Machine Learning Methods for High-Dimensional Imbalanced Biomedical Data

by

Tao Yang

A Thesis Presented in Partial Fulfillment
of the Requirements for the Degree
Master of Science

Approved July 2013 by the
Graduate Supervisory Committee:

Jieping Ye, Chair
Yalin Wang
Hasan Davulcu

ARIZONA STATE UNIVERSITY

August 2013

ABSTRACT

Learning from high dimensional biomedical data attracts lots of attention recently. High dimensional biomedical data often suffer from the curse of dimensionality and have imbalanced class distributions. Both of these features of biomedical data, high dimensionality and imbalanced class distributions, are challenging for traditional machine learning methods and may affect the model performance. In this thesis, I focus on developing learning methods for the high-dimensional imbalanced biomedical data. In the first part, a sparse canonical correlation analysis (CCA) method is presented. The penalty terms is used to control the sparsity of the projection matrices of CCA. The sparse CCA method is then applied to find patterns among biomedical data sets and labels, or to find patterns among different data sources. In the second part, I discuss several learning problems for imbalanced biomedical data. Note that traditional learning systems are often biased when the biomedical data are imbalanced. Therefore, traditional evaluations such as accuracy may be inappropriate for such cases. I then discuss several alternative evaluation criteria to evaluate the learning performance. For imbalanced binary classification problems, I use the undersampling based classifiers ensemble (UEM) strategy to obtain accurate models for both classes of samples. A small sphere and large margin (SSLM) approach is also presented to detect rare abnormal samples from a large number of subjects. In addition, I apply multiple feature selection and clustering methods to deal with high-dimensional data and data with highly correlated features. Experiments on high-dimensional imbalanced biomedical data are presented which illustrate the effectiveness and efficiency of my methods.

DEDICATION

This thesis is dedicated to my parents

for their love, encouragement

and endless support.

ACKNOWLEDGEMENTS

I would like to thank many people who supported and helped me during my research and graduate studies. The completion of this thesis is impossible without them.

My warmest thanks to my advisor, Dr. Jieping Ye, for his excellent guidance, support and encouragement during my research and studies. I would also like to thank Dr. Yalin Wang and Dr. Hasan Davulcu, for serving on my thesis committee. Thanks for their valuable insight and guidance.

Thanks to Yashu Liu for his help in my research and helpful suggestions to my thesis. Thanks also to Dr. Chao Zhang and Dr. Binbin Lin for their review of my thesis and their valuable advice.

I am also thankful to other fellow members in the Center for Evolutionary Medicine and Informatics (CEMI) of the Biodesign Institute at Arizona State University, for their help, friendship and support: Dr. Zheng Wang, Dr. Jie Wang, Dr. Pinghua Gong, Dr. Lei Yuan, Dr. Rita Chattopadhyay, Jiayu Zhou, Sen Yang, Shuo Xiang, Qian Sun, Zhi Nie, Cheng Pan, Rashmi Dubey and Lei Zhang.

My sincere thanks go to Dr. Lenore Dai and Dai Dai, for their help and support during my study at Arizona State University.

Finally, I would like to express my deepest gratitude to my beloved parents, for their enduring love, support and encouragement throughout my life. This thesis is dedicated to them.

TABLE OF CONTENTS

LIST OF TABLES

Table Page

LIST OF FIGURES

Chapter 1

Background and Introduction

1.1 Background and Challenges

In recent decades, machine learning techniques have been extensively applied to solve problems in computational biology and bioinformatics. For example, some learning models have been developed to distinguish the patients from healthy controls based on biomedical data. Also, there are several works on investigating the patterns, the mechanisms and the interactions among biological molecules [1, 2, 3].

Recently, there have been many interests in the learning problems of biomedical data. Given a raw biomedical data, it is generally difficult to automatically identity interesting patterns contained in the data. Moreover, it will become much more difficult if there are multiple data sets available, or multiple clinical subtypes need to be recognized. For example, in the Depression research (see Chapter 2 for details), we have six data sets on a set of samples, and we can also define more than ten clinical subtypes (labels) based on the data. The aforementioned facts imply that the learning tasks can be formalized for a variety of research targets. Therefore, it is excepted to determine a certain research target before applying further machine learning methods to learn inherent relationships and patterns from these biomedical data sets with multiple labels.

It is noteworthy that the biomedical data are often imbalanced, that is, the number of patients is often much smaller than the number of available healthy controls. However, a large number of traditional learning systems are designed under the assumption that the data have balanced class distributions. Thus, these classical methods are often biased when the biomedical data are imbalanced. For instance, in binary classification tasks, standard accuracy-based classifiers will be dominated by the majority class of observations [4]. Meanwhile, the characteristics of minority-class examples cannot be well captured. In practice, it is expected that the obtained models should perform well on

both of the majority set and the minority set. The principle reason behind this expectation is that the minority set of samples in the biomedical data may be very significant. For example, in the Depression research, we would like to find an accurate model for the information of patients, while the patients just belong to the so-called minority set.

In addition, the biomedical data often suffer from the curse of dimensionality [5]. The curse of dimensionality refers to the scenario that the number of features p, is much greater than the number of subjects n, *i.e.,* $p \gg n$. In some cases, the data also contain many redundant features. Such high-dimensional data sets often appear in biology, *e.g.,* microarray data and protein data. Although there are many attempts to use traditional machine learning methods to deal with the biomedical data, most of these methods are built under the assumption that the data set has a relatively low dimensionality. However, this assumption is not often valid in practice. To learn from a high-dimensional biomedical data set, a desired learning model should satisfy the following three requirements:

- Overcome the curse of dimensionality;
- Perform well for the data with highly-correlated variables;[1]
- Be effective and efficient for a variety of high-dimensional applications.

To sum up, to learn from a biomedical data set, we need to determine an appropriate research target. Moreover, the biomedical data often suffer from the high-dimensionality and have imbalanced class distributions. Both of these characteristics bring severe challenges for traditional machine learning methods, and may affect the model performance.

[1] In contrast, Lasso, for example, cannot handle correlation-structure among the features [6].

1.2 Problems Setup

In this thesis, I focus on developing learning methods for the high-dimensional imbalanced biomedical data. There are essentially three objectives in this thesis:

- Select the potential patterns among the data and class labels for a given biomedical data set with multiple class labels;
- Develop accurate classifiers that can effectively and efficiently identify patients from healthy controls;
- Find significant biomarkers from the biomedical data set.

1.3 Related Work

In this section, we review several works related to the learning process for high-dimensional imbalanced biomedical data.

There are several machine learning methods that can be used to learn the potential patterns among multiple biomedical data sets and these labels, *e.g.,* Canonical correlation analysis (CCA). CCA is a widely used linear method to investigate the relationship between two sets of multidimensional variables [7]. To deal with high-dimensional data, sparsity has been introduced into the CCA formulation, *e.g.,* the sparse CCA via linear regression [8, 9, 10], the sparse CCA via iterative greedy algorithm [11, 12, 13, 14, 15, 16] and the sparse CCA via Bayesian learning [17, 18, 19].

After determining the appropriate research targets (patterns), we focus on addressing two challenges in learning from the biomedical data: the imbalanced class distributions and the curse of dimensionality.

To deal with the imbalanced class distributions or, more specifically, deal with imbalanced binary classification problems, an intuitive idea is to balance the training set.

Recently, many studies suggest sampling methods are effective. There are various sampling methods that have been proposed, *e.g.,* random undersampling and oversampling [20, 21], informed undersampling [22], synthetic minority oversampling technique (SMOTE) [23], sampling with data cleaning techniques [4, 24] and the cluster-based oversampling (CBO) [25]. In addition to sampling strategies, the cost-sensitive framework is proposed in the imbalanced learning by using multiple cost matrices that generate the costs for misclassifying any abnormal subjects [26, 27]. There are also other effective methods for the imbalanced learning such as Kernel-based methods [28] and active learning methods [29].

Moreover, there are many strategies proposed to solve the high-dimensional data problems, *e.g.,* feature selection and feature extraction techniques [30, 31], clustering methods [6], and sparse approaches [8]. Essentially, the aim of high-dimensional learning is to reduce the dimensionality as well as to keep the distinguishable features.

1.4 Methods and Approaches

In this thesis, I focus on developing learning methods for high-dimensional imbalanced biomedical data. I first consider a sparse canonical correlation analysis method that uses the penalty terms to control the sparsity of the projection matrices. This method is then applied to find patterns among data sets and outcomes, or to find patterns among different data sources. To deal with the biomedical data with imbalanced class distributions, I present several evaluation criteria to evaluate the learning performance. In order to build accurate models for both classes of samples, I consider the undersampling method on the training set. Meanwhile, to ensure the robustness of the learning models, a further approach that combines the undersampling method and the ensemble strategy is discussed. Moreover, a small sphere and large margin approach is also presented, which can be used to detect rare abnormal samples from a large number of subjects. The rest parts of the thesis introduce several feature selection methods and clustering methods. Both of these

methods are aiming to improve the learning performance of the high-dimensional biomedical data. Feature selection methods can further produce significant features, and the clustering methods can help us reduce the redundancy in the data.

1.5 Thesis Organization

The thesis is organized as follows. Chapter 2 introduces the sparse canonical correlation analysis (CCA) method and its applications. In Chapter 3, I discuss some learning problems of imbalanced data, including the choice of appropriate evaluation criteria, the undersampling-based classifiers ensemble (UEM) method, feature selection methods to the UEM framework, and the novelty detection ideas. I present the method of clustering highly correlated variables in Chapter 4 and conclude the thesis in Chapter 5.

Chapter 2

Sparse Canonical Correlation Analysis

Canonical correlation analysis (CCA), first proposed by H. Hotelling in 1936 [7], is a classical method for measuring the linear relationship between two sets of multidimensional variables. In this chapter, I first introduce the linear correlation coefficient and CCA, and then introduce the sparse CCA method for high-dimensional data. Finally I discuss some applications of sparse CCA to biomedical data.

2.1 Linear Correlation and Canonical Correlation Analysis

Given two vectors $\mathbf{x}$ and $\mathbf{y}$, $\mathbf{x}, \mathbf{y} \in \mathbb{R}^n$, the linear correlation coefficient (or called *Pearson product-moment correlation coefficient* [32]), which measures the strength of linear dependence between these two variables, is defined as the covariance of the two vectors divided by the product of their standard deviations:

$$\rho_{\mathbf{x},\mathbf{y}} = \frac{cov(\mathbf{x},\mathbf{y})}{\sigma_{\mathbf{x}}\sigma_{\mathbf{y}}} \tag{2.1}$$

$$= \frac{\mathbf{x}^T\mathbf{y}}{\|\mathbf{x}\|_2\|\mathbf{y}\|_2}, \tag{2.2}$$

where both of the variables are standardized as mean zero, *i.e.,* $\bar{x} = \bar{y} = 0$.

By extending to the situation of multidimensional variables, we use the canonical correlation analysis to evaluate the relationship between multiple variables. It has been observed that the correlation analysis is sensitive to the coordinates. Although the two variables have a strong linear relationship, their correlation may not be well-expressed due to the inappropriate choice of coordinates [33]. Thus we use the CCA method to find a set of optimal basis vectors that maximize the correlation between the base-projections of the variables.

Consider two data matrices $\mathbf{X} = [\mathbf{x}_1, \ldots, \mathbf{x}_n] \in \mathbb{R}^{p\times n}$ and $\mathbf{Y} = [\mathbf{y}_1, \ldots, \mathbf{y}_n] \in \mathbb{R}^{q\times n}$, where $\mathbf{x}_i$, and $\mathbf{y}_i (1 \leq i \leq n)$ correspond to two different views of the same sample source,

respectively. Assume that each row (feature) in $\mathbf{X}$ and $\mathbf{Y}$ is centered, *i.e.,* $\sum_{i=1}^{n} x_{ji} = 0$ for any $1 \leq j \leq p$ and $\sum_{i=1}^{n} y_{ki} = 0$ for an $1 \leq k \leq q$. Let $\mathbf{w}_x \in \mathbb{R}^p$ and $\mathbf{w}_y \in \mathbb{R}^q$ be two transformation vectors for each variable, and then $\mathbf{w}_x^T\mathbf{X}$ and $\mathbf{w}_y^T\mathbf{Y}$ denote the projections of two variables in the new coordinate systems. CCA gives an optimal pair of $\mathbf{w}_x$ and $\mathbf{w}_y$ that maximizes the correlation between the two projections, that is,

$$\rho = \max_{\mathbf{w}_x, \mathbf{w}_y} corr\ (\mathbf{w}_x^T\mathbf{X}, \mathbf{w}_y^T\mathbf{Y}) \tag{2.3}$$

$$= \max_{\mathbf{w}_x, \mathbf{w}_y} \frac{\mathbf{w}_x^T\mathbf{X}\mathbf{Y}^T\mathbf{w}_y}{\|\mathbf{w}_x^T\mathbf{X}\|_2 \|\mathbf{w}_y^T\mathbf{Y}\|_2}. \tag{2.4}$$

Note that $\mathbf{w}_x$ and $\mathbf{w}_y$ refer to the canonical vectors (or weights), and $(\mathbf{w}_x^T\mathbf{X}, \mathbf{w}_y^T\mathbf{Y})$ is termed as the pair of canonical variables [34].

2.2 Sparse CCA Method

In recent years, CCA has been widely used in various applications, *e.g.,* learning semantic representations for web images [33]; obtaining multiple-assays measurements (gene expression, DNA copy number, *etc.*) of samples taken from one single set of patients [11]. However, as pointed out in [11], traditional CCA method may not be suitable to the high-dimensional situation, where the feature dimension (number of features) is much larger than the number of observations.

To circumvent such problem, sparse method has been introduced to extend the canonical correlation analysis. The main idea of sparse CCA is to maximize the correlation coefficient between two projected variables, such that the projections are achieved in the reduced subspaces of the original feature spaces, *i.e.,* only a small set of variables will be selected in each projection.

Various sparse CCA methods have been proposed recently, *e.g.,* the sparse CCA via linear regression [8, 9, 10], the sparse CCA via iterative greedy algorithm [11, 12, 13, 14, 15, 16] and the sparse CCA via Bayesian learning [17, 18, 19].

In this thesis, I process the sparse CCA method with a penalty strategy (*i.e.,* the so-called *penalized* CCA proposed by Witten *et al.* (2009) [11, 15]). Consider two data sets $\mathbf{X}$ and $\mathbf{Y}$ of n observations with dimension p and q respectively, that is, $\mathbf{X} \in \mathbb{R}^{n \times p}$ and $\mathbf{Y} \in \mathbb{R}^{n \times q}$. Each column of $\mathbf{X}$ and $\mathbf{Y}$ are centered and scaled to have mean zero and standard deviation one. Denote $\mathbf{w}_x \in \mathbb{R}^p$ and $\mathbf{w}_y \in \mathbb{R}^q$ as two projection (transformation) matrices for $\mathbf{X}$ and $\mathbf{Y}$ respectively. Then, the sparse CCA can be formulated as:

$$\begin{aligned} &\max_{\mathbf{w}_x, \mathbf{w}_y} \mathbf{w}_x^T \mathbf{X}^T \mathbf{Y} \mathbf{w}_y \\ &\text{subject to } \mathbf{w}_x^T \mathbf{X}^T \mathbf{X} \mathbf{w}_x \leq 1, \mathbf{w}_y^T \mathbf{Y}^T \mathbf{Y} \mathbf{w}_y \leq 1, \\ &P_x(\mathbf{w}_x) \leq c_x, P_y(\mathbf{w}_y) \leq c_y, \end{aligned} \tag{2.5}$$

where $P_x(\cdot)$ and $P_y(\cdot)$ are the convex penalty functions, and c_x and c_y are both evaluated from bounded intervals w.r.t. the penalty functions. Note that the values of c_x and c_y would yield feasible solutions for the penalized CCA, even when $p, q \gg n$. A brief diagram of the sparse CCA method is shown in Fig. 2.1.

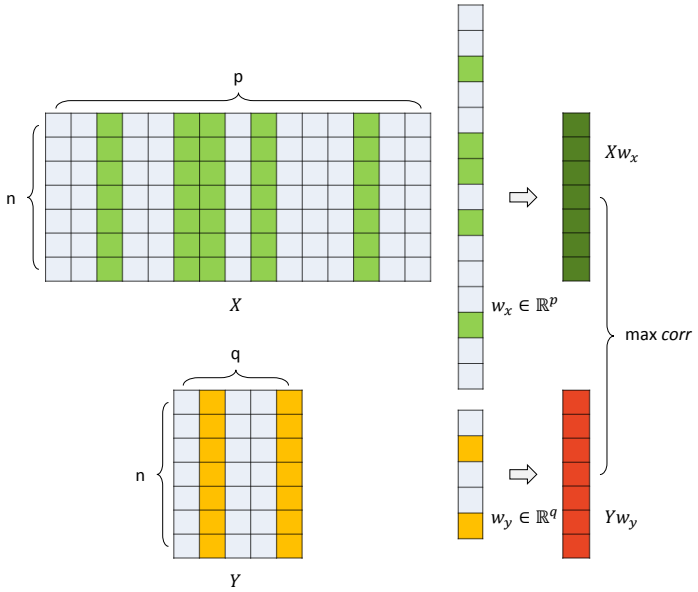


Figure 2.1: A diagram of the sparse canonical correlation analysis method, with only one pair of the canonical vectors.

Previous works indicated that, in high-dimensional situation, the assumption that the covariance matrix of features is diagonal can guarantee satisfactory results [35, 36]. Thus, $\mathbf{X}^T\mathbf{X}$ and $\mathbf{Y}^T\mathbf{Y}$ are replaced with the identity matrix $\mathbf{I}$, and then the sparse CCA criterion (2.5) can be simplified as:

$$\begin{aligned} &\max_{\mathbf{w}_x,\mathbf{w}_y} \mathbf{w}_x^T\mathbf{X}^T\mathbf{Y}\mathbf{w}_y \\ &\text{subject to } \|\mathbf{w}_x\|_2^2 \leq 1, \|\mathbf{w}_y\|_2^2 \leq 1, \\ &\qquad P_x(\mathbf{w}_x) \leq c_x, P_y(\mathbf{w}_y) \leq c_y. \end{aligned} \tag{2.6}$$

In general, the penalty function P_x (or P_y) has multiple forms, *e.g., lasso* and *fused lasso*. In this thesis, I focus on the lasso penalty, that is,

$$P_x(\mathbf{w}_x) = \|\mathbf{w}_x\|_1 = \sum_{i=1}^{n} |w_{xi}|. \tag{2.7}$$

Meanwhile, in order to constrain $\mathbf{w}_x$ to be sparse, the range of c_x that restricts the penalty function P_x should satisfy $1 \leq c_x \leq \sqrt{p}$ accordingly.

