Imperial College London
Department of Computing

Machine Learning for the Classification of Dementia in the Presence of Mis-labelled Data

by

Mr. Liang Chen

Supervised by Prof. Daniel Rueckert

Submitted in part fulfilment of the requirements for
the MSc Degree in Advanced Computing of Imperial College London

September 2013

Acknowledgements

Firstly, I would like to appreciate my supervisor, Prof. Daniel Rueckert. It is sure that without his supervision, there will not be this work. He guided the development of this work and proposed many valuable suggestions.

Secondly, I want to thank Dr. Katherine R. Gray and Dr. Ricardo G. Moreno, both of whom are post-docs in BioMedIA Group. They not only provide me with all the data and related information in this work. They participated all the discussion and proposed invaluable advice. In addition, they spent large amount of time in evaluating my results and commenting on my report. I am also grateful to Tong Tong, who is a PhD candidate in BioMedIA Group. He helped me get started of multiple instance learning.

Last but not least, I feel like expressing my gratitude to the whole of my family. Although they are in the far east, they support my study at UK firmly and they are always with me. Thanks also to all of my friends in UK and China.

Abstract

Diagnosis of Alzheimer's disease is a global problem. Automated image-based classification for individual patients plays a pivotal role in computer-assisted diagnosis. The Alzheimer's Disease Neuroimaging Initiative provides a large amount of images for researchers to find approaches to help clinicians with their diagnosis. In this work, magnetic resonance images from it is taken into account. However, there might be a small number of mistakes in the diagnostic labels. This work tries to find them out via machine learning techniques.

This work includes a review of many recent papers about automated diagnosis of Alzheimer's disease based on machine learning. We therefore did large-scale statistics on the data based on voxels in the whole brain and around the hippocampus. Several data which is believed as classified correctly was pick up. The best classification accuracy we achieved is 86.77%, which is between ADs and controls on voxels around hippocampus. Our best accuracy is comparable to other published studies. We also assessed the quality of different clinical sites, discovering that almost all clinical sites provided equally reliable imaging data and diagnostic labels.

This paper then developed a framework of multiple instance learning to identify potentially mis-labelled data. We used volumes of hippocampus as the feature in this part. In the first stage, we generated the synthetic data according to the distribution of real data to test and improve the model. Then we applied the model to the real data to unearth the potential mis-classified individuals.

In conclusion, the results suggest that the model in this work could be applied to support the clinical diagnosis by helping to identify labelling errors. Future work may contribute to simplify the whole procedure, improve the robustness of the algorithms, and it may also consider different modalities, including PET imaging.

Contents

List of Tables

List of Figures

1 Introduction

1.1 Alzheimer's Disease

Alzheimer's disease (AD) is a common type of dementia that occurs in the middle-aged and elderly population. The progressive dementia may result in a brain disorder. If one suffers from dementia, their memory and other cognitive functions would decline when compared to their previous level of function. AD is partly diagnosed by a history of decline in performance and by abnormalities noted from clinical examination and psychological tests. Presence of neurofibrillary tangles and neuritic plaques and degeneration of specific nerve cells are pathologic characteristics of AD [42].

Dementia is highly heritable but genetically complex [24]. Definite AD diagnosis requires histopathologic confirmation [42]. Currently there is no definite cure for AD. However, Sano et al [54] reported that the progression of AD might be slowed down if the patient's impairment is moderate.

Since a definite AD diagnosis is difficult to attain, McKhann et al [42] proposed a clinical criteria dividing the AD into probable, possible, and definite categories. If one is clinically probable AD, he/she must have a typical insidious onset of dementia and meanwhile the dementia should have been progressed. In addition, AD should be the only fact that accounts for the decline in cognitive functionality. If the patient suffered from other significant diseases apart from AD, which also may cause the progressive decline of cognitive functionality but AD is the most likely pathogenic factor, then a possible AD diagnosis is issued.

AD patients might exhibit a cognitive decline for more than 10 years before a clinical onset of the disease [25]. For the earliest preclinical stage of AD, there is a term called mild cognitive impairment (MCI) [25]. If one is detected as MCI, it does not mean he suffers from dementia at all. Individuals with MCI may only have more defects in cognition than normal people in the same ages. However, not all the patients suffering MCI will remain clinically normal. Some of them, who remain stable are called stable MCI (sMCI); the others are progressive, which is known as progressive MCI (pMCI). We can diagnose pMCI before it progressed to the most severe stage. During the long-time progress, pMCI patients could be diagnosed at different time points since patients can progress at different rates and to progress from MCI to AD can take many years. With the development of the disease, the cognitive and functional performance of a patient greatly declined in clinics. Meanwhile, plaques and tangles grow fast. Figure 1.1 shows the progression in detail. It is clear that with the progression of the disease, the cognitive and functional performance of patients declined faster and faster.

