitative proteomics.