To solve the aforementioned problem, an iterative greedy algorithm was proposed by Witten [11]. At each iteration, one of $\mathbf{w}_x$ or $\mathbf{w}_y$ is fixed and the criterion (2.6) will be convex in $\mathbf{w}_y$ or $\mathbf{w}_x$. When the penalty functions P_x and P_y are L_1 (lasso) penalties, such iterative algorithm has a low computational cost and the detailed steps are described in Algorithm 1.

This penalized CCA method can be easily extended, for instance, to the multiple factors (components) CCA, to the sparse CCA with nonnegative weights. Some existing works also applied this work for analyzing multiple data sets, *i.e.,* the sparse multiple CCA method [11].

Algorithm 1 The sparse CCA Algorithm

Input: $\mathbf{X}, \mathbf{Y}, c_x, c_y$

Output: $\mathbf{w}_x, \mathbf{w}_y$

1: Initialize $\mathbf{w}_y$ to some initial values, *e.g.,* $\|\mathbf{w}_y\|_2 = 1$.
2: **while** not convergence **do**
3: $\quad \mathbf{w}_x \leftarrow \arg\max_{\mathbf{w}_x} \mathbf{w}_x^T \mathbf{X}^T \mathbf{Y} \mathbf{w}_y$ subject to $\|\mathbf{w}_x\|^2 \leq 1, \|\mathbf{w}_x\|_1 \leq c_x$.
4: $\quad \mathbf{w}_y \leftarrow \arg\max_{\mathbf{w}_y} \mathbf{w}_x^T \mathbf{X}^T \mathbf{Y} \mathbf{w}_y$ subject to $\|\mathbf{w}_y\|^2 \leq 1, \|\mathbf{w}_y\|_1 \leq c_y$.
5: **end while**

Each update for $\mathbf{w}_x$, takes the form

$$\mathbf{w}_x \leftarrow \frac{S(\mathbf{X}^T \mathbf{Y} \mathbf{w}_y, \triangle_1)}{\|S(\mathbf{X}^T \mathbf{Y} \mathbf{w}_y, \triangle_1)\|_2}, \tag{2.8}$$

where $\triangle_1 = 0$ is chosen so that $\mathbf{w}_x \leq c_x$, and $\triangle_1 > 0$ if $\mathbf{w}_x = c_x$. $S(\cdot)$ is the soft-thresholding operator, such that $S(x,a) = \text{sgn}(x)(|x| - a)_+$; $\mathbf{w}_y$ can also be obtained in the similar way.

2.3 Applications of sparse CCA to Biomedical Data

Next, I introduce the data sets used in the experiments and then present two instances of applications of sparse CCA: (1) detect patterns among data sets and labels, and (2) detect patterns among different data sources.

Data Description

Depression is a common mental disorder that affects about 350 million people worldly [37]. World Health Organization (WHO) characterizes the Major depressive disorder (MDD) (also known as clinical depression, major depression, *etc*) as "episodes of sadness, loss of interest or pleasure, feelings of guilt or low self-worth, disturbed sleep or appetite, feelings of tiredness and poor concentration".

It is believed that the integration of commonly studied indices of depression and molecular patient profiling offer the chance of better understanding the biomarkers of

Major Depression, and these biomarkers may be applied to develop and guide more efficient drug development and testing programmes [38].

(a) (b) (c) (d) (e)

Figure 2.2: Depression data set, (a) Personal and medical history, (b) Cognition, (c) Electrical brain-body function, (d) Brain structure, (e) Molecular profiles.

The Depression database contains five types of features (shown in Fig. 2.2), which were selected to reflect an integrative profile of information about: *a*) Personal medical history; *b*) Cognition; *c*) Electrical brain-body function (EBBF); *d*) Brain structure (sMRI, fMRI) and *e*) Molecular profiles.

Assessment paradigms *a*, *b*, *c*, and *e* are undertaken in the current studies. There are totally 275 samples collected from 249 individuals in the Depression data sets, and the major molecular profiles include Metabolite, Microarray, Protein and Transcripts profiles. In this thesis, some outliers are eliminated due to the inconsistent performance, the failure of quality controls, the long storage period, *etc*. A summary of sample statistics is shown in table 2.1.

Table 2.1: Depression data sets sample statistics and missing value imputation methods.

Data	Number of Samples	Dimensions	Missing Ratio	Imputation Methods
Cognitive	196	57	None	N.A.
EBBF	196	288	0.0835	EM / KNN / SVD
Metabolite	199	270	0.0127	EM / KNN / SVD
Microarray	228	54675	None	N.A.
Protein	206	41637	0.1965	halfMin / SVD
Transcripts	196	17502	None	N.A.

Table 2.1 indicates that the J&J-Depression data sets are involved in missing value problems. I thus applied multiple missing data imputation methods for these data, including:

- **Expectation maximization (EM) algorithm** [39]:
  Suppose the data are Gaussian distribution. The (regularized) EM algorithm is an iterative method based on ridge regression analysis that can estimate the mean values and covariance matrices from observations and impute the missing values.

- **Minimal / 2 (halfMin) algorithm** [40]:
  Assume that most of the missing values are too small to be detected. Thus, we impute all the missing values by half of the minimum value in the corresponding feature.

- **K-nearest neighbors (KNN) algorithm** [41, 42, 43]:
  Impute the missing values with a weighted mean of the k nearest-neighbor columns.

- **Singular value decomposition (SVD) algorithm** [42]:
  Employ low-rank SVD to approximate the whole data set, replace the missing parts and repeat the whole processing until convergence.

In the Depression research, some most commonly confounding effects likes age and gender, are already been considered. However, previous works pointed out that for the Metabolite data, the concentrations of a large number of metabolites are strongly affected by their storage time, since the plasma samples were stored at -20°. In order to reduce the storage time confounder, I applied two correction methods and the details are given in Appendix A.

Sparse CCA and Pattern Predetection

Classification is a typical machine learning task, which is aimed to categorize subjects into a fixed set of categories. Suppose we are given a set of data with multiple class labels. It is difficult, at first sight, to figure out what kind of data has prominent characteristics that can improve classification performance. Therefore, we discuss whether sparse CCA is an efficient tool to analyze inherent relations and patterns among the data sets and labels.

Detect Patterns among the Depression Data Sets

In the Depression research, one of the targets is to discriminate the depressive patients from healthy control subjects (HCs). There are more refined categorises of Depression (Dprs), *e.g.,* Anxiety Depression (AnxDprs), Melancholic Depression (MelanDprs) and Generalized Anxiety Disorder (GAD). Before further choosing the proper research targets, I use sparse CCA to predetect the patterns among the Depression data sets.

To apply the sparse CCA method, I test each data set from Table 2.1 with a set of labels. For example, we treat the Metabolite data and the set of labels as $\mathbf{X}$ and $\mathbf{Y}$ in formula (2.6) respectively. The choice of c_x is related to the sparsity of the Metabolite data, *i.e.,* the numbers of selected metabolites. At each time, we fix c_x and tune c_y and track the changes of labels that have been enclosed in each test. The canonical vectors $\mathbf{w}_x$ and $\mathbf{w}_y$ and the correlation coefficient between the projections of $\mathbf{X}$ and $\mathbf{Y}$ are recorded for each setting. I tested all six data sets with eleven class labels in total, and the experimental results are shown in Table 2.2 and Table 2.3.

Tables 2.2 & 2.3 summarize the results of the pattern predetection experiments for the Depression data sets. Each subtable in Tables 2.2 & 2.3 is the sparse CCA experiment results for a certain Depression data set and the class labels. The top part shows the settings of two constraints c_x and c_y, the middle part is the correlation between the two projections, and the bottom part records the canonical vector $\mathbf{w}_y$ for the class labels. The

values in a canonical vector correspond to the weights for each feature, which measures the significance of each variable. Recalling the projection process of $\mathbf{w}_y^T\mathbf{Y}$, a feature with a higher absolute weight means that this feature vector will contribute more to the

Table 2.2: The sparse CCA experiments between the Depression data sets and the class labels (part I): the weights of top five selected labels for each data set are recorded.

(a) Cognitive

c_x (Data)	0.50	0.50	0.50	0.50	0.50
c_y (Label)	0.30	0.35	0.40	0.50	0.55
Correlation	0.3422	0.3360	0.3325	0.3240	0.3195
Dprs		-0.177	-0.271	-0.472	-0.504
MelanDprs-C	**-1.000**	**-0.984**	**-0.958**	**-0.796**	**-0.732**
nMelanDprs-C					
MelanDprs-M					-0.005
nMelanDprs-M					
GADDprs			-0.098	-0.379	-0.433
GAD					
AnxDprs				-0.011	-0.150
Anx					
GAD(inDprs)					
Anx(inDprs)					

(b) EBBF

c_x (Data)	0.40	0.40	0.40	0.40	0.40
c_y (Label)	0.30	0.35	0.40	0.50	0.58
Correlation	0.4026	0.4044	0.3970	0.4084	0.4112
Dprs	**-1.000**	**-0.984**	**-0.918**	**-0.881**	**-0.749**
MelanDprs-C				-0.191	-0.394
nMelanDprs-C			-0.397		
MelanDprs-M				-0.206	-0.337
nMelanDprs-M					
GADDprs					-0.032
GAD					
AnxDprs		-0.177	-0.012	-0.380	-0.412
Anx					
GAD(inDprs)					
Anx(inDprs)					

(c) Metabolite

c_x (Data)	0.30	0.30	0.30	0.30	0.30
c_y (Label)	0.30	0.35	0.38	0.45	0.55
Correlation	0.4778	0.4789	0.4762	0.4674	0.4537
Dprs			-0.056	-0.160	-0.290
MelanDprs-C	**-1.000**	**-0.984**	**-0.971**	**-0.935**	**-0.848**
nMelanDprs-C					
MelanDprs-M					-0.084
nMelanDprs-M					
GADDprs		-0.177	-0.234	-0.300	-0.366
GAD					
AnxDprs				-0.097	-0.236
Anx					
GAD(inDprs)					
Anx(inDprs)					

(d) Microarray

c_x (Data)	0.05	0.05	0.05	0.05	0.05
c_y (Label)	0.30	0.35	0.45	0.55	0.58
Correlation	0.5874	0.5793	0.5516	0.5511	0.5519
Dprs		-0.177	-0.466	-0.595	-0.584
MelanDprs-C			**-0.871**	**-0.638**	**-0.602**
nMelanDprs-C					
MelanDprs-M					
nMelanDprs-M					
GADDprs				-0.117	-0.228
GAD					
AnxDprs	**-1.000**	**-0.984**	-0.155	-0.473	-0.494
Anx					-0.002
GAD(inDprs)					
Anx(inDprs)					

Note. The criteria for the class labels are given as follows:

Name	Positive Class Definition	Negative Class Definition
Dprs	depression patients	healthy controls
MelanDprs-C	depression patients with melancholic features defined by CORE scores	healthy controls
nMelanDprs-C	depression patients without melancholic features defined by CORE scores	healthy controls
MelanDprs-M	depression patients with melancholic features defined by MINI interview	healthy controls
nMelanDprs-M	depression patients without melancholic features defined by MINI interview	healthy controls
GADDprs	depression patients with GAD	healthy controls
GAD	GAD patients	non-GAD samples
AnxDprs	depression patients with anxiety	healthy controls
Anx	anxiety patients	non-anxiety samples
GAD(inDprs)	depression patients with GAD	depression patients without GAD
Anx(inDprs)	depression patients with anxiety	depression patients without anxiety

Table 2.3: The sparse CCA experiments between the Depression data sets and the class labels (part II): the weights of top five selected labels for each data set are recorded.

(a) Protein

c_x (Data)	0.05	0.05	0.05	0.05	0.05
c_y (Label)	0.30	0.31	0.35	0.45	0.55
Correlation	0.5765	0.5747	0.5715	0.5634	0.5552
Dprs		0.029	0.098	0.234	0.339
MelanDprs-C	**1.000**	**1.000**	**0.993**	**0.940**	**0.850**
nMelanDprs-C					
MelanDprs-M					0.082
nMelanDprs-M					
GADDprs			0.070	0.234	0.313
GAD					
AnxDprs				0.085	0.241
Anx					
GAD(inDprs)					
Anx(inDprs)					

(b) Transcripts

0.08	0.08	0.08	0.08	0.08
0.30	0.35	0.40	0.50	0.59
0.5192	0.5212	0.5391	0.5310	0.5283
-1.000	**-0.984**	-0.386	-0.557	-0.560
	-0.177	**-0.922**	**-0.758**	**-0.608**
				-0.007
			-0.004	-0.318
		-0.018	-0.338	-0.464

projection. Furthermore, it also implies the feature is more important.

Take the Table 2.2c as an example. A projected Metabolite data have a correlation around 0.47 with the projected labels set. Labels MelanDprs-C, GADDprs, Dprs, AnxDprs and MelanDprs-M are selected sequentially. This implies that, for the Metabolite data, label MelanDprs-C can be a better discriminant criterion compared with other class labels. Other Depression data sources, *i.e.,* Cognitive, Protein and Transcripts, also express similar results (see Tables 2.2a & 2.3a & 2.3b). Among all eleven class labels, the label Dprs and the label MelanDprs-C are the two most commonly selected labels. Therefore, the usage of Dprs or MelanDprs-C as the classification criterion seems to be a proper target in the Depression research.

Verification via Classifications

To verify the above assumption, I test each pair of data and class labels in the classification tasks. In this thesis, I focus on two kinds of classifiers, the random forest (RF) and the support vector machine (SVM).

The random forest is an ensemble learning method that consists of a collection of

tree-structured classifiers [44], while the support vector machine aims to build a decision boundary for the two classes. In this thesis, I employed SVM classifiers by the LIBSVM package [45]. For all SVM classifiers, I choose the linear kernel and set the regularization parameter as 1 in all cases. The classification experiments are executed based on the 10-fold cross validation. The cross validation strategy divides the entire data set into ten parts; at each time, we choose one part as the testing set and the remaining nine parts as the training set. The cross validation process is repeated 10 times, and the strategy ensures that every sample is used in the testing set exactly once.

Before the discussion of the classification results, it is necessary to address two problems in the Depression research: one is the high dimensionality, the other is the imbalanced class distributions. As shown in Table 2.1, there are more than ten thousands of features in the following three data sets: Microarray, Protein and Transcripts. The high dimensionality will significantly affect the learning effectiveness and efficiency. Table 2.4 summarizes the sample statistics for different subtypes of mood disorders. It is clear that most of the learning targets have imbalanced class distributions and these imbalanced class distributions will bias the traditional classifier toward the majority class. To deal with the imbalanced problem, I use an undersampling strategy to adjust the class

Table 2.4: The Depression sample statistics for different subtypes of mood disorders.

	Positive	Negative	Ratio($\frac{Pos}{Neg}$)
Dprs	128	128	1.00
MelanDprs-C	32	128	0.25
nMelanDprs-C	92	128	0.72
MelanDprs-M	67	128	0.52
nMelanDprs-M	57	128	0.45
GADDprs	48	128	0.38
GAD	48	208	0.23
AnxDprs	73	127	0.57
Anx	74	182	0.41
GAD(inDprs)	48	80	0.60
Anx(inDprs)	73	55	1.33

distributions of a data set (see Section 3.3 for details). Some advanced methods are also discussed in Chapter .

Table 2.5 summarizes the classification results of the experiments on each pair of the Depression data sets and class labels [targets GAD(inDprs) and Anx(inDprs) are ignored]. The table includes the accuracy values obtained from either the random forest classifier or the SVM classifier. The table is then colored according to the accuracy values, where a darker color means a higher accuracy. It is clear that in Table2.5, the Depression data sets on the MelanDprs-C target achieve the better performance, that is, the depression patients with the melancholic features defined by CORE scores are more likely to be distinguished from the healthy controls. Similarly, more significant patterns are also shown in the Dprs target. Thus, the classification results well verify the aforementioned conclusion.

Table 2.5: The classification performance of each pair of the Depression data sets and class labels. The cell is colored according to the accuracy value.

	Dprs	Melan Dprs-C	nMelan Dprs-C	Melan Dprs-M	nMelan Dprs-M	GAD Dprs	GAD	AnxDprs	Anx
Cognitive	63.52%	63.44%	61.11%	65.84%	61.32%	56.30%	51.89%	52.99%	53.61%
EBBF	55.56%	59.24%	54.98%	51.00%	52.71%	58.59%	59.58%	54.33%	54.40%
Metabolite	64.04%	76.47%	54.17%	61.28%	53.98%	54.03%	62.11%	58.51%	52.69%
Microarray	61.61%	64.74%	50.20%	56.09%	62.00%	63.73%	58.87%	62.73%	53.32%
Protein	57.07%	64.87%	49.63%	50.38%	52.74%	57.19%	58.84%	51.33%	47.00%
Transcripts	62.63%	62.61%	57.60%	57.21%	53.10%	57.01%	51.53%	62.39%	55.77%

Next, I present the results in a different view as shown in Fig. 2.3, and the performance results are grouped by different data sets. It can be observed from Fig. 2.3 that the first two columns in each group show higher accuracies than the other columns in the same group. Namely, the Dprs and the MelanDprs-C targets perform well in classification tasks.