In 2006, there were 26.6 million cases of AD in the world. Predictions indicate that by the year of 2050, the worldwide prevalence of AD will grow to 106.8 millions [5]. The ageing of the world's population may lead to an increase of AD patients. As a result of it, more and more patients may

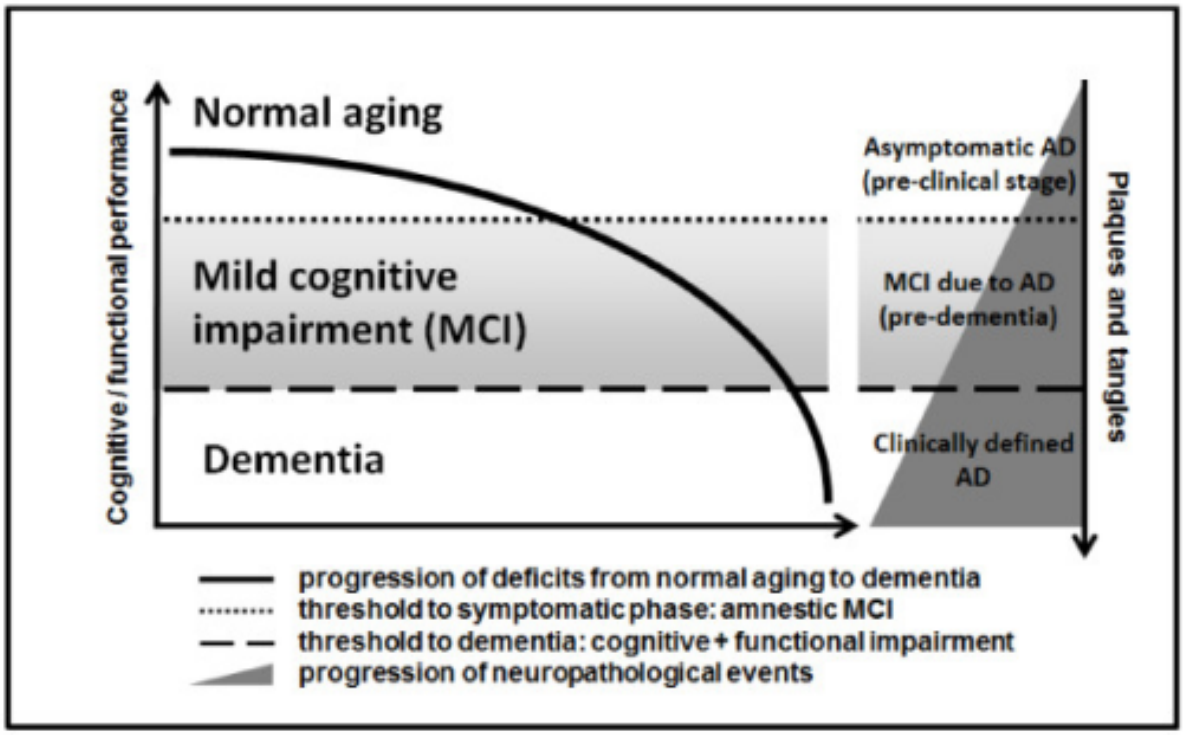


Figure 1.1: AD progression. From normal ageing to dementia, the cognitive and functional performanced delined rapidly. More and more plaques and tangles appear with the progression of AD. This figure is downloaded from `http://openi.nlm.nih.gov/detailedresult.php?img=3022870_1741-7015-8-89-2&query=Alzheimer'sDiseaseprogression&fields=all&favor=none&it=g&sub=none&uniq=0&sp=none&req=4&npos=77&prt=`.

require hospitalisation, which will challenge the national healthcare systems to meet the requirements. The costs spent by the healthcare system for AD treatment will grow rapidly.