The above classification results imply that the patterns detected by the sparse CCA

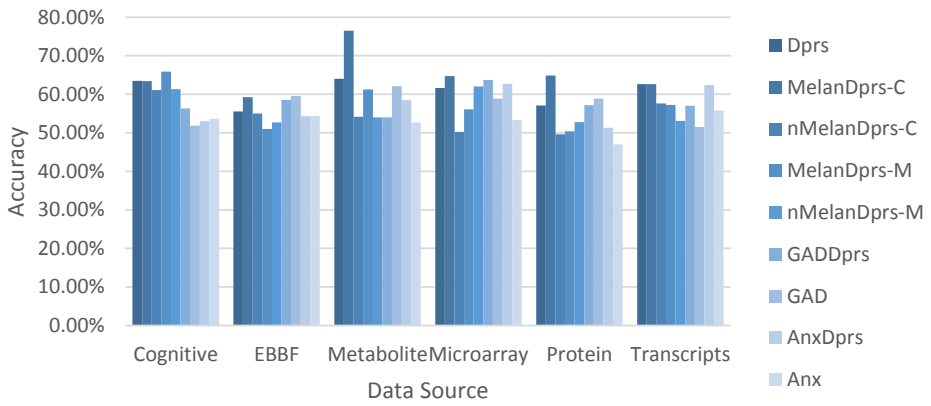


Figure 2.3: The classification performance of each pair of the Depression data sets and class labels grouped by data set.

method is meaningful. Therefore, the sparse CCA method is an effective tool to detect the patterns among the data sets and class labels.

Pattern Detection between Data Sets via sparse CCA Method

The sparse CCA method has been widely applied to deal with other practical problems, *e.g.,* to find patterns among different data sources. When applying the sparse CCA on two data matrices, the method will maximize the correlation between the projections of these matrices. In other words, between the two data sets, a set of higher correlated variables will be determined by the sparse CCA approach. Thus, the sparse CCA method is used to help reveal the potential associations among the features from different data sources.

In this section, I test the above idea on the Alzheimer's disease neuroimaging initiative (ADNI) study. There are many data sources in the ADNI longitudinal study, including blood tests, cerebrospinal fluid tests (CSF), magnetic resonance imaging (MRI), positron emission tomography (PET) imaging, *etc*. The clinical / psychometric assessments (called META) data are also collected. The following experiments are aimed to explore the patterns between the META data and the MRI data. I use all samples

available in these two data sets and a brief report of sample statistics is summarized in Table 2.6.

Table 2.6: The sample statistics of the META data and the MRI data in ADNI study.

	Number of samples	Dimension
META	632	52
MRI	632	305

In order to obtain the most significant patterns between the two data sets, we tune several pairs of penalty constraints. These penalty constraints will directly affect the sparsity of the projection matrices, and further impact on the correlation between the two projections. In practice, I restrict the sparse CCA method to select no more than 15 features from the META data, and no more than 30 features from the MRI data. Under these conditions, I further choose the pair of parameters that yields a highest correlation around 0.6544 between the projections. The selected features and their weights are shown in Table 2.7.

Table 2.7: The sparse CCA experiment results on the META data and the MRI data in ADNI study: (a) the META data; (b) the MRI data.

(a) META

META features	Weights
ADAS_sub4	0.4196
MMSE	-0.3882
CDR	0.3852
LDELTOTAL	-0.3703
FAQ	0.3165
ADAS_sub1	0.3053
LIMMTOTAL	-0.2734
ADAS_sub8	0.2264
CATVEGESC	-0.1621
ADAS_sub7	0.1504
BNTTOTAL	-0.1154
DIGITSCOR	-0.0606
TRABSCOR	0.0410
CATANIMSC	-0.0309

(b) MRI

MRI features	Weights
Cortical Thickness Average of LeftEntorhinal	-0.5150
Cortical Thickness Average of RightEntorhinal	-0.5119
Volume (WM Parcellation) of LeftHippocampus	-0.4686
Volume (WM Parcellation) of RightHippocampus	-0.3532
Cortical Thickness Average of LeftMiddleTemporal	-0.2501
Cortical Thickness Average of LeftInferiorTemporal	-0.1808
Volume (Cortical Parcellation) of LeftEntorhinal	-0.1397
Cortical Thickness Average of RightMiddleTemporal	-0.0998
Volume (WM Parcellation) of RightAmygdala	-0.0453
Cortical Thickness Average of LeftFusiform	-0.0339
Volume (WM Parcellation) of LeftAmygdala	-0.0213

Table 2.7 illustrates that the META features including Alzheimer's disease assessment scale-cognitive subscale (ADAS or ADAS-cog) scores, mini-mental state examination (MMSE), clinical dementia rating (CDR), and the MRI features including entorhinal cortical (ERC) thickness and hippocampus volume are strongly correlated. These results are consistent with prior research in this area.

For example, Velayudhan *et al.* [46] demonstrate that the ERC is a region that will be affected early in AD, and the ERC thickness is related to both longitudinal MMSE and ADAS-cog scores. Li *et al.* [47] demonstrate that the atrophy of the entorhinal cortex has significant association with the ADAS-cog. Jonathan *et al.* [48] show that the MMSE scores and CDR scores are correlated with hippocampal atrophy. Table 2.7 also include LDELTOTAL and LIMMTOTAL, which refer to the tests of logical memory, and are available in the neuropsychological battery tests. Kwangsik *et al.* [49] show that there are strong associations between neuropsychological battery scores and lateral temporal atrophy.

To sum up, the results shown in Table 2.7 are consistent with prior research findings. Therefore, the sparse CCA method is potentially effective in identifying interesting AD biomarkers.

Chapter 3

Learning from Imbalanced Data

In this chapter, I discuss several issues of learning from the imbalanced biomedical data. A large number of existing learning systems are designed under the assumption that the data have balanced class distributions and low-dimensionality. However, most of biomedical data do not satisfied this assumption in practice. In this chapter, I will study the imbalanced learning problems, and discuss appropriate evaluation criteria. I also introduce the sampling methods and ensemble strategies. To obtain significant biomakers and improve the learning performance, I apply multiple feature selection methods. Moreover, I introduce an effective classification solution based on one-class SVM.

3.1 Introduction to Imbalanced Learning

A large number of standard learning algorithms assume that the distributions of two classes are balanced or the misclassification costs are equal (or similar) to each other [4]. The data imbalance brings many challenges to machine learning [50, 51, 52]. For instance,

- **Improper evaluation criteria**

  Evaluation criteria play critical role in algorithm design and result evaluation. Traditional performance measurements (*e.g.,* accuracy and error rate) make the systems pay more attention to the majority class and there is little chance to capture the characteristics from the minority class.

- **Absolute rarity and relative rarity**

  Absolute rarity means that the number of available samples from the minority class is small, even if the whole training set is large. Relative rarity refers to the case of the relative lack of data, which makes rare samples have a small probability to be detected. Either kind of rarities makes it difficult to learn accurate models from the minority class.

- **Inappropriate inductive bias**

  Many general biases applied in conventional learning algorithms are designed for a better ability of generalization and avoiding overfitting, which affects the performance of the learning models for the minority set.

- **Noise**

  When the data set includes noises, it is often difficult to distinguish the noise from the minority observations.

Therefore, for complex imbalanced data, standard machine learning systems may perform poorly. The characteristics of data distributions must be properly captured in order to achieve satisfactory performance.

Imbalance is a common phenomenon in the biomedical domain. We usually have more examples that are normal, *e.g.,* there are often more healthy control subjects available than the patients in biomedical research. Moreover, as mentioned above, biomedical data also suffer from the curse of dimensionality, as the number of features often greater than the number of samples. In short, the imbalanced class distributions are common in biomedical data. The limited number of samples and the high dimensionality make imbalanced learning problems much more difficult.

3.2 Appropriate Evaluation Criteria

To choose the appropriate performance measure is a significate issue in imbalanced learning problems. Since accuracy or error rate cannot well reflect the characteristics of both classes, some alternative evaluation criteria need to be considered.

The confusion matrix, developed by Kohavi and Provost in 1998 [53], is a matrix showing actual conditions and classification results. Given an instance and its classification result, there are four possibilities [54]:

- **True Positive**, if the instance is positive and the classification result is positive;
- **False Negative**, if the instance is positive and the classification result is negative;
- **True Negative**, if the instance is negative and the classification result is negative;
- **False Positive**, if the instance is negative and the classification result is positive.

		Condition	
		Positive	Negative
Test Outcome	Positive	True Positive (TP)	False Positive (FP)
	Negative	False Negative (FN)	True Negative (TN)
	Count	n_{pos}	n_{neg}

Figure 3.1: The confusion matrix.

Recall the aforementioned confusion matrix shown in Fig. 3.1. Terms n_{pos} and n_{neg} denote the number of positive and negative samples, respectively. The classification accuracy measures how many instances are correctly classified, defined as

$$\text{accuracy} := \frac{TP+TN}{TP+TN+FP+FN} = \frac{TP+TN}{n_{pos}+n_{neg}}. \tag{3.1}$$

In imbalanced data, without loss of generality, we assume that $n_{pos} \ll n_{neg}$. Then it is possible that we may obtain a high accuracy, even if the classifier assigns all samples as negative. Thus, accuracy is not an appropriate criterion to evaluate the performance of the learning model.

Besides accuracy, sensitivity and specificity are commonly used as the evaluation criteria:

$$\text{sensitivity} := \frac{TP}{TP+FN}; \tag{3.2}$$

$$\text{specificity} := \frac{TN}{TN+FP}. \tag{3.3}$$

The sensitivity (also called Recall rate) measures the proportion of real positive cases that are correctly identified as such. The specificity measures the proportion of real negative cases that are correctly identified as such. In other words, the sensitivity indicates the quality of models that captures the positive set, while the specificity refers to the quality of models that captures the negative cases. Thereby, these two performance criteria can better evaluate the classification results for imbalanced data.

Other criteria have also been considered in imbalanced learning, *e.g., Harmonic mean* (H-mean), *geometric mean* (G-mean), *precision* and *F-measure* [55, 56, 57]. These criteria are defined as follows:

$$\mathrm{H-mean} := \frac{2 \cdot \mathrm{sensitivity} \cdot \mathrm{specificity}}{\mathrm{sensitivity} + \mathrm{specificity}}, \tag{3.4}$$

$$\mathrm{G-mean} := \sqrt{\mathrm{sensitivity} \times \mathrm{specificity}}, \tag{3.5}$$

$$\mathrm{precision} := \frac{TP}{TP+FP}, \tag{3.6}$$

$$\mathrm{F-measure} := (1+\beta^2) \cdot \frac{\mathrm{precision} \cdot \mathrm{sensitivity}}{\beta^2 \cdot \mathrm{precision} + \mathrm{sensitivity}}. \tag{3.7}$$

Moreover, the receiver operating characteristic (ROC) graph [54] also commonly used to evaluate classifiers.

3.3 Sampling Methods and Classifiers Ensemble

Random Undersampling Method

To learn a better model from an imbalanced data set, a simple and intuitive idea is to balance the training set. To achieve data balance, we need the training set to contain approximately equal numbers of observations from each category. Sampling method is about selecting suitable samples from the entire observation set, and it is frequently used to deal with data imbalance issue. In this thesis, I focus on the undersampling method.

Some existing studies suggested that the undersampling method is effective to deal with imbalance [20, 21]. Random undersampling is the technique used to adjust the class

distributions of a data set. Given an imbalanced data with $n_{pos} \ll n_{neg}$, the undersampling strategy randomly removes samples from the majority class, *i.e.,* negative set, until the number of examples kept in the majority class matches with the size of the minority set.

In some cases, sampling may cause negative impacts on learning, since undersampling may discard some potentially useful training instances from the majority class. For example, given an SVM classifier, the trained hyperplane between the positive and negative set is significantly affected by the undersampling method. More specifically, the information captured from the original majority class of samples become less, and accordingly, the learned hyperplane may not well reflect the majority set. Thus, both of accuracy and specificity may be low if we perform classification based on such obtained model. However, on the contrary, since we get a balanced class distribution after random undersampling, the classifier can better capture the characteristics of the minority class. Therefore, the sensitivity may increase.

In learning from the imbalanced biomedical data, it is desired that a learning model will produce a high accuracy, a satisfactory sensitivity and a desired specificity. For example, in the Depression Metabolite data - Melancholic Depression research, we have an imbalanced Metabolite data set with a ratio around $1:5$ of the positive class (patients with melancholic depression) and negative class (healthy controls). One of the research targets is to build accurate classifiers that are able to well identify the melancholic depressive patients from healthy controls. Traditional machine learning methods are ineffective because the classifiers trained based on the imbalanced cases will be biased toward the majority class, that is, the depressive patients may not be well identified.

Compared with undersampling, the random oversampling is a process of randomly resampling from the minority class. Previous studies have shown oversampling is often less effective than undersampling [20, 58]. There are also other sampling methods proposed in the literature including *e.g.,* the informed undersampling [22], the synthetic

minority oversampling technique (SMOTE) [23], the sampling with data cleaning techniques [4, 24] and the cluster-based oversampling (CBO) [25].

Classifiers Ensemble

As discussed above, the random undersampling tends to discard some potentially useful samples from the majority class. To address this problem, I introduce ensemble methods to solve imbalanced learning problems.

Ensemble methods refer to the process of combining multiple models to improve predictive performance [59, 60, 61]. Many ensemble methods have been proposed, *e.g.,* the bootstrap aggregating (bagging), the boosting, the Bayesian model averaging and combination. In this thesis, I apply a bagging strategy for imbalanced-data learning.

The idea of classifiers ensemble is to build a prediction model by combining a set of individual decisions from multiple classifiers [62, 63]. Such a combination is processed based on the weighted voting or the unweighted voting (majority voting). The ensemble predictions are often more accurate than the individual classifiers. Take the following as an example:

Example 1. Given a sample, there are n (n is odd) base classifiers available, and each classifier is independent with an error rate p. I then use majority voting to do ensemble learning. The probability that the ensemble classifiers makes a wrong prediction is:

$$P(\text{error}) = \sum_{i=\lceil \frac{n}{2} \rceil}^{n} \left(\begin{array}{c} n \\ i \end{array} \right) p^i (1-p)^{n-i}. \tag{3.8}$$

Assume $n = 31$, we then arrive at the following results: if $p = 0.5$, $P(\text{error}) = 0.5$; if $p = 0.45$, $P(\text{error}) = 0.2868$; if $p = 0.4$, $P(\text{error}) = 0.1248$; if $p = 0.35$, $P(\text{error}) = 0.0424$; if $p = 0.3$, $P(\text{error}) = 0.0095$.

Example 1 indicates that if the individual error rate is $p < 0.5$, the error rate of voting ensemble will decrease.

Undersampling-based Ensemble Framework

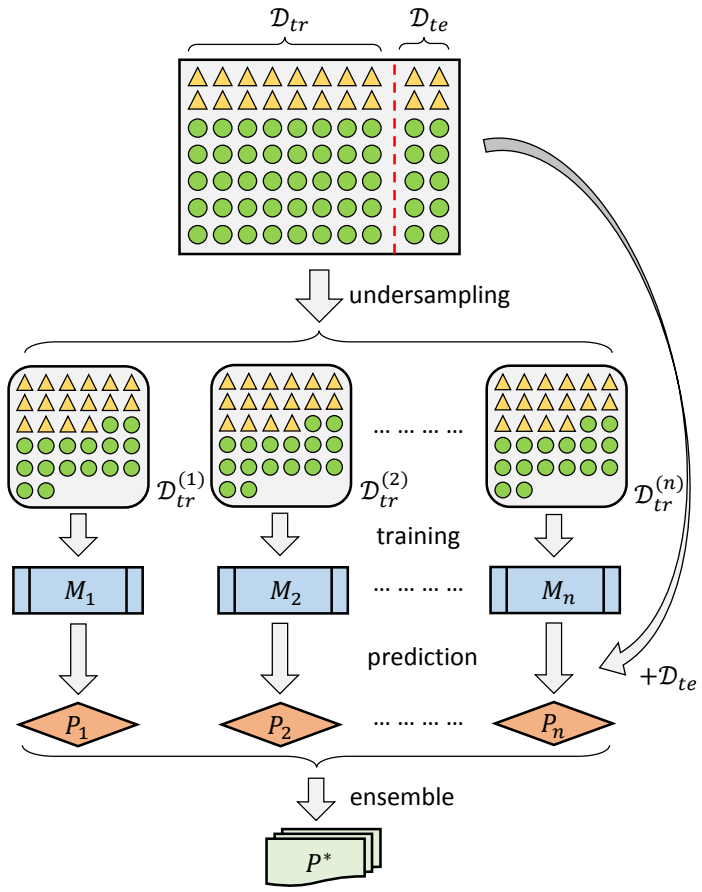


Figure 3.2: The framework of undersampling-based classifiers ensemble (UEM).

I apply the ensemble idea in imbalanced learning by incorporating the undersampling method. Figure 3.2 shows the framework of the undersampling-based classifiers ensemble (UEM) method. Given an imbalanced training data set $\mathscr{D}_{tr}$ and some testing data $\mathscr{D}_{te}$, there are three tasks in the UEM framework:

1. **Training**: apply undersampling on the entire training data set $\mathscr{D}_{tr}$ multiple times and obtain the corresponding sub-training set $\mathscr{D}_{tr}^{(1)}, \cdots, \mathscr{D}_{tr}^{(n)}$ (n refers to the total times of undersampling). For a single subset $\mathscr{D}_{tr}^{(i)}$, a classifier model M_i is learned.

2. **Prediction**: on the testing data $\mathscr{D}_{te}$, make prediction using each classifier model M_i and obtain classification result P_i and its weight w_i.

3. **Ensemble**: combining all predictors and assign the class labels via majority voting or weighted voting. That is, the final prediction is given by $P^* = \sum_i^n w_i P_i$. Note that we fix $w_i = 1$ for $1 \leq i \leq n$, if for majority voting.

The UEM framework takes the advantages of undersampling and ensemble method, that is, every classifier model is learned from a balanced data and the combination of multiple models is expected to improve the prediction performance.

3.4 Feature Selection Methods for Imbalanced Learning

Introduction to Feature Selection

Feature selection refers to the process of choosing features from the original set based on some criteria, and then the derived subset of features will be used to develop the resultant learning models [30, 31]. There is a principal assumption that supports the usage of feature selection: the data contain some redundant or irrelevant variables. This phenomenon commonly appears in high-dimensional data. Since most of biomedical data suffer from the curse of dimensionality, we expect that the feature selection would be an effective tool in learning from the biomedical data, as well as a powerful dimension reduction technique for the high-dimensional data.