1.2 Biomarkers for Alzheimer's Disease

There are various biomarkers which might prove to be significant diagnosis tools for AD, including cerebrospinal fluid (CSF), neurofibrillary tangles (NFTs), amyloid plaques, plasma biomarkers, anatomical markers, as well as imaging biomarkers [6]. These biomarkers play pivotal roles in AD. Figure 1.2 shows the change of biomarkers with the progression of the disease.

The whole brain can be parcellated into three parts: white matter (WM), grey matter (GM), and the CSF [6]. Some biochemical changes that occur in the brain might be reflected in the CSF. Therefore, proteins in CSF can act as biomarkers for neurological diseases, including AD. In addition, both magnetic resonance imaging (MRI) and positron emission tomography (PET) scans can provide a visualization of the whole brain from which we are able to see the CSF.

NFTs [6] and the amyloid plaques, shown in Figure 1.3, are the other factors of AD. Similar to CSF, both of them are related to AD. In terms of CSF, the molecular composition can be used to measure the change in the brain. NFT refers to the intracellular filamentous aggregates of the

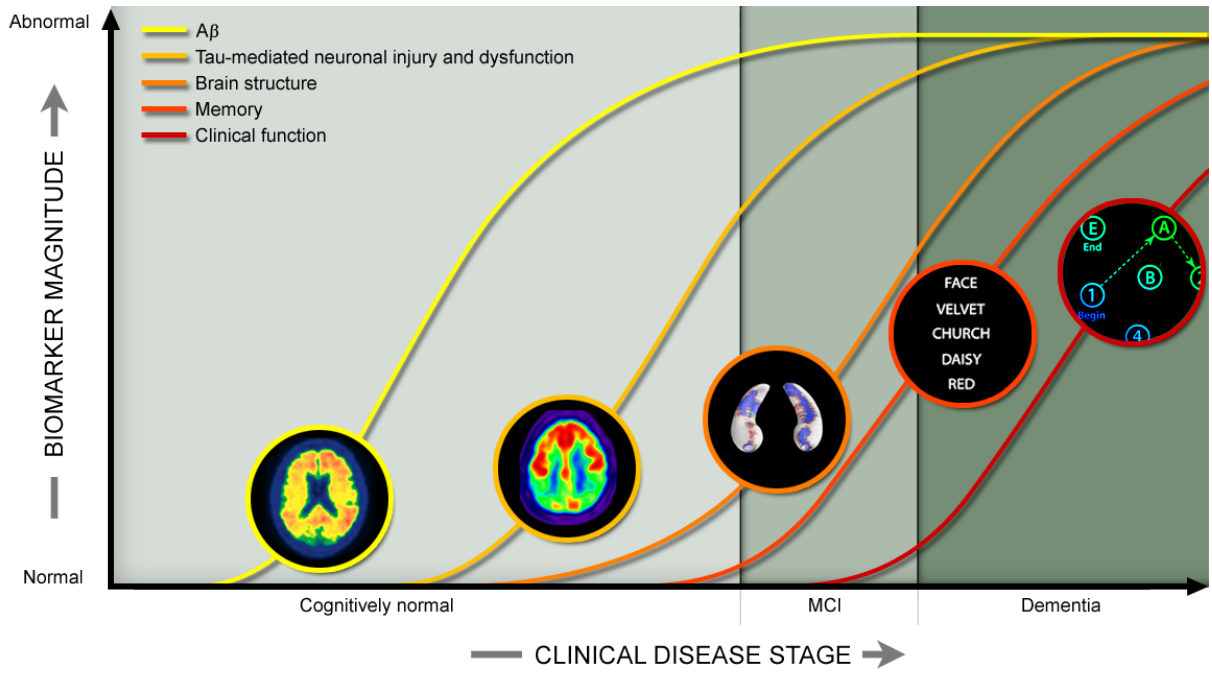


Figure 1.2: AD biomarkers. It was detected that there are five biomarkers playing pivotal roles in AD progression. A patient from normal to abnormal, the biomarkers magnitudes increase significantly. However, different biomarkers increases most rapidly in different stages. This figure is downloaded from `http://adni.loni.ucla.edu/`.

micro-tubule, binds protein tau, which can be used as AD biomarkers. In the left part of Figure 1.3, we can see that each neuron and its surroundings are smooth and healthy. The Alzheimer's picture (right part of Figure 1.3) shows obvious pathological changes in those areas because of the presence of NFTs and amyloid plaques.