Moreover, in bioinformatics research, scientists are often interested in the following questions:

- What features can best characterize the different classes of samples?
- What features may have great impact on classification performance?
- Which biomarkers are the causative factors?

All of the above questions are related to feature selection. Note that compared with feature extraction techniques, feature selection methods directly select a subset of features from the original feature space. From this view of point, feature selection produce more interpretable models.

To sum up, feature selection methods can improve the efficiency of the learning models. After the reduction of feature dimension, the overall learning complexity and running time decrease significantly. Moreover, in a reduced feature space, we can better visualize the patterns in the data set and detect the noise or outliers from the samples.

Feature Selection Methods

Feature selection methods can be generally categorized into three types: filters, wrappers and embedded methods. In this thesis, I take advantages of the feature selection algorithms provided by the Arizona State University (ASU) Feature Selection Repository [64]. The following summarizes the feature selection algorithms employed in this thesis[1].

Gini Index

The Gini index proposed by Corrado Gini is a filter method. A Gini score is calculated to measures the abilities of features to distinguish between classes [65, 66, 64]. Based on the Lorentz curve, the Gini index (GI) of a term (feature) f among C classes is defined as

$$GI(f) = 1 - \sum_{i=1}^{C} [pr(c_i|f)]^2. \tag{3.9}$$

In a normalized sample space,

$$Pr(c_i|f) = \frac{Pr(f|c_i)}{\sum_{k=1}^{|C|} Pr(f|c_i)}, \tag{3.10}$$

$$Pr(f|c_i) = \frac{1 + N(f,c_i)}{|V| + \sum_{f \in V} N(f,c_i)}. \tag{3.11}$$

[1]Other feature selection methods, *e.g.,* sparse Logistic Regression with Bysesian Regularization, Chi-square Score, Fisher Score, Kruskal-Wallis, Minimum Redundancy and Maximum Relevance (mRmR) and Student's t-test, are also included in the UFSEM framework, but I omit the details in this thesis.

Note that $N(f, c_i)$ denotes the probability that the term f occurs in a class c_i and V is the vocabulary set.

According to the criterion (3.9), the procedure of computation of the Gini index is independent for each feature. Moreover, a feature with a smaller Gini score is more significant.

Information Gain

The information gain is also a filter method that measures the dependence between a feature and the class labels [67, 64]. To evaluate the information gain (IG) of a feature f among $C = \{c_i\}_{i=1}^{m}$ classes, we use the following formula:

$$\begin{aligned} IG(f) = &- \sum_{i=1}^{m} Pr(c_i) \log Pr(c_i) \\ &+ Pr(f) \sum_{i=1}^{m} Pr(c_i|f) \log Pr(c_i|f) \\ &+ Pr(\bar{f}) \sum_{i=1}^{m} Pr(c_i|\bar{f}) \log Pr(c_i|\bar{f}). \end{aligned} \tag{3.12}$$

Similar to the Gini index, the information gain method deals with each feature independently. However, a higher score obtained by the information gain indicates that the corresponding feature is more relevant.

In addition, it is noteworthy to point out that, the above two feature selection methods do not eliminate the redundant features, since both methods evaluate each feature independently and rank all features based on their weights.

Stability Selection

In order to deal with the high-dimensional data and the data with redundant features, I introduce the stability selection approach. Stability Selection is a feature selection algorithm based on subsampling and combines the usage of a proper amount of regularization [68].

To Stability Selection, the first step is to randomly subsample half samples from the original data set. For each subsampling, we next utilize the lasso and the sparse logistic regression to select variables. Multiple parameter values are tested in each subsampling to choose truly relevant variables. Then, the algorithm calculates the selection probability for each variable and rank the features from the maximum selection probability to the minimum probability. Besides, an extension of calculating the average of top-k selection probabilities [69] is also implemented in this thesis.

In all experiments of this thesis, the stability selection is executed based on 1000 times subsampling and 10 regularization parameter values. I use the sparse logistic regression function from the SLEP package [70] and the parameter values are determined such that about 10 - 300 features are selected (or $\frac{1}{3}$ of all features at maximum).

Feature Selection Methods for the UEM Framework

Figure 3.3 shows the framework of the combination of feature selection and undersampling-based classifiers ensemble (UFSEM). Compared with the previous UEM framework, there are some modifications as follows:

- In the training stage, after obtaining n subsets from the original training data via undersampling, a feature selection method F is applied to each subset $\mathscr{D}_{tr}^{(i)}$ to obtain the corresponding ranking list F_i. We then train the classifier model M_{Fi} by using $\mathscr{D}_{tr}^{(i)}$ and F_i, *i.e.,* a subset of selected features from $\mathscr{D}_{tr}^{(i)}$ is used to learn model M_{Fi};
- In the prediction stage, for each model M_{Fi} obtained in the training stage, we use the corresponding ranking list F_i to re-express the testing set $\mathscr{D}_{te}^{(Fi)}$. We then make a single prediction P_{Fi} by using the data $\mathscr{D}_{te}^{(Fi)}$ and the model M_{Fi}.

Since the feature dimension is reduced after feature selection, the learning complexity and running time decrease significantly. For complex imbalanced (*i.e.,*

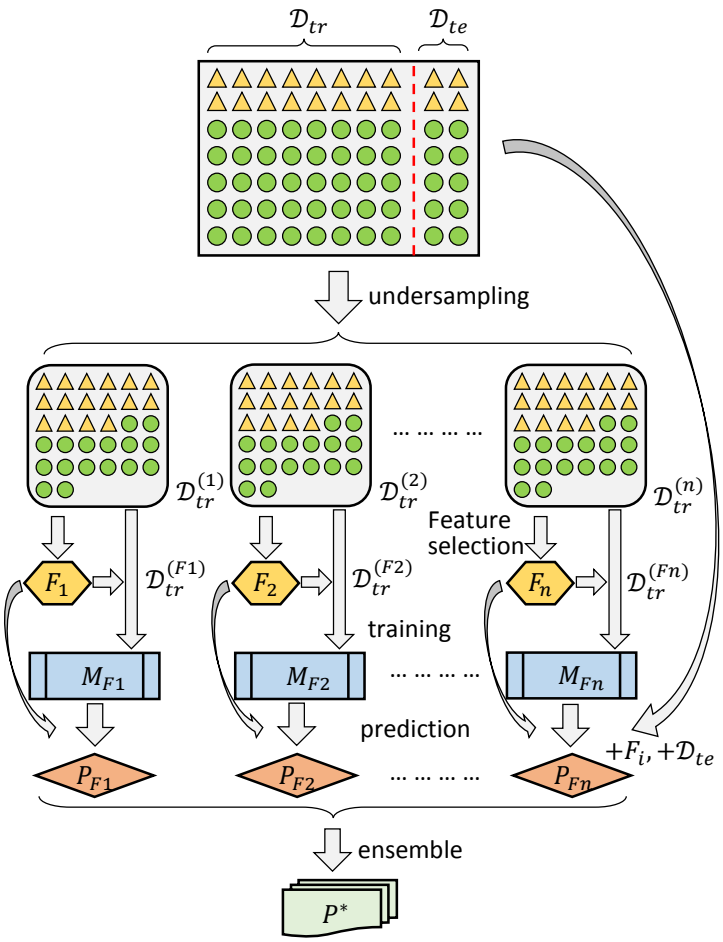


Figure 3.3: The framework of the combination of feature selection and undersampling-based classifiers ensemble (UFSEM).

imbalanced and high-dimensional) data, the UFSEM framework will further improve the learning performance.

The UFSEM framework in figure 3.3 illustrate a pipeline of one single feature selection strategy. This framework can be easily extended to more complicated cases, *e.g.*, the combination of multiple feature selection methods and the one based on different numbers of selected features.

3.5 Novelty Detection Idea for Imbalanced Learning

In addition to the methods discussed in Sections 3.3 and 3.4, to deal with the imbalanced classification problems, another idea is to use the novelty detection method.

The novelty detection, as known as anomaly detection or outlier detection, refers to the technique of building patterns that capture the characteristics of normal samples and detect any divergence or unexpected behaviors [71]. Sometimes, the novelty detection is also called one-class classification problem. Differing from binary classification problem that discriminates the positive and negative samples, one-class classification takes advantage of the information from the normal class and aims at finding a better description (hypersphere) for the normal data. For imbalanced biomedical data with $n_{pos} \ll n_{neg}$, healthy controls are apparently considered as the normal samples in novelty detection, and patients are treated as outliers.

Recently studies on novelty detection studies suggest that we should focus on the normal class, but also utilize the available abnormal information, *e.g.,* Support Vector Data Description (SVDD) approach [72]. Given a data set $\mathbf{X} = [\mathbf{x_1}, \ldots, \mathbf{x_n}] \in \mathbb{R}^{n \times d}$, we denote $n = m_1 + m_2$, where the first m_1 samples of n are normal (positive) samples, and the remain, m_2 samples are outliers (negative class). The objective of SVDD is to build a hypersphere in the feature space $\mathscr{F}$ that includes most of the normal observations and keeps the outliers outside. This idea can be formalized as the following optimization problem:

$$\min_{R,\mathbf{c},\xi} R^2 + C_1 \sum_{i=1}^{m_1} \xi_i + C_2 \sum_{j=m_1+1}^{n} \xi_j, \tag{3.13}$$

$$\text{subject to } \|\phi(\mathbf{x}_i) - \mathbf{c}\|^2 \leq R^2 + \xi_i, 1 \leq i \leq m_1; \tag{3.14}$$

$$\|\phi(\mathbf{x}_j) - \mathbf{c}\|^2 \leq R^2 - \xi_j, m_1 < j \leq n; \tag{3.15}$$

$$\xi_k \geq 0, 1 \leq k \leq n, \tag{3.16}$$

where $\phi(\cdot)$ stands for the mapping function from $\mathbf{X}$ to $\mathscr{F}$; $\mathbf{c} \in \mathscr{F}$ and $R > 0$ are the center and the radius of the hypersphere builded in $\mathscr{F}$, respectively; $\xi = [\xi_1, \ldots, \xi_n]^T \in \mathbb{R}^n$ is the vector of slack variables; and C_1, C_2 are the tuning parameters.

Based on SVDD, Wu and Ye proposed another approach - the small sphere and large margin (SSLM) method, which utilizes additional information from the negative set [73]. SSLM method extends SVDD by maximizing the margin ρ between normal samples and the outliers, while SVDD only keeps the outliers away from the region. The SSLM formulation is given as follows:

$$\min_{R,\mathbf{c},\xi} R^2 - C_0\rho^2 + C_1 \sum_{i=1}^{m_1} \xi_i + C_2 \sum_{j=m_1+1}^{n} \xi_j, \tag{3.17}$$

$$\text{subject to } \|\phi(\mathbf{x}_i) - \mathbf{c}\|^2 \leq R^2 + \xi_i, 1 \leq i \leq m_1; \tag{3.18}$$

$$\|\phi(\mathbf{x}_j) - \mathbf{c}\|^2 \leq R^2 + \rho^2 - \xi_j, m_1 < j \leq n; \tag{3.19}$$

$$\xi_k \geq 0, 1 \leq k \leq n, \tag{3.20}$$

where ρ is a real number, $\rho^2 \geq 0$ is denoted as the margin between the boundary of the hypersphere and the outliers, and C_0 is the tuning parameter for the margin.

SVDD and SSLM methods take advantage of the information from both classes of samples. These one-class classification ideas can be used in the complex imbalanced learning problems, especially for the case that few minority samples are available. We can treat the majority class of samples as normal ones, and the samples of the minority class are regarded as the outliers.

Compared with the UFSEM framework, these one-class classification approaches have many beneficial. First of all, similar to the UFSEM framework, one-class classification can perform well in imbalanced classification tasks. Second, abnormal detection may be more efficient because only one classification operation is needed.

However, it is noteworthy to point out that neither SVDD nor SSLM supports feature selection.

3.6 Experiments

In this section, I conduct several experiments using the imbalanced biomedical data in the Depression research. The undersampling-based classifiers ensemble framework is used to deal with the imbalanced data set. In order to reduce the dimensionality and obtain useful features, multiple feature selection methods are implemented in the UFSEM framework. Moreover, I employ the SSLM one-class approach to identify abnormal samples.

Note that, in the following parts, I mainly focus on the target MelanDprs-C, *i.e.,* the task of differentiating the melancholic depression patients from the healthy controls. I use the Metabolite data, the Protein data and the Transcripts data in the experiments. Some dimension reduction strategies are applied, for example:

- Use a reduced Metabolite data set that removes a list of metabolites, these metabolites are highly sensitive to the storage time (see Appendix B for details);
- Use a reduced Protein data set based on the Immune Gene list and the related mapping file (detailed gene list is not attached in the thesis due to its length);
- Use a reduced Transcripts data based on on the Depression Gene list (detailed gene list is not attached in the thesis due to its length).

Table 3.1: The sample statistics of the Depression data set used in the MelanDpres-C target.

	All	Pos	Neg	Dimension
Metabolite	118	21	97	270
Metabolite_R	118	21	96	228
Protein_D	122	29	93	3181
Transcript_I	124	28	96	1438

Table 3.1 summarizes the sample statistics of the used data sets. It shows that the data sets used in the MelanDprs-C target are all severely imbalanced. The Metabolite data are further corrected based on the storage time at the Depression and imputed using KNN method. In addition, the Protein data are imputed by the halfMin method. The feature selection methods discussed in this thesis are restricted to the Gini index method, the information gain method and Stability Selection. Moreover, I still use the random forest and the SVM as the classifiers.

Undersampling Method and Imbalanced Learning

I first demonstrate the usage of the undersampling method in imbalanced learning.

Table 3.2 summarizes the classification performance of multiple the Depression data sets on the MelanDprs-C target. Note that, in this experiment, we do not balance the training set, but only use the 10-fold cross validation. It is clear that although the accuracies shown in Table 3.2 are around 80% in most of the data sets, the sensitivities are much lower than the accuracies or the specificities. Recall the sensitivity measures the proportion of real positive cases that are correctly classified and here, the melancholic depression patients denote the positive class. Therefore, the results imply that the classifiers are all biased toward the negative set, *i.e.,* the majority class.

Table 3.2: The melancholic depression classification performance on the Metabolite data, the Protein data, and the Transcripts data; all features are included in this experiment.

	Metabolite		Metabolite_R		Protein_D		Transcripts_I	
	RF	SVM	RF	SVM	RF	SVM	RF	SVM
Accuracy	82.58%	81.32%	82.38%	78.88%	81.05%	80.99%	78.95%	78.25%
Sensitivity	15.00%	35.00%	10.00%	25.00%	26.67%	43.33%	13.33%	35.00%
Specificity	98.00%	91.44%	98.00%	90.44%	97.89%	92.44%	97.78%	90.56%

The above results are not surprising. Next, I will show the effectiveness of the undersampling method.

The undersampling method randomly removes some samples in the majority class until the two classes achieve balance. In this experiment, I do undersampling once on the training set in each cross validation. A classifier is then trained based on a sampled training set and tested with the testing set. Table 3.3 summarizes the classification performance based on the one-time undersampling method.

Table 3.3: The classification performance of melancholic depression based on the undersampling method.

	Metabolite		Metabolite_R		Protein_D		Transcripts_I	
	RF	SVM	RF	SVM	RF	SVM	RF	SVM
Accuracy	76.95%	71.21%	73.92%	60.88%	72.79%	70.29%	64.46%	65.36%
Sensitivity	66.83%	60.00%	62.22%	40.00%	68.33%	61.67%	65.00%	65.00%
Specificity	79.32%	73.89%	76.57%	65.33%	74.00%	72.78%	64.44%	65.33%

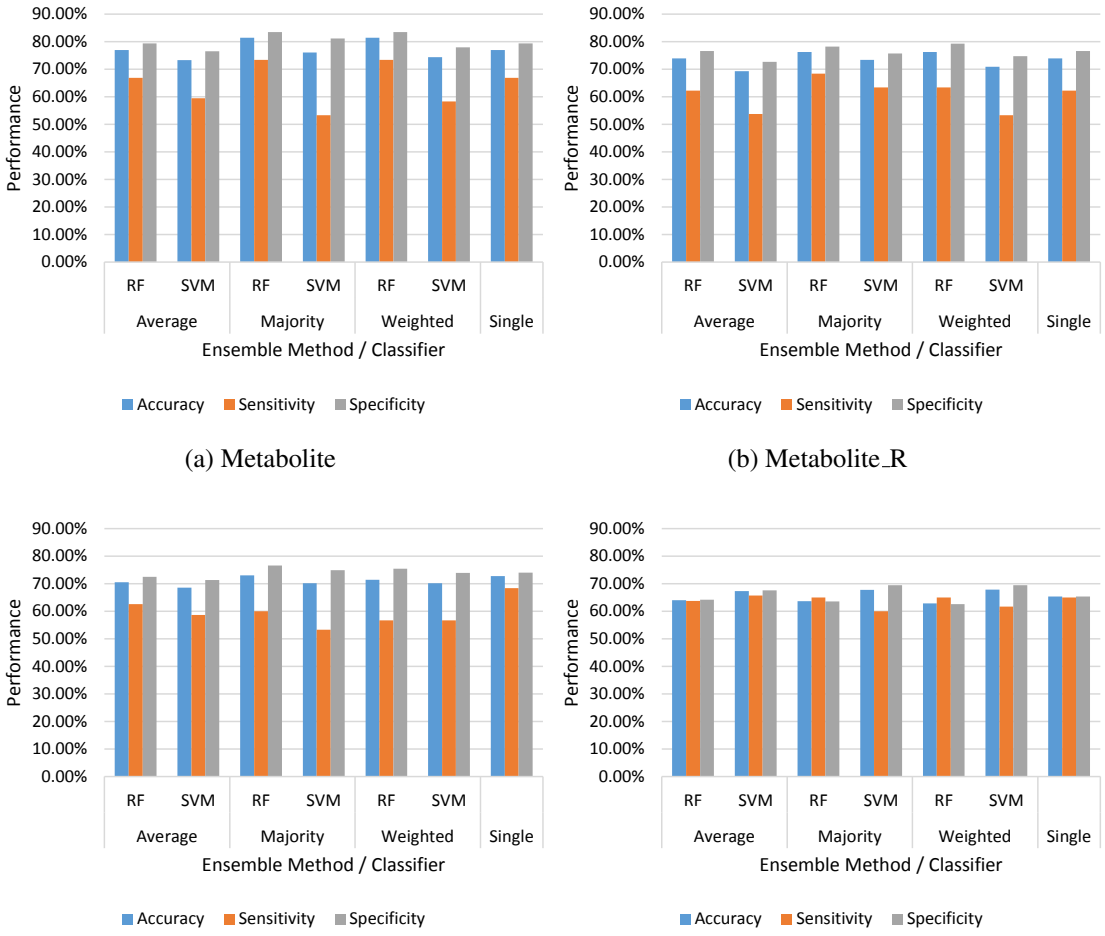
Compared with Table 3.2, the three performance criteria: accuracy, sensitivity and specificity are closer to each other in all cases. Thus, applying undersampling can lead to similar learning performance on both classes of samples.

UEM Framework and Imbalanced Learning

As mentioned in Section 3.3, the undersampling method may bring some uncertainty due to the inappropriate samplings. In other words, the method may discard some potentially significant instances of the majority class in training. An intuitive solution is to increase the number of sampling. The key to the UEM framework is the ensemble learning phase. Based on the undersampling method, I next consider the undersampling-based classifiers ensemble framework.

The figures shown in Fig. 3.4 are the classification results of the UEM framework, which is based on 30 undersamplings. We compare the classification performance of the simple averaging strategy, the majority voting strategy and the weighted voting strategy. At the end of each figure, I enclosed the previous best results of the single undersampling.

Compared with the classification results obtained in a single undersampling, Fig. 3.4a & 3.4b show that, for the two Metabolite data sets, the UEM framework achieves around $3\% - 5\%$ improvement on most performance measurements. The classification performance of the Protein data shown in Fig. 3.4c shows a slight decrease and the Transcripts data shown in Fig. 3.4d does not show a clear difference. It is noteworthy that



(a) Metabolite

(b) Metabolite_R

(c) Protein_D

(d) Transcripts_I

Figure 3.4: The melancholic depression classification performance based on the UEM framework and different ensemble strategies. Average means the average strategy, Majority refers to the majority voting strategy and Weighted stands for the weighted voting strategy.

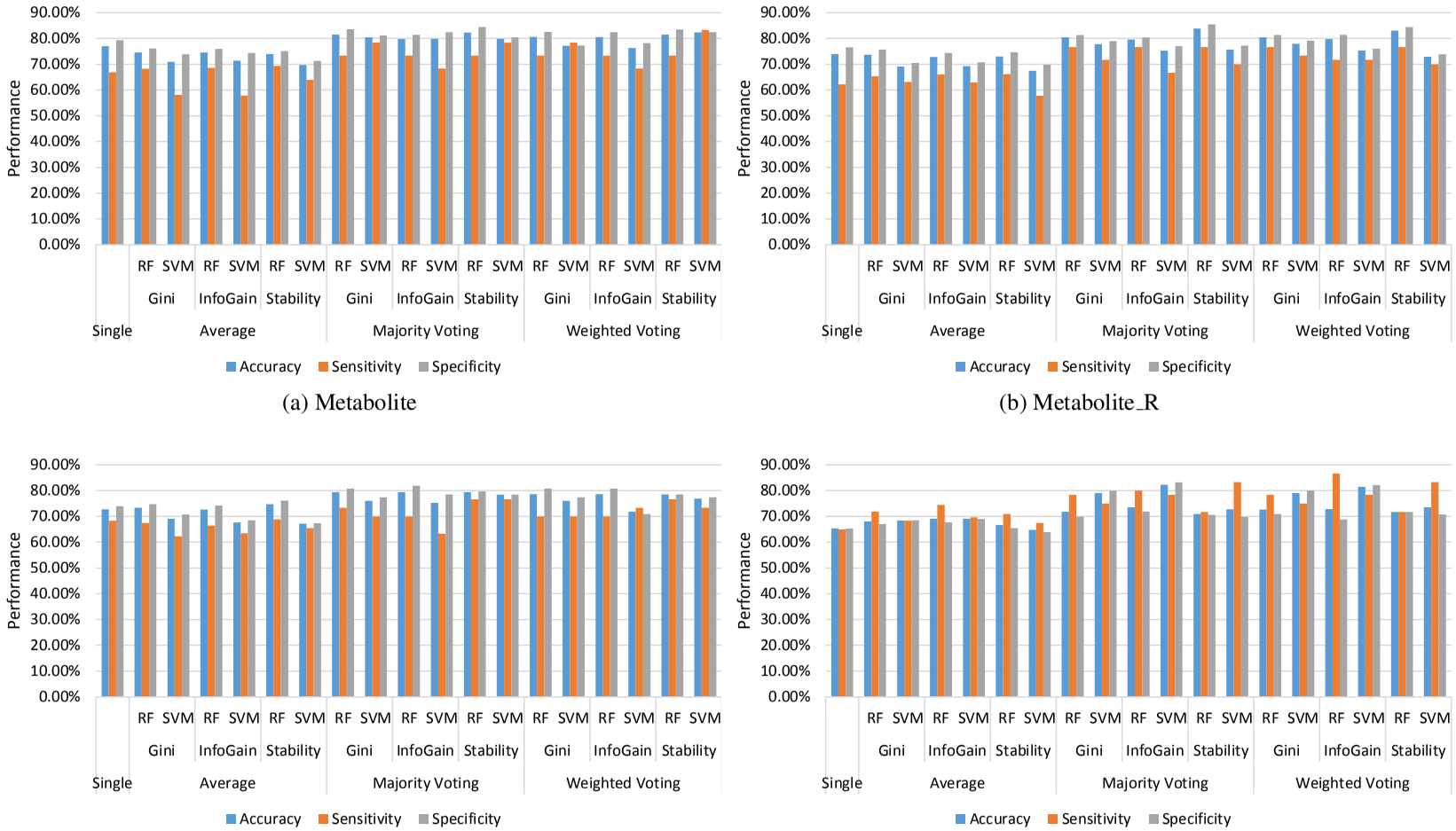
the results obtained from multiple undersamplings are more reliable than those from a single sampling.

We can also compare different ensemble strategies via Figs. 3.4. The majority voting and the weighted voting strategies perform better than the average strategy in the two Metabolite data sets, while these three strategies do not show any significant difference on the Protein data and the Transcript data.

UEM Framework with Feature Selection Methods

The UFSEM framework introduced in Section 3.4 is an imbalanced learning tool that takes advantage of feature selection. In the following experiments, I apply the Gini index method, the information gain method and Stability Selection in the learning tasks. For each sampled training set, I first obtain a feature ranking list via a feature selection method. The next step is to train the classifier based on a certain number of features. In order to improve the efficiency of the learning system and obtain useful variables, I use the first $3, 6, \cdots, 45$ features to train the classifiers.

Fig. 3.5 illustrates the melancholic depression classification results based on the UFSEM framework. It is clear from the figures that the classification performance is significantly improved in all data sets. Both majority voting and weighted voting strategies give better results than the single undersampling approach and the average strategy. More especially, Fig. 3.5a shows that the Gini-SVM-Majority voting, the Stability-SVM-Majority voting and the Stability-SVM-Weighted voting strategies perform around 80% in the Metabolite data, in terms of accuracy, sensitivity and specificity. The reduced Metabolite data shown in Fig. 3.5b indicate that the majority voting based Random Forest classifiers obtain good performance in both feature selection methods. The best performance is obtained via Stability Selection and the majority voting strategy in the Protein data in Fig. 3.5c. In addition, for the Transcript data, the InfoGain-SVM method increases the learning performance by 15%.



(a) Metabolite

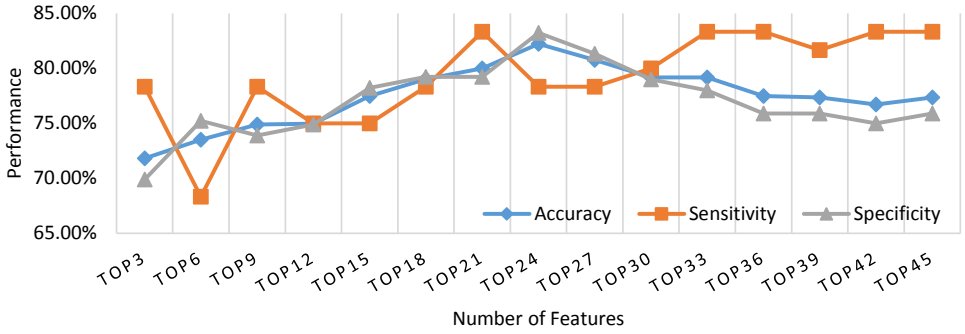
(b) Metabolite_R

(c) Protein_D

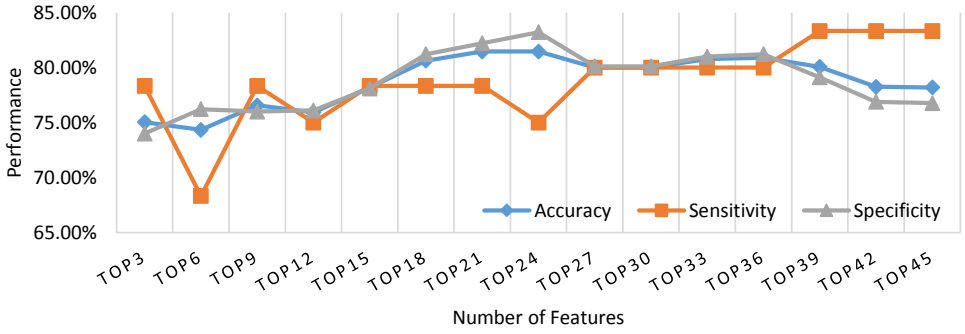
(d) Transcripts_I

Figure 3.5: The melancholic depression classification performance based on the UFSEM framework and multiple ensemble strategies and multiple feature selection methods.

In the UFSEM framework, we can compare the classification performance of using different number of features. Figure 3.6 presents the experiments based on the Transcript data set using the SVM classifier and the information gain method. The figures include both majority voting and weighted voting strategies. The blue lines show the accuracies. It can be observed in Fig. 3.6a that we obtain the best performance via the top 24 selected features. In the case of weighted voting shown in Fig. 3.6b, the classifiers achieve better performance by using approximately $27 - 36$ features.



(a) Information Gain & Majority Voting



(b) Information Gain & Weighted Voting

Figure 3.6: The melancholic depression classification performance on the Transcript data; Information Gain is used for feature selection and different number of features are used.

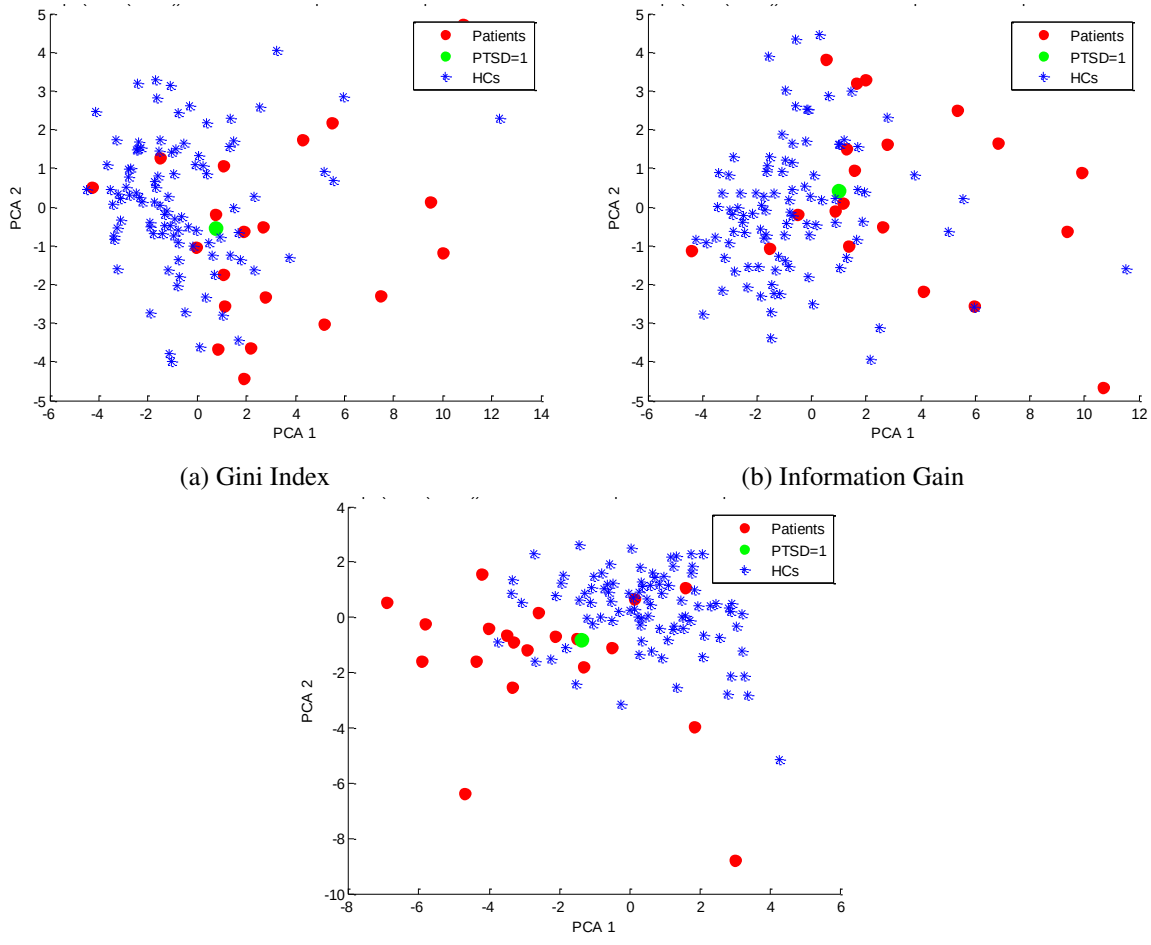
In addition, the UFSEM framework can provide a feature ranking among the features, which is the byproduct of the feature selection methods. After obtaining the feature rankings from each undersampling, we can combine these lists together and generate a final ranking.

Table 3.4: The metabolite features obtained by different feature selection methods using the UFSEM framework based on the two Metabolite data sets.

Knn3	All features Gini	Reduced features Gini	All features InfoGain	Reduced features InfoGain	All features Stability	Reduced features Stability
Rank 1	Triacylgyceride_hydroperoxide_.C16.0,C18.1,C18.2.OOH.	Triacylgyceride_hydroperoxide_.C16.0,C18.1,C18.2.OOH.	Triacylgyceride_hydroperoxide_.C16.0,C18.1,C18.2.OOH.	Triacylgyceride_hydroperoxide_.C16.0,C18.1,C18.2.OOH.	Azelaic_acid_ratio	Triacylgyceride_hydroperoxide_.C16.0,C18.1,C18.2.OOH.
Rank 2	Triacylgyceride_hydroperoxide_.C18.1,18.2,C18.2.OOH._.additio	Triacylgyceride_hydroperoxide_.C18.1,18.2,C18.2.OOH._.additio	Triacylgyceride_hydroperoxide_.C18.1,18.2,C18.2.OOH._.additio	Triacylgyceride_hydroperoxide_.C18.1,18.2,C18.2.OOH._.additio	Triacylgyceride_hydroperoxide_.C16.0,C18.1,C18.2.OOH.	Triacylgyceride_hydroperoxide_.C18.1,18.2,C18.2.OOH._.additio
Rank 3	Azelaic_acid_ratio	Cysteine_.minor._Cystine._ratio	Azelaic_acid_ratio	Serotonine_ng_per_ml	Triacylgyceride_hydroperoxide_.C18.1,18.2,C18.2.OOH._.additio	Cysteine_.minor._Cystine._ratio
Rank 4	Triacylgyceride_hydroperoxide_.C16.0,C18.1,C18.3.OOH._.additi	Dodecanol_ratio	Serotonine_ng_per_ml	Cysteine_.minor._Cystine._ratio	Cysteine_.minor._Cystine._ratio	Indole.3.acetic_acid_ratio
Rank 5	Cysteine_.minor._Cystine._ratio	Serotonine_ng_per_ml	Triacylgyceride_hydroperoxide_.C16.0,C18.1,C18.3.OOH._.additi	Dodecanol_ratio	Indole.3.acetic_acid_ratio	Cystine_ratio
Rank 6	Dodecanol_ratio	Cystine_ratio	Cysteine_.minor._Cystine._ratio	Cystine_ratio	Unknown.68100434._ratio	Ratio Glu_versus_Gln
Rank 7	Serotonine_ng_per_ml	Lysophosphatidylcholine_.C20.4._ratio	Dodecanol_ratio	Normetanephrine_ng_per_ml	Cystine_ratio	Unknown.58100024._ratio
Rank 8	Cystine_ratio	Urea_ratio	Cystine_ratio	Lysophosphatidylcholine_.C20.4._.additional.	Unknown.58100024._ratio	Unknown.38100468._ratio
Rank 9	Unknown.68100434._ratio	Conjugated_linoleic_acid_.C18.trans.10.cis.12.2._ratio	Normetanephrine_ng_per_ml	Methionine_ratio	Unknown.38100468._ratio	Normetanephrine_ng_per_ml
Rank 10	Lysophosphatidylcholine_.C20.4._ratio	Normetanephrine_ng_per_ml	Taurine_ratio	TAG_.C55H100O6._.e_g__C16.0_C18.1_C18.2._ratio	Ratio Glu_versus_Gln	Corticosterone_ng_per_ml
Rank 11	Urea_ratio	TAG_.C55H100O6._.e_g__C16.0_C18.1_C18.2._ratio	Unknown.68100426._ratio	Urea_ratio	Urea_ratio	Urea_ratio
Rank 12	Conjugated_linoleic_acid_.C18.trans.10.cis.12.2._ratio	Glycerol__lipid_fraction_ratio	Lysophosphatidylcholine_.C20.4._.additional.	Eicosenoic_acid_.C20.cis.11.1._ratio	Normetanephrine_ng_per_ml	Unknown.38100389._ratio
Rank 13	Normetanephrine_ng_per_ml	Heptadecanoic_acid_.C17.0._ratio	Methionine_ratio	Lysophosphatidylcholine_.C20.4._ratio	X3_4.Dihydroxyphenylglycol_ng_per_ml	Arginine_ratio
Rank 14	TAG_.C55H100O6._.e_g__C16.0_C18.1_C18.2._ratio	Glyoxylate_ratio	Unknown.68100434._ratio	Unknown.68100045._ratio	Cryptoxanthin_ratio	Androstendion_ng_per_ml
Rank 15	Glycerol__lipid_fraction_ratio	Isoleucine_ratio	TAG_.C55H100O6._.e_g__C16.0_C18.1_C18.2._ratio	Glycerol__lipid_fraction_ratio	Unknown.38100389._ratio	Phosphatidylcholine__8_ratio
Rank 16	Heptadecanoic_acid_.C17.0._ratio	Phenylalanine_ratio	Urea_ratio	Phenylalanine_ratio	Corticosterone_ng_per_ml	Unknown.68100002._ratio
Rank 17	Unknown.68100426._ratio	Unknown.68100045._ratio	Eicosenoic_acid_.C20.cis.11.1._ratio	Conjugated_linoleic_acid_.C18.trans.10.cis.12.2._ratio	Unknown.68100002._ratio	Serotonine_ng_per_ml
Rank 18	Glyoxylate_ratio	Methionine_ratio	Lysophosphatidylcholine_.C20.4._ratio	Palmitic_acid_.C16.0._ratio	Triacylgyceride_hydroperoxide_.C16.0,C18.1,C18.3.OOH._.additi	Lysophosphatidylcholine_.C20.4._ratio
Rank 19	Isoleucine_ratio	Metanephrine_ng_per_ml	Unknown.68100045._ratio	Isoleucine_ratio	Lysophosphatidylcholine_.C20.4._.additional.	Metanephrine_ng_per_ml
Rank 20	Unknown.38100434._ratio	Glucose_ratio	Glycerol__lipid_fraction_ratio	Cholic_acid_ratio	Metanephrine_ng_per_ml	Lysophosphatidylcholine_.C20.4._.additional.