Collecting CSF is invasive and unlikely to become a routine procedure in geriatric clinics [6]. It motivates scientists to find peripheral biomarkers to assist in AD diagnosis. They reported that Aβ in plasma is valuable in AD diagnosis. However, Chintamaneni et al [6] revealed that the use of Aβ 42 as the sole biomarker is not very reliable. Additionally they proposed to use the ratio of plasma Aβ 42/Aβ 40 to identify whether a normal individual would progress to MCI or not.

Cerebral atrophy and macroscopic vascular alterations are two AD anatomical biomarkers [6]. As it is shown in Figure 1.4, there is a severe atrophy associated to a patient's brain (right part), compared with a healthy one (left part). The atrophy may result in a dilation of the ventricular system and a widening of cortical sulci. According to Figure 1.4, the degree of atrophy could also be used as as a marker of disease progression.

The hippocampus plays a key role in diagnosis of AD [17]. Hippocampus is sensitive to AD because it is a key anatomical structure to the consolidation of memory. Therefore it is of particular interest in AD research and its diagnosis. Particularly, the atrophic change of the hippocampus has attracted large amounts of interest due to the assumption that the hippocampal tissue loss is related to AD. In terms of atrophy, early studies focused on the measurements of the volume of hippocampus, which has been proposed as one of the main diagnostic symptom.

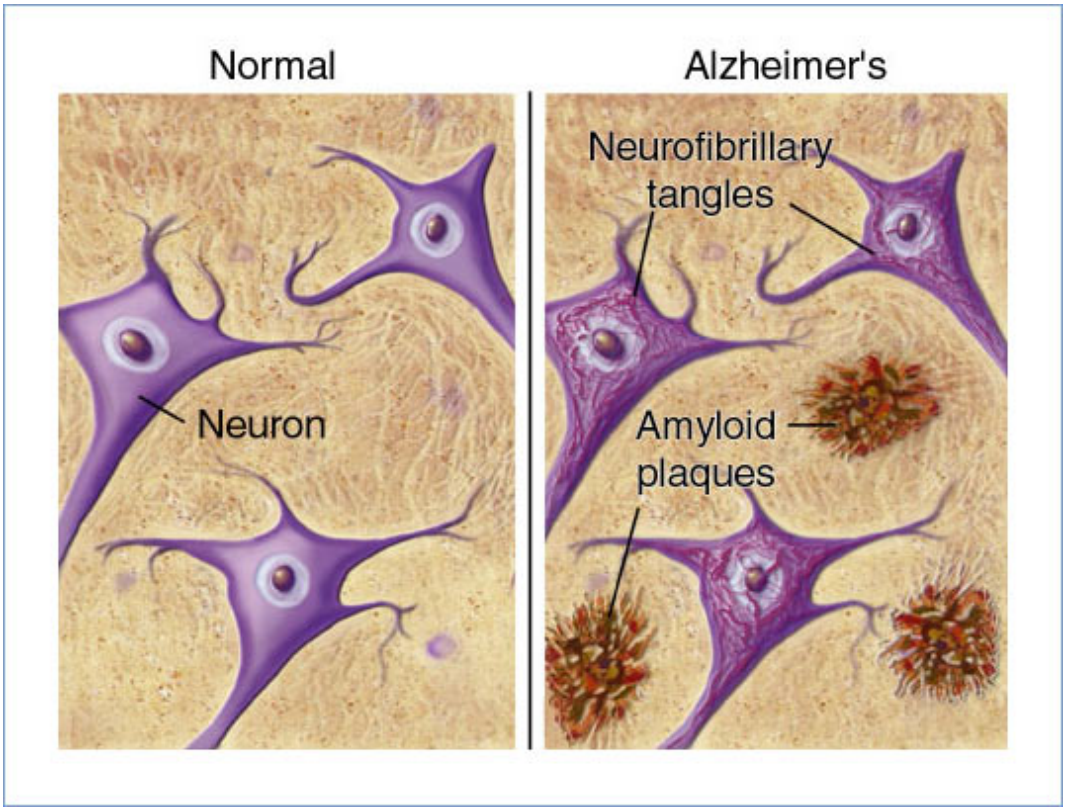


Figure 1.3: NFTs and amyloid tangles.In the left picture, it shows a brain area of neurons in normal. The picture in the right dipicts the same area where there are some pathological changes due to the presence of NFTs and amyloid plaques. They could be the causes of AD. This figure is downloaded from `http://pakmed.net/`.