Table 3.4 shows an example of the feature comparison table. The data sets shown in the table is the Metabolite data. I list the top 20 features selected by each feature selection method. The first 10 features obtained from the all-feature Metabolite data shown using into different colors. The features colored as yellow are the ones not included in the reduced Metabolite data set. A red arrow indicates the change of ranking of the top 10 metabolite features between the all-feature Metabolite data set and the reduced Metabolite data set.

In addition, based on the number of features - performance analysis and the feature



(a) Gini Index

(b) Information Gain

(c) Stability Selection

Figure 3.7: Visualization of sample distributions based on the top 2 PCs of Metabolite Data; features are obtained using multiple feature selection methods.

ranking, we can choose a certain number of features to visualize the data. The objective of visualization is to communicate information in the data by the means of graphics [74]. For example, using a principal component analysis (PCA), we can project the data into a $2D$ or $3D$ space.

Figure 3.7 illustrates the sample distributions based on first two PCs obtained from the previous UFSEM framework. The PCA is performed on a list of selected Metabolite features. More especially, features in Fig. 3.7a are picked by the Gini index method; features in Fig. 3.7b are selected by the infomation gain method; and features in Fig. 3.7c are chosen by Stability Selection. It can be observed from the figures that the melancholic depression patients and healthy controls are well separated.

SSLM and Imbalanced Learning

In Section 3.5, I discussed some alternative approaches that can deal with the imbalanced classification problems, that is, the novelty detection methods. Meanwhile, I consider the SSLM approach, which maximizes the margin between the outliers and the normal samples.

I first compare the classification performance of the SVDD method and the SSLM method. The experiment results shown in Fig. 3.8 are obtained from the Metabolite data set and the target here is to identify the minority class of melancholic depression patients from the majority class of healthy controls. Moreover, various training ratios have been tested on the majority class of samples.

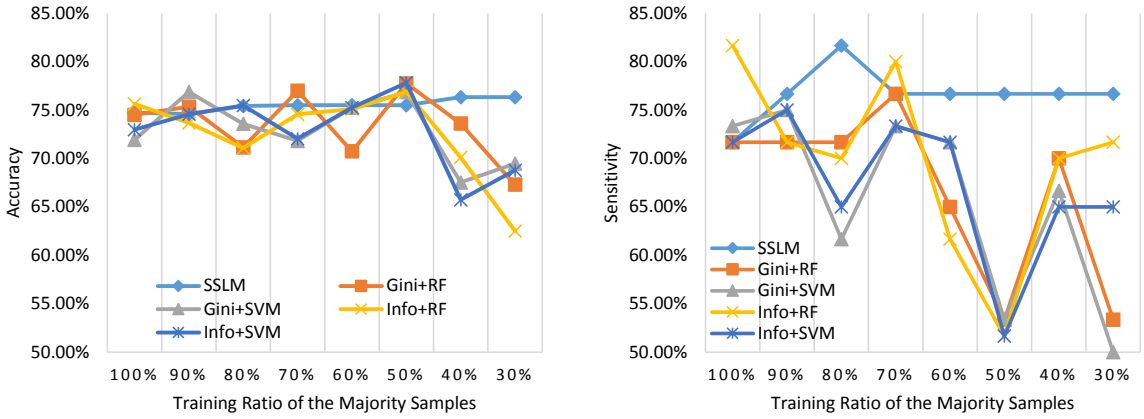
Figure 3.8 illustrates that the SSLM approach provides competitive performance compared to SVDD for most of experimental conditions. In the 70% and 30% positive training ratio cases, the SSML approach is more stable than the other one. The results here are also close to those the UFSEM framework shown in Fig. 3.5a. In addition, the experiment results demonstrate that the SSML can produce satisfactory performance, even

Figure 3.8: Comparison of the melancholic depression classification performance between the SVDD method and the SSLM method on Metabolite data; different training ratios of patients are used.

if we use a small set of samples from the minority class (which will cause the data set extremely imbalanced).

I then compare SSLM with the UFSEM framework. In each cross validation partition, I randomly remove a certain proportion of samples in the minority class, and then do the undersampling. The experiment results shown in Fig. 3.9 summarize the accuracy, the sensitivity and the specificity obtained using different positive training ratios.

It can be observed in Fig. 3.9a that, if we control the minority class training ratio at 90%, the learning accuracies obtained by the UFSEM approach are slightly better than the SSLM method. Figure 3.9b illustrates that the sensitivity drops significantly if we use less than 70% samples from the patients. These results show that, the SSLM approach is effective in the complex imbalanced biomedical learning. Even if the minority set is very small, the SSLM approach can still perform well.



(a) Accuracy

(b) Sensitivity

(c) Specificity

Figure 3.9: Comparison of the melancholic depression classification performance between the SSLM method and the UFSEM framework on Metabolite data; different training ratios of patients are used.

Chapter 4

Clustering Methods in High-Dimensional Learning

Clustering, as an important unsupervised learning method, refers to the procedure of assigning data into groups. A cluster is a subset of data that have a small within-cluster distance and are dissimilar to objects outside the cluster. The criteria that evaluate the similarity or the distance between data points including Euclidean distance, L_1 distance and the correlation *etc*. For a certain application, the appropriate distance criterion should be applied.

Many clustering algorithms have been proposed in the past. These methods can be generally categorized into four types: Exclusive, Overlapping, Hierarchical, and Probabilistic Clustering. Exclusive clustering requires that each data point only belong to a single cluster, while the overlapping approach does not have this restriction. Hierarchical clustering and Probabilistic clustering are based on clusters union and the probabilistic approach, respectively.

The biomedical data often suffer from the curse of dimensionality, and the data may contain many strongly correlated variables. The elimination of these similar variables can reduce the dimensionality of the data and may improve the performance of learning algorithms.

In this chapter, I present some basic clustering algorithms, and then introduce the approaches of using clustering methods in learning from the high-dimensional biomedical data, that is, clustering highly correlated variables in data is followed by further operations.

4.1 K-means Clustering

K-means clustering is one of the most well-known unsupervised learning algorithms for clustering problems. K-means algorithm is an exclusive clustering method that partitions

n observations into k clusters. Each cluster is described by a centroid such that each observation belongs to one cluster with a minimal distance to the corresponding centroid, and the over-all distance is minimized.

Given a dataset $\mathbf{X} = [\mathbf{x_1}, \ldots, \mathbf{x_n}] \in \mathbb{R}^{n \times d}$, we denote

$$\pi_j = \{v \mid x_v \text{ belongs to cluster } j\} \tag{4.1}$$

as a cluster, and thus

$$\Pi = \{\pi_j\}_{j=1}^{k} \tag{4.2}$$

is a partitioning of $\mathbf{X}$, which assigns n samples into k clusters. The centroid of a cluster is defined as

$$c_j = \frac{1}{n_j} \sum_{v \in \pi_j} x_v, \tag{4.3}$$

where n_j is to the number of elements in set π_j. Assume that Euclidean distance measure is used, and then for a certain partitioning Π, the quality of the resulted clustering is evaluated by the sum-of-squares cost function:

$$Q(\Pi) = \sum_{j=1}^{k} \sum_{v \in \pi_j} \|x_v - c_j\|^2. \tag{4.4}$$

In K-means, our target is to minimize the objective function, *i.e.,* $\min Q(\Pi)$.

The K-means clustering algorithm uses an iterative refinement approach that is shown in Algorithm 2.

Algorithm 2 The K-means Clustering Algorithm

Input: $\mathbf{X}, k$

Output: Π

1: Initialization: Pick up k centroids in the objects space.
2: **while** not convergence **do**
3: Assign all data points of $\{x_v\}_{v=1}^{n}$ to their nearest centroid using a certain measurement.
4: Recalculate the centroid of each cluster.
5: **end while**

The choice of the positions of the initial cluster centroids is one of the key steps of the K-means algorithm. An intuitive method is to randomly select k observations from all n samples. Other strategies like Random Partition [75] are also utilized in various applications. In addition, although this procedure usually converges fast, it cannot be guaranteed that the algorithm achieves the global minimum solution. In practice, multiple trials are necessary to obtain an approximately optimal solution. That is, we repeat the K-means clustering algorithm multiple times, at each time, using a new set of initial centroids to evaluate the cost calculated by the cost function. We then choose the best solution from the results.

4.2 Hierarchical Clustering

The objective of hierarchical clustering is to build a hierarchy structure based on the data points. This hierarchy structure is usually represented as a binary tree or a dendrogram of clusters.

Agglomerative hierarchical clustering method is one of the most frequently used approaches, which is shown in Algorithm 3.

This clustering method is a bottom-up monotonic procedure. The hierarchical clustering algorithm can provide the whole tree structure of the objects, and thus it is easy

Algorithm 3 The Agglomerative Hierarchical Clustering Algorithm

Input: $\mathbf{X} = [\mathbf{x_1}, \ldots, \mathbf{x_n}] \in \mathbb{R}^{n \times d}$

Output: The corresponding dendrogram.

1: Initialization: Assigning each sample of $\mathbf{X}$ to a cluster; there are n clusters in total.
2: Compute the distances between every two clusters.
3: **while** exists more than one cluster **do**
4: Find the closest pair of clusters and join them together, *i.e.,* merge them into a new cluster.
5: Compute the distances between the new cluster and the old parts.
6: **end while**

to obtain a certain number of clusters from the dendrogram. It is noteworthy to point out the two aspects in the hierarchical clustering method: one is the select of distance metric, and the other is the linkage strategy.

As mentioned above, there are many distance criteria that can measure the distance between two data points, for example, Euclidean distance and the correlation. The choice of an appropriate distance criterion is very important for a particular application, *e.g.,* using the correlation measuring the similarity among genes or among other biological molecules [6]. Note that, these distances are defined based on individual data points. However, in hierarchical clustering, we also concern the distance between two clusters of data points. Next, I introduce the linkage strategies.

The strategy of linkage is aimed to measure the distance between two clusters of observations. Given two clusters $A = \{a_1, \ldots, a_n\}$ and $B = \{b_1, \ldots, b_m\}$ with n and m elements respectively. The following are some frequently used linkage criteria:

- **Single Linkage**:

$$d(A,B) = \min(dist(a_i, b_j)), 1 \leq i \leq n, 1 \leq j \leq m. \tag{4.5}$$

- **Complete Linkage**:

$$d(A,B) = \max(dist(a_i, b_j)), 1 \leq i \leq n, 1 \leq j \leq m. \tag{4.6}$$

- **Average Linkage**:

$$d(A,B) = \frac{1}{n \times m} \sum_{i=1}^{n} \sum_{j=1}^{m} dist(a_i, b_j). \tag{4.7}$$

Single Linkage uses the minimum distance between elements in two clusters, while Complete Linkage uses the maximum distance. Moreover, Average Linkage use the mean distance between all pair of elements in two clusters. There are also other linkage criteria utilized in practices, *e.g.,* Median Linkage, Centroid Linkage and Wards Linkage.

Different linkage strategies will lead to various hierarchical structures. For example, Complete Linkage will bring a lot of compact clusters with similar diameters, and Single Linkage method will result in the chaining phenomenon [76].

4.3 Clustering Methods in Features

Different from the traditional clustering methods that find a partitioning in samples, I focus on the organization of variables. In learning from high-dimensional biomedical data, traditional learning algorithms may not perform well due to the high dimensionality.

My current approach is to process clustering method on the feature dimension, that is, to cluster highly correlated features in data. Consider a data set $\mathbf{X} \in \mathbb{R}^{n \times d}$, where n is the number of samples and d is the number of features. The target here is to partition d features into k clusters. Before clustering, we require the data to be centered and scaled, *i.e.,* with mean zero and standard deviation one for each feature. I then apply the clustering method to group the similar features into a cluster and keep the dissimilar variables away from each other.

There will be many advantages brought from the strategy that clustering variables in high-dimensional data set:

- Clustering method reduces the data dimension. After clustering, we can represent the dataset using k cluster centroids, which is much smaller than the original feature dimension. Thereby, the clustering method can further reduce the complexity of learning task.
- The combination of highly correlated features can achieve a more reliable learning result. For instance, in the regression tasks, those highly correlated variables will lead to the multicollinearity, and then cause the inaccurate estimation of regression [77].

- Some learning algorithms are insensitive to the correlation structures among the variables, so that the information store in those redundant features will be omitted by the learning systems [6]. For example, Lasso is designed to select only one variable from a group of correlated variables. Therefore, even if a variable is useful, it still may be ignored in the scenario of data with highly correlated variables.

To sum up, applying clustering methods on feature space can reduce the dimensionality of data, and meanwhile, improve the reliability, the performance and the efficiency of learning systems. Moreover, clustering variables in high-dimensional biomedical data have biological interpretability.

4.4 Experiments

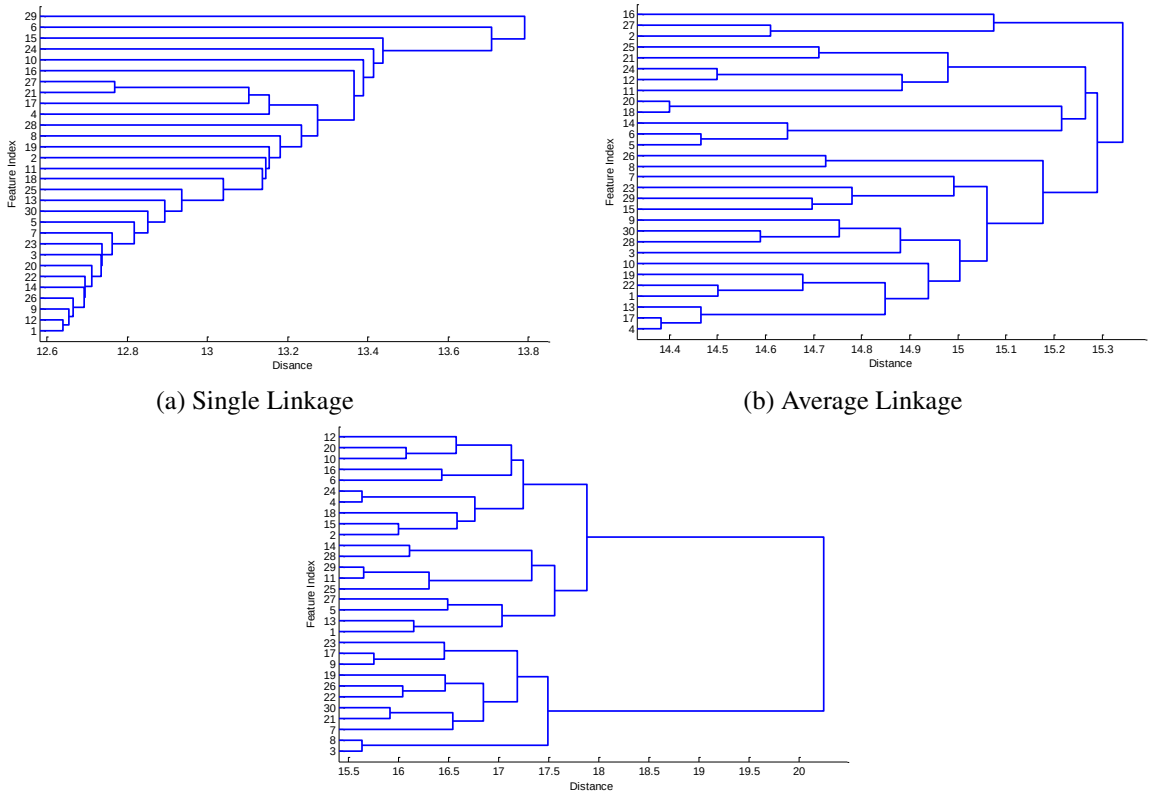
In this section, I provide several experiments of using clustering methods on the Metabolite data in the Depression research. These experiments include: (1) the comparison of different linkage strategies for the hierarchical clustering (2) the classification performance related to the data with clustered variables.

Comparison among Different Linkage Strategies

As mentioned in Section 4.2, the linkage strategies are used to measure the distance between two clusters of observation. By using different linkage strategies, we will obtain various cluster structures for hierarchical clustering.

I tested Single Linkage, Average Linkage and Complete Linkage on the Depression Metabolite data - MelanDprs-C target. Figure 4.1 illustrates the cluster structures by dendrograms. Each dendrogram is obtained from a hierarchical clustering with a certain linkage strategy. The clusters given in Fig. 4.1a show an approximately chaining structure, which is caused by the single linkage strategy. Figure. 4.1b illustrates the result of the average linkage strategy and the cluster structure is expressed as the

combination of several subchaings. Moreover, it is clear that the complete linkage method, shown in Fig. 4.1c, brings many compact clusters with similar diameters.



(a) Single Linkage

(b) Average Linkage

(c) Complete Linkage

Figure 4.1: Dendrograms of hierarchical clusters, based on different linkage strategies on the Depression Metabolite data; each dendrogram is built base on top 30 levels of the hierarchical tree and the distance criterion is the correlation.

Recall that in the high-dimensional biomedical data learning, clustering is aimed to group highly correlated variables among the original data. Consider the experiment results shown in Fig. 4.1, if we choose a cutoff at a certain level of the hierarchical tree, the single linkage is more likely to generate a cluster with a large number of features and the remain clusters are both single ones. However, the clusters obtained via the complete linkage will contains several features. This method is known as the farthest neighbour clustering. Moreover, the average linkage can be considered as an intermediate state of the

single linkage strategy and the complete linkage strategy. Therefore, I conclude that the complete linkage is more suitable for our need.

In the experiments, I cluster 270 metabolites into 100 groups. The cluster statistics of the usage of different linkage strategies are shown in Table 4.1. Compared with the single linkage and the average linkage, the complete linkage strategy result in the lowest standard deviation (SD), the smallest maximum cluster size, as well as the minimum number of single clusters. Table 4.1 also verifies that the complete linkage clustering is the most satisfactory method. Therefore, these results imply that the choice of the Complete Linkage strategy may be a proper one for the agglomerative hierarchical clustering on the biomedical data.

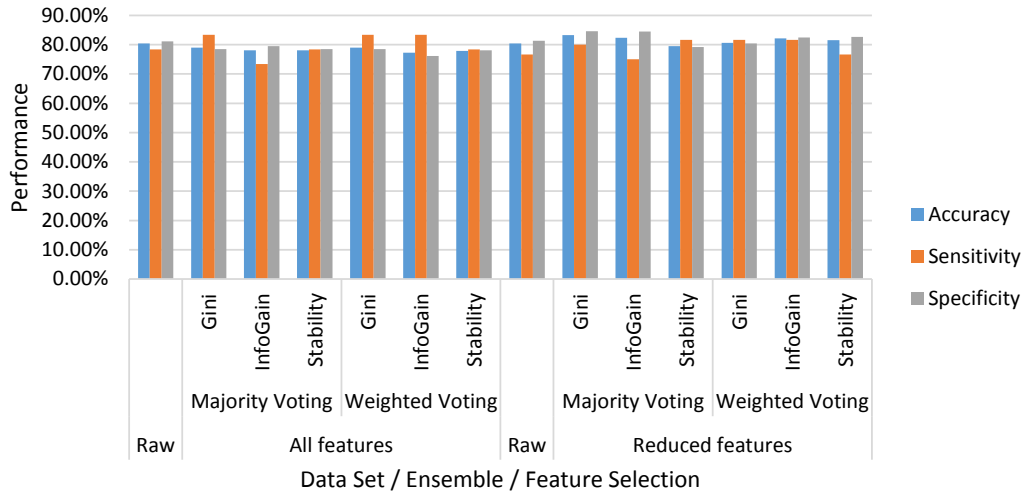
Table 4.1: The statistics of clusters among different linkage strategies on the Depression Metabolite data set.

	SD	Max size of a cluster	Number of single clusters
Single Linkage	14.19	143	80
Average Linkage	4.15	36	45
Complete Linkage	2.76	23	30

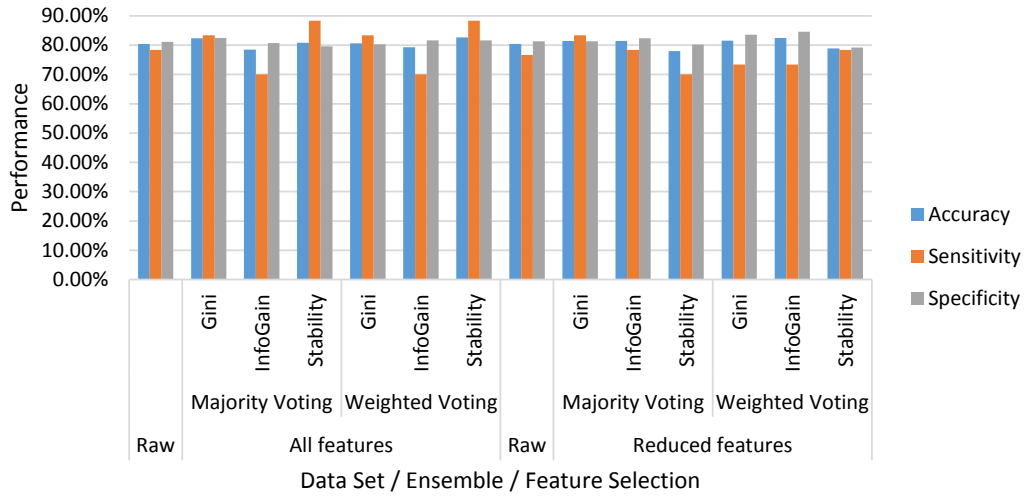
Clustering Methods and High-Dimensional Learning

In the learning process of the high-dimensional biomedical data set, the objectives of the clustering methods on the feature space are as follows: (1) to reduce dimension; (2) to group highly correlated variables; and (3) to select related features together. After obtaining a clustered data set, the next step is to check its usability in the learning tasks.

The experiments shown in Fig. 4.2 are based on the Metabolite data. The learning target here, again, is to identify the melancholic depression patients. For the all-feature Metabolite data and the reduced Metabolite data, I built 100 clusters on each data set via the K-means clustering method and the hierarchical clustering method separately. I then use the UFSEM framework to learn each clustered data set.



(a) K-means clustering



(b) Hierarchical clustering

Figure 4.2: Comparison of the melancholic depression classification performance on clustered Metabolite data.

Figure 4.2a illustrates the classification results based on the K-means clustering. It is observed that the clustered all-feature Metabolite data produce similar performance compared with those shown in Fig. 3.4a. The sensitivities improved via the Gini index in both ensemble strategies. For the clustered reduced Metabolite data, the Gini index method also performs 3% better than those in Fig. 3.4a. Figure 4.2b summarizes the results of the hierarchical clustering. The stability selection method improves the

sensitivities on the all-feature Metabolite data by 10%. The Gini index method and the Stability selection method also provide competitive performance on the reduced Metabolite data. Note that, both clustering methods bring better sensitivity scores in the learning tasks, which implies that the classifiers can better identify the patients based on these clustered data sets.

Therefore, the experiments results imply that the combination of clustering methods and the UFSEM framework can provide satisfactory performance in the high-dimensional imbalanced biomedical data learning.

Chapter 5

Conclusion and Outlook

5.1 Summary of Conclusions

The major objective of this thesis is to study some machine learning techniques that can be used to learn the high-dimensional imbalanced biomedical data sets. There are mainly three issues in the learning tasks: (1) how to identify the patterns among multiple data sets and class labels; (2) how to learn from a imbalanced data set; (3) how to improve the performance in the high-dimensional learning.

Firstly, I discuss the inherent characteristics of the biomedical data and reveal several the challenges in the learning process for the high-dimensional imbalanced biomedical data. I then address the research targets and analyze the related works.

The sparse canonical correlation analysis method is the principal topic in the second part. I first present the basic idea of correlation analysis and the usage of CCA method in multidimensional variables. To deal with high-dimensional data, a penalized CCA approach is presented. The choice of an appropriate penalty function and the corresponding penalty constraints can yield the sparse solutions for the canonical vectors. I then demonstrate how to use the sparse CCA to detect the patterns among a set of high-dimensional data sets and the class labels. The classification experiment shows positive results that the signal detected by the sparse CCA method is significant. Some experiments also indicate that the sparse CCA can be used to find patterns between two data sources.

Next, I address the challenges in imbalanced learning. In addition to accuracy, I discuss some alternative criteria to evaluate the learning performance, *e.g.,* sensitivity and specificity. It is important to obtain the accurate models from both classes of samples in an imbalanced learning task. To deal with the imbalanced class distributions, I present the

undersampling method and combine it with the ensemble learning idea. In order to obtain significant features, and improve the learning efficiency, I introduce multiple feature selection methods and apply them to the previous approach. The UFSEM framework, proposed in this thesis, is an effective method to learn from the imbalanced data sets. Moreover, the small sphere and large margin approach is discussed as an alternative method for the imbalanced classification tasks. In practice, both the UFSEM framework and the SSLM method show satisfactory performance in learning from the imbalanced biomedical data.

The last part summarizes the approaches of using clustering methods in dealing with high-dimensional data. To deal with the data containing many highly correlated variables, I employ the K-means clustering method and the hierarchical clustering method in the feature dimension. Experiments demonstrate significant improvements by clustering the highly correlated features in the data.

5.2 Future Works

There are several directions that can be explored in the future work. For example, the sparse CCA method can be used to study multiple (three or more) data sets. Recently, scientists are interested in taking advantages of multiple data sources and expect that interesting patterns can be detected between the data sets. In addition, I also test some ideas of using sparse CCA as a feature selection method in the UFSEM framework. We can further employ other feature selection methods and different classifiers in the UFSEM framework. Currently, I apply a very simple ensemble methods, we plan to explore other more advanced ensemble methods in the future work [78].

REFERENCES

[1] Y. Zhang and J. C. Rajapakse, *Machine learning in bioinformatics*, vol. 4. Wiley. com, 2009.

[2] P. Larrañaga, B. Calvo, R. Santana, C. Bielza, J. Galdiano, I. Inza, J. A. Lozano, R. Armañanzas, G. Santafé, A. Pérez, *et al.*, "Machine learning in bioinformatics," *Briefings in bioinformatics*, vol. 7, no. 1, pp. 86–112, 2006.

[3] I. Inza, B. Calvo, R. Armañanzas, E. Bengoetxea, P. Larrañaga, and J. A. Lozano, "Machine learning: an indispensable tool in bioinformatics," in *Bioinformatics methods in clinical research*, pp. 25–48, Springer, 2010.

[4] H. He and E. A. Garcia, "Learning from imbalanced data," *Knowledge and Data Engineering, IEEE Transactions on*, vol. 21, no. 9, pp. 1263–1284, 2009.

[5] R. L. Somorjai, B. Dolenko, and R. Baumgartner, "Class prediction and discovery using gene microarray and proteomics mass spectroscopy data: curses, caveats, cautions," *Bioinformatics*, vol. 19, no. 12, pp. 1484–1491, 2003.

[6] P. Bühlmann, P. Rütimann, S. van de Geer, and C.-H. Zhang, "Correlated variables in regression: clustering and sparse estimation," *Journal of Statistical Planning and Inference*, 2013.

[7] H. Hotelling, "Relations between two sets of variates," *Biometrika*, vol. 28, no. 3/4, pp. 321–377, 1936.

[8] R. Tibshirani, "Regression shrinkage and selection via the lasso," *Journal of the Royal Statistical Society. Series B (Methodological)*, pp. 267–288, 1996.

[9] L. Sun, S. Ji, and J. Ye, "A least squares formulation for canonical correlation analysis," in *Proceedings of the 25th international conference on Machine learning*, pp. 1024–1031, ACM, 2008.

[10] L. Sun, S. Ji, and J. Ye, "Canonical correlation analysis for multilabel classification: a least-squares formulation, extensions, and analysis," *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 33, no. 1, pp. 194–200, 2011.

[11] D. M. Witten, R. Tibshirani, *et al.*, "Extensions of sparse canonical correlation analysis with applications to genomic data," *Statistical applications in genetics and molecular biology*, vol. 8, no. 1, pp. 1–27, 2009.

[12] S. Waaijenborg, P. C. Verselewel de Witt Hamer, and A. H. Zwinderman, "Quantifying the association between gene expressions and dna-markers by penalized canonical correlation analysis," *Statistical Applications in Genetics and Molecular Biology*, vol. 7, no. 1, 2008.

[13] S. Waaijenborg and A. H. Zwinderman, "Penalized canonical correlation analysis to quantify the association between gene expression and dna markers," in *BMC proceedings*, vol. 1, p. S122, BioMed Central Ltd, 2007.

[14] A. Wiesel, M. Kliger, and A. O. Hero III, "A greedy approach to sparse canonical correlation analysis," *arXiv preprint arXiv:0801.2748*, 2008.

[15] D. M. Witten, R. Tibshirani, and T. Hastie, "A penalized matrix decomposition, with applications to sparse principal components and canonical correlation analysis," *Biostatistics*, vol. 10, no. 3, pp. 515–534, 2009.

[16] Y. Yamanishi, J.-P. Vert, A. Nakaya, and M. Kanehisa, "Extraction of correlated gene clusters from multiple genomic data by generalized kernel canonical correlation analysis," *Bioinformatics*, vol. 19, no. suppl 1, pp. i323–i330, 2003.

[17] F. R. Bach and M. I. Jordan, "A probabilistic interpretation of canonical correlation analysis," 2005.

[18] C. Fyfe and G. Leen, "Two methods for sparsifying probabilistic canonical correlation analysis," in *Neural Information Processing*, pp. 361–370, Springer, 2006.

[19] P. Rai and H. Daumé III, "Multi-label prediction via sparse infinite cca," *Advances in Neural Information Processing Systems*, vol. 22, pp. 1518–1526, 2009.

[20] C. Drummond, R. C. Holte, *et al.*, "C4. 5, class imbalance, and cost sensitivity: why under-sampling beats over-sampling," in *Workshop on Learning from Imbalanced Datasets II*, vol. 11, Citeseer, 2003.

[21] N. Japkowicz *et al.*, "Learning from imbalanced data sets: a comparison of various strategies," in *AAAI workshop on learning from imbalanced data sets*, vol. 68, 2000.

[22] X.-Y. Liu, J. Wu, and Z.-H. Zhou, "Exploratory undersampling for class-imbalance learning," *Systems, Man, and Cybernetics, Part B: Cybernetics, IEEE Transactions on*, vol. 39, no. 2, pp. 539–550, 2009.

[23] N. V. Chawla, K. W. Bowyer, L. O. Hall, and W. P. Kegelmeyer, "Smote: synthetic minority over-sampling technique," *arXiv preprint arXiv:1106.1813*, 2011.

[24] I. Tomek, "Two modifications of cnn," *IEEE Trans. Syst. Man Cybern.*, vol. 6, pp. 769–772, 1976.

[25] T. Jo and N. Japkowicz, "Class imbalances versus small disjuncts," *ACM SIGKDD Explorations Newsletter*, vol. 6, no. 1, pp. 40–49, 2004.

[26] C. Elkan, "The foundations of cost-sensitive learning," in *International joint conference on artificial intelligence*, vol. 17, pp. 973–978, Citeseer, 2001.

[27] K. M. Ting, "An instance-weighting method to induce cost-sensitive trees," *Knowledge and Data Engineering, IEEE Transactions on*, vol. 14, no. 3, pp. 659–665, 2002.

[28] G. Wu and E. Y. Chang, "Class-boundary alignment for imbalanced dataset learning," in *ICML 2003 workshop on learning from imbalanced data sets II, Washington, DC*, pp. 49–56, 2003.

[29] S. Ertekin, J. Huang, L. Bottou, and L. Giles, "Learning on the border: active learning in imbalanced data classification," in *Proceedings of the sixteenth ACM conference on Conference on information and knowledge management*, pp. 127–136, ACM, 2007.

[30] I. Guyon and A. Elisseeff, "An introduction to variable and feature selection," *The Journal of Machine Learning Research*, vol. 3, pp. 1157–1182, 2003.

[31] Y. Saeys, I. Inza, and P. Larrañaga, "A review of feature selection techniques in bioinformatics," *bioinformatics*, vol. 23, no. 19, pp. 2507–2517, 2007.

[32] K. Pearson, "Mathematical contributions to the theory of evolution. iii. regression, heredity, and panmixia," *Philosophical Transactions of the Royal Society of London. Series A, Containing Papers of a Mathematical or Physical Character*, vol. 187, pp. 253–318, 1896.

[33] D. R. Hardoon, S. Szedmak, and J. Shawe-Taylor, "Canonical correlation analysis: An overview with application to learning methods," *Neural Computation*, vol. 16, no. 12, pp. 2639–2664, 2004.

[34] R. A. Johnson and D. W. Wichern, *Applied multivariate statistical analysis*, vol. 5. Prentice hall Upper Saddle River, NJ, 2002.

[35] S. Dudoit, J. Fridlyand, and T. P. Speed, "Comparison of discrimination methods for the classification of tumors using gene expression data," *Journal of the American statistical association*, vol. 97, no. 457, pp. 77–87, 2002.

[36] R. Tibshirani, T. Hastie, B. Narasimhan, and G. Chu, "Class prediction by nearest shrunken centroids, with applications to dna microarrays," *Statistical Science*, pp. 104–117, 2003.

[37] M. v. O. Marina Marcus, M. Taghi Yasamy and S. S. Dan Chisholm, "Depression: A global public health concern," *World Health Organization paper written for the World Federation of Mental Health*, 2012.

[38] L. Brain Resource Company, Johnson & Johnson Pharmacy Research & Development, "An integrative approach for the detection of biomarkers in major depression," *BRC / J&J PRD Biomarker Depression Trial*.

[39] T. Schneider, "Analysis of incomplete climate data: Estimation of mean values and covariance matrices and imputation of missing values," *Journal of Climate*, vol. 14, no. 5, pp. 853–871, 2001.

[40] D. Albrecht, O. Kniemeyer, A. A. Brakhage, and R. Guthke, "Missing values in gel-based proteomics," *Proteomics*, vol. 10, no. 6, pp. 1202–1211, 2010.

[41] T. Speed, *Statistical analysis of gene expression microarray data*, vol. 11. Chapman and Hall/CRC, 2003.

[42] T. Hastie, R. Tibshirani, G. Sherlock, M. Eisen, P. Brown, and D. Botstein, "Imputing missing data for gene expression arrays," 1999.

[43] O. Troyanskaya, M. Cantor, G. Sherlock, P. Brown, T. Hastie, R. Tibshirani, D. Botstein, and R. B. Altman, "Missing value estimation methods for dna microarrays," *Bioinformatics*, vol. 17, no. 6, pp. 520–525, 2001.

[44] L. Breiman, "Random forests," *Machine learning*, vol. 45, no. 1, pp. 5–32, 2001.

[45] C.-C. Chang and C.-J. Lin, "LIBSVM: A library for support vector machines," *ACM Transactions on Intelligent Systems and Technology*, vol. 2, pp. 27:1–27:27, 2011.

[46] L. Velayudhan, P. Proitsi, E. Westman, J.-S. Muehlboeck, P. Mecocci, B. Vellas, M. Tsolaki, I. Kłoszewska, H. Soininen, C. Spenger, *et al.*, "Entorhinal cortex thickness predicts cognitive decline in alzheimer's disease," *Journal of Alzheimer's Disease*, vol. 33, no. 3, pp. 755–766, 2013.

[47] X. Li, J. Jiao, S. Shimizu, I. Jibiki, K.-i. Watanabe, and T. Kubota, "Correlations between atrophy of the entorhinal cortex and cognitive function in patients with alzheimer's disease and mild cognitive impairment," *Psychiatry and clinical neurosciences*, vol. 66, no. 7, pp. 587–593, 2012.

[48] J. H. Morra, Z. Tu, L. G. Apostolova, A. E. Green, C. Avedissian, S. K. Madsen, N. Parikshak, X. Hua, A. W. Toga, C. R. Jack, *et al.*, "Automated 3d mapping of hippocampal atrophy and its clinical correlates in 400 subjects with alzheimer's disease, mild cognitive impairment, and elderly controls," *Human brain mapping*, vol. 30, no. 9, pp. 2766–2788, 2009.

[49] K. Nho, S. L. Risacher, P. K. Crane, C. DeCarli, M. M. Glymour, C. Habeck, S. Kim, G. J. Lee, E. Mormino, S. Mukherjee, *et al.*, "Voxel and surface-based topography of memory and executive deficits in mild cognitive impairment and alzheimers disease," *Brain imaging and behavior*, vol. 6, no. 4, pp. 551–567, 2012.

[50] G. M. Weiss, "Mining with rarity: a unifying framework," *ACM SIGKDD Explorations Newsletter*, vol. 6, no. 1, pp. 7–19, 2004.

[51] R. Caruana, "Learning from imbalanced data: Rank metrics and extra tasks," in *Proc. Am. Assoc. for Artificial Intelligence (AAAI) Conf*, pp. 51–57, 2000.

[52] R. Akbani, S. Kwek, and N. Japkowicz, "Applying support vector machines to imbalanced datasets," in *Machine Learning: ECML 2004*, pp. 39–50, Springer, 2004.

[53] K. Ron and P. Foster, "Glossary of terms: Special issue on applications of machine learning and the knowledge discovery process," *Machine Learning*, vol. 30, pp. 271–274, 1998.

[54] T. Fawcett, "An introduction to roc analysis," *Pattern recognition letters*, vol. 27, no. 8, pp. 861–874, 2006.

[55] C. D. Manning, P. Raghavan, and H. Schütze, *Introduction to information retrieval*, vol. 1. Cambridge University Press Cambridge, 2008.

[56] M. Kubat, R. C. Holte, and S. Matwin, "Machine learning for the detection of oil spills in satellite radar images," *Machine learning*, vol. 30, no. 2-3, pp. 195–215, 1998.

[57] D. M. Powers, "Evaluation: From precision, recall and f-factor to roc, informedness, markedness & correlation," *School of Informatics and Engineering, Flinders University, Adelaide, Australia, Tech. Rep. SIE-07-001*, 2007.

[58] R. C. Holte, L. E. Acker, and B. W. Porter, "Concept learning and the problem of small disjuncts," in *Proceedings of the Eleventh International Joint Conference on Artificial Intelligence*, vol. 1, Citeseer, 1989.

[59] R. Maclin and D. Opitz, "Popular ensemble methods: An empirical study," *arXiv preprint arXiv:1106.0257*, 2011.

[60] R. Polikar, "Ensemble based systems in decision making," *Circuits and Systems Magazine, IEEE*, vol. 6, no. 3, pp. 21–45, 2006.

[61] L. Rokach, "Ensemble-based classifiers," *Artificial Intelligence Review*, vol. 33, no. 1-2, pp. 1–39, 2010.

[62] T. G. Dietterich, "Ensemble methods in machine learning," in *Multiple classifier systems*, pp. 1–15, Springer, 2000.

[63] T. Hastie, R. Tibshirani, and J. J. H. Friedman, *The elements of statistical learning: data mining, inference, and prediction*, vol. 1. Springer New York, 2009.

[64] Z. Zhao, F. Morstatter, S. Sharma, S. Alelyani, A. Anand, and H. Liu, "Advancing feature selection research," *ASU Feature Selection Repository*, 2010.

[65] C. Gini, "Variabilità e mutabilità," *Reprinted in Memorie di metodologica statistica (Ed. Pizetti E, Salvemini, T). Rome: Libreria Eredi Virgilio Veschi*, vol. 1, 1912.

[66] S. R. Singh, H. A. Murthy, and T. A. Gonsalves, "Feature selection for text classification based on gini coefficient of inequality.," *Journal of Machine Learning Research-Proceedings Track*, vol. 10, pp. 76–85, 2010.

[67] Y. Yang and J. O. Pedersen, "A comparative study on feature selection in text categorization," in *ICML*, vol. 97, pp. 412–420, 1997.

[68] N. Meinshausen and P. Bühlmann, "Stability selection," *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, vol. 72, no. 4, pp. 417–473, 2010.

[69] Y. L. J. H. Jiayu Zhou, Jimeng Sun and J. Ye, "Patient risk prediction model via top-k stability selection," *SIAM Conference on Data Mining*, 2013.

[70] J. Liu, S. Ji, and J. Ye, "Slep: Sparse learning with efficient projections," 2009.

[71] V. Chandola, A. Banerjee, and V. Kumar, "Anomaly detection: A survey," *ACM Computing Surveys (CSUR)*, vol. 41, no. 3, p. 15, 2009.

[72] D. M. Tax and R. P. Duin, "Support vector data description," *Machine learning*, vol. 54, no. 1, pp. 45–66, 2004.

[73] M. Wu and J. Ye, "A small sphere and large margin approach for novelty detection using training data with outliers," *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 31, no. 11, pp. 2088–2092, 2009.

[74] V. Friedman, "Data visualization and infographics," *Graphics, Monday Inspiration*, vol. 14, p. 2008, 2008.

[75] G. Hamerly and C. Elkan, "Alternatives to the k-means algorithm that find better clusterings," in *Proceedings of the eleventh international conference on Information and knowledge management*, pp. 600–607, ACM, 2002.

[76] B. S. Everitt, S. Landau, M. Leese, and D. Stahl, "Hierarchical clustering," *Cluster Analysis, 5th Edition*, pp. 71–110, 2001.

[77] D. E. Farrar and R. R. Glauber, "Multicollinearity in regression analysis: The problem revisited," *The Review of Economics and Statistics*, vol. 49, no. 1, pp. 92–107, 1967.

[78] Z.-H. Zhou, *Ensemble methods: foundations and algorithms*. CRC Press, 2012.

[79] J. Dukart, M. L. Schroeter, and K. Mueller, "Age correction in dementia–matching to a healthy brain," *PloS one*, vol. 6, no. 7, p. e22193, 2011.

[80] J. Trygg and S. Wold, "Orthogonal projections to latent structures (o-pls)," *Journal of Chemometrics*, vol. 16, no. 3, pp. 119–128, 2002.

APPENDIX A

Reduce Storage Time Confounder in the Metabolite Data

The previous works have pointed out that, in the Depression research, the Metabolite data suffer from the confounding effect of storage time.

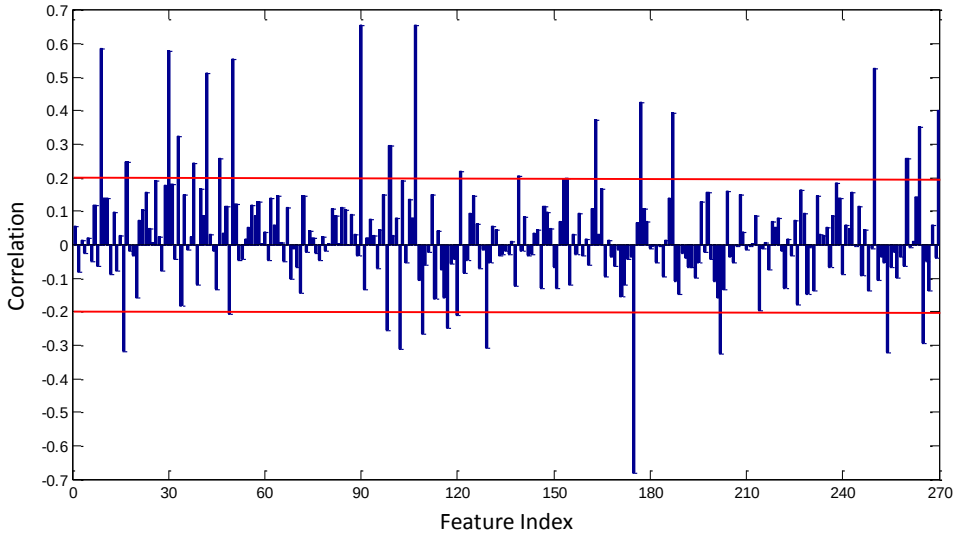


Figure A.1: The pairwise linear correlation coefficient between each metabolite and the storage time at the Depression, use all valid samples and impute missing values via KNN.

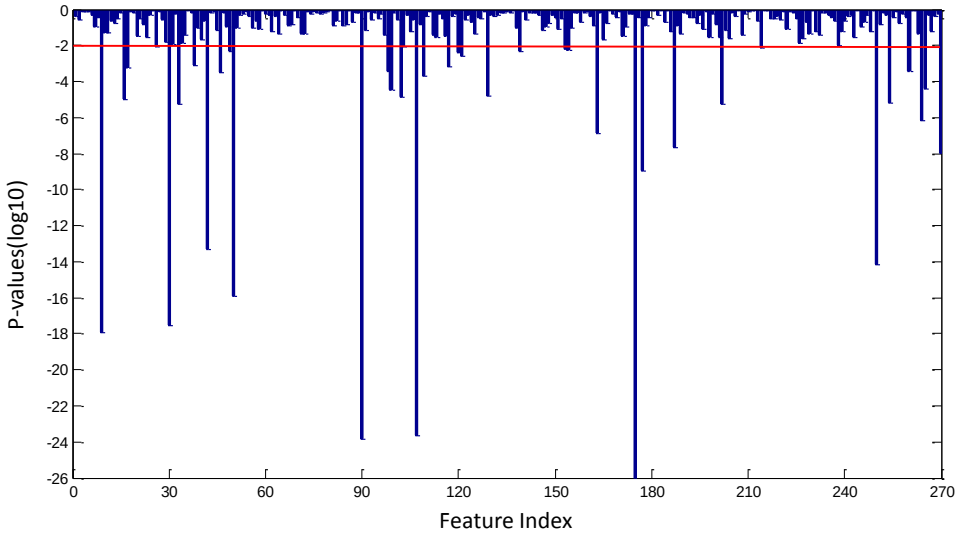


Figure A.2: The p-values for testing the hypothesis of no correlation against the alternative that there is a nonzero correlation for each pair of metabolite and the storage time at the Depression, use log_{10} transformation, use all valid samples and impute missing values via KNN.

Figures A.1 & A.2 illustrate the experiment results of the pairwise correlation test between each metabolite feature and the storage time at the Depression. We consider that a significant correlation will result in the correlation $|\rho| > 0.2$ or $p < 0.01$ (*i.e.,* $\log_{10}(p) < -2$). Since the plasma samples were stored at -20°, the concentration levels of a large number of metabolites will be strongly affected by the storage time duration.

It has been detected that, for most of metabolites, the relation between the metabolite concentration with storage time at the Depression is relatively linear within 200 days. In order to reduce this storage time confounder, I first remove two types of samples: the one is the samples that are stored longer than 200 days; and the other is the samples that failure in quality control. I then correct the storage time confounder by taking the residuals of a linear regression line of storage time for each metabolite separately.

In this thesis,I correct the Metabolite data in two ways: one is based on all samples; and the other is based on healthy controls (HCs). Detailed procedures are described below.

Let X be one column of original metabolite feature, and T be the column vector of the storage time at the Depression for the corresponding samples (all valid samples included). Then, the metabolite feature could be corrected in the following two ways:

- **Correct the storage time confounder based on all samples**

  Because the storage time duration has a linear confounding effect to most of the metabolites, we can obtain the linear regression model:

$$X = T\beta_1 + \beta_0, \tag{A.1}$$

  where β_1 stands for the effect of storage time T on X and β_0 is the bias. Problem (A.1) can be solved by the least square estimation as:

$$\begin{pmatrix} \beta_0 \\ \beta_1 \end{pmatrix} = (S^T S)^{-1} S^T X, \tag{A.2}$$

where S is a two-column matrix where the entries of the first column are all one and the entries of the second column are evaluated as T. Once β_1 is determined, the corrected metabolite feature X_c is:

$$X_c = X - T\beta_1. \tag{A.3}$$

- **Correct the storage time confounder based on healthy controls**

  Recent studies [79] proposed a similar correction method but only use the information from healthy controls. To fit the linear model built on only healthy controls, we would replace criterion (A.1) as:

$$X_{HCs} = T_{HCs}\beta_1 + \beta_0, \tag{A.4}$$

  where X_{HCs} and T_{HCs} correspond to the healthy controls' metabolite feature column and the storage time vector, respectively. Other steps remain the same.

APPENDIX B

Feature Evaluation and Removal in the Metabolites Data Set

As discussed in Appendix 5.2, the Depression Metabolite data suffer from the confounding effect of storage time. To reduce this time confounder, I apply two correction algorithms in the practice. However, the correction methods can not eliminate this inherent effect in both metabolites. Therefore, in this appendix, I propose another approach that is to remove some unstable metabolite features in the original Metabolite data set.

This work is base on a previous O-PLS[1] test. We detect 72 metabolite features that are sensitive to the storage time via O-PLS. These features can be categorized into four groups according to their sensitivities to the storage time duration and the O-PLS loading values: (1) highly sensitive & increase; (2) highly sensitive & decrease; (3) sensitive & increase; (4) sensitive & decrease. We then compare these O-PLS test results with the original metabolite concentrations and obtain 44 metabolites shown in Table B.1. Those metabolites are sensitive to the storage time and meanwhile, the concentrations changing tends of those features are varied between the melancholic depression patients and healthy controls.

To further reduce the time confounder, I consider remove those features from the original set. However, it is clear that removing those features may affect the learning performance. Compared with the feature rankings obtained from the UFSEM framework (see Table 3.4), some features in Table B.1 are very significant to the learning system. For these potentially important metabolites, I then apply some further analysis to explore the features such as t-test, correlation analysis and visualization method. Eventually, I keep 2 of them in the reduced Metabolite data set and remove the other 42 metabolites.

[1]O-PLS: Orthogonal Projection to Latent Structures Algorithm [80]. The O-PLS test is implemented to predict storage time at -20C loading.

Table B.1: The list of abnormal metabolites and their O-PLS test result. The texts in bold face are considered to be kept; others are removed in the reduced Metabolite data set.

O-PLS Test	Metabolite name
highly sensitive	Glutamine
& decrease	3 4 Dihydroxyphenylacetic acid
	3 4 Dihydroxyphenylalanine DOPA
	3 4 Dihydroxyphenylglycol
	Adrenaline
	Noradrenaline
	beta Carotene
highly sensitive	Aspartate
& increase	Glutamate
	Triacylgyceride hydroperoxide C16 0 C18 1 C18 2 OOH
	Triacylgyceride hydroperoxide C16 0 C18 1 C18 3 OOH additional Triacylgyceride hydroperoxide C16 0 C18 2 C18 2 OOH
	Triacylgyceride hydroperoxide C18 1 18 2 C18 2 OOH additional Triacylgyceride hydroperoxide C16 0 C18 1 C20 4 OOH Triacylgyceride hydroperoxide C18 1 C18 1 C18 3 OOH
	Azelaic acid
	Unknown 68100024
	Unknown 68100033
	Unknown 68100060
	Unknown 68100426
	Unknown 68100427
	Unknown 68100434
	Unknown 68100437
sensitive	Taurine
& decrease	Cryptoxanthin
	1 2 Dioleoyl glycero 3 phosphatidylserine
	1 Octadecenyl 2 arachidonoylglycero 3 phosphocholine Plasmalogen
	Phosphatidylcholine 13
	Phosphatidylcholine 3
	Phosphatidylcholine C16 0 C22 6 or C18 2 C20 4
	Phosphatidylcholine C18 0 C22 6
	Sphingomyelin
	Unknown 38100405
	Unknown 38100434
	Unknown 38100438
	Unknown 38100474
	Unknown 58100143
	Unknown 68100428
	Unknown 68100430
	Unknown 68100433
sensitivee	Serine
& increas	Melissic acid C30 0
	3 3' 5 Triiodo L thyronine
	Lysophosphatidylethanolamine
	Unknown 58100019
	Unknown 58100144
	Unknown 58100156