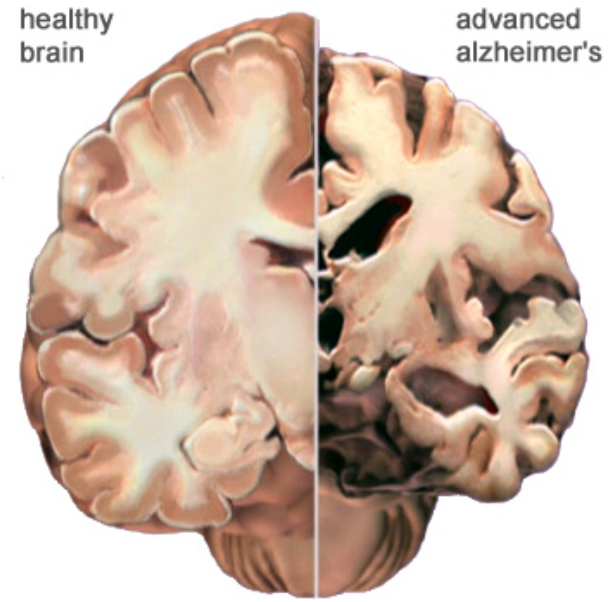


Figure 1.4: Brain atrophy. The left half of the brain is from a healthy individual. The right half of the brain comes from an advanced AD patient. It is obvious that the AD patient's brain has a severe atrophy. This figure is downloaded from `http://adni.loni.ucla.edu/`.

1.3 Neuroimaging

As we mentioned above, it is extremely difficult to diagnose AD. Although physical examinations and psychological testing help clinicians with the diagnosis, only microscopic examination of brain tissue can provide an absolute confirmation [30]. However, with the development of neuroimaging techniques, clinicians are able to examine the structural and functional changes in the brain by way of medical imaging. Among all the techniques in this area, MRI, PET, X-ray computed tomography (CT), singe-photon emission computed tomography (SPECT), and diffusion tensor imaging (DTI) are widely used [22]. In this work, we will only focus on MRI, which exploits the phenomenon of nuclear magnetic resonance (NMR) to produce high quality structural images of the internal organs and other tissues. Typically, the structural changes in the brain associated with AD can be non-invasively assessed using MRI.

Magnetic Resonance Imaging

The internal organs and other tissues in our bodies can be scanned in high quality with the help of the phenomenon of NMR [22]. The patient is always asked to lie in a powerful static magnetic fields, then a structural MRI begins to scan the body. The magnetic field aligns the spins of hydrogen atoms in the body. The application of a radio frequency (RF) electromagnetic pulse can perturb this alignment, which results in the resonance emission of a measurable RF signal. The quality of the obtained images depends on the strength of the static magnetic field. Generally, it is 1.5T or 3T. The magnetic field gradients determined the spatial localisation within the body. The frequency of the resonance signal detected therefore becomes dependent on the location from which it was emitted. Hence, different RF signals can distinguish different tissues.

According to structural MRI, the GM of AD patients decreases significantly, especially in the parahippocampal gyrus, the hippocampus, the amygdala, the posterior association cortex and the subcortical nuclei including the cholinergic basal forebrain [23]. In longitudinal studies, the rate of changes in neuropsychology test and atrophy of brain regions might reflect the progression of the disease, as well as measuring the effects of treatment by using MRI. Figure 1.5 shows the detection of potential disease-modifying treatment effects. In the first graph, before withdrawal of treatment, the cognitive function with disease-modifying decreases slower than with treatment of placebo. After the withdrawal of treatment, the symptomatic decline of cognitive function becomes more rapid than with the treatment of placebo while the disease-modifying decline of cognitive function is still slower than with the treatment of placebo. Hence, in terms of neuropsychology, the disease-modifying treatment can slow down the decline of cognitive function. In the second graph, the difference between the rate of symptomatic decline and the rate of decline with placebo is not significantly obvious. However, cognitive function decreases slower with the treatment of disease-modifying treatment. Similarly, the rate of brain atrophy can also be slow down with the help of disease-modifying treatment. All the rates are obtained by MRI measurements on the volume of brain regions.

Hampel et al [23] reported high reliability in the measurements of volumes from repeated MRI scans. As a result, the volumes of the region of interest (ROI) in the brain can be used as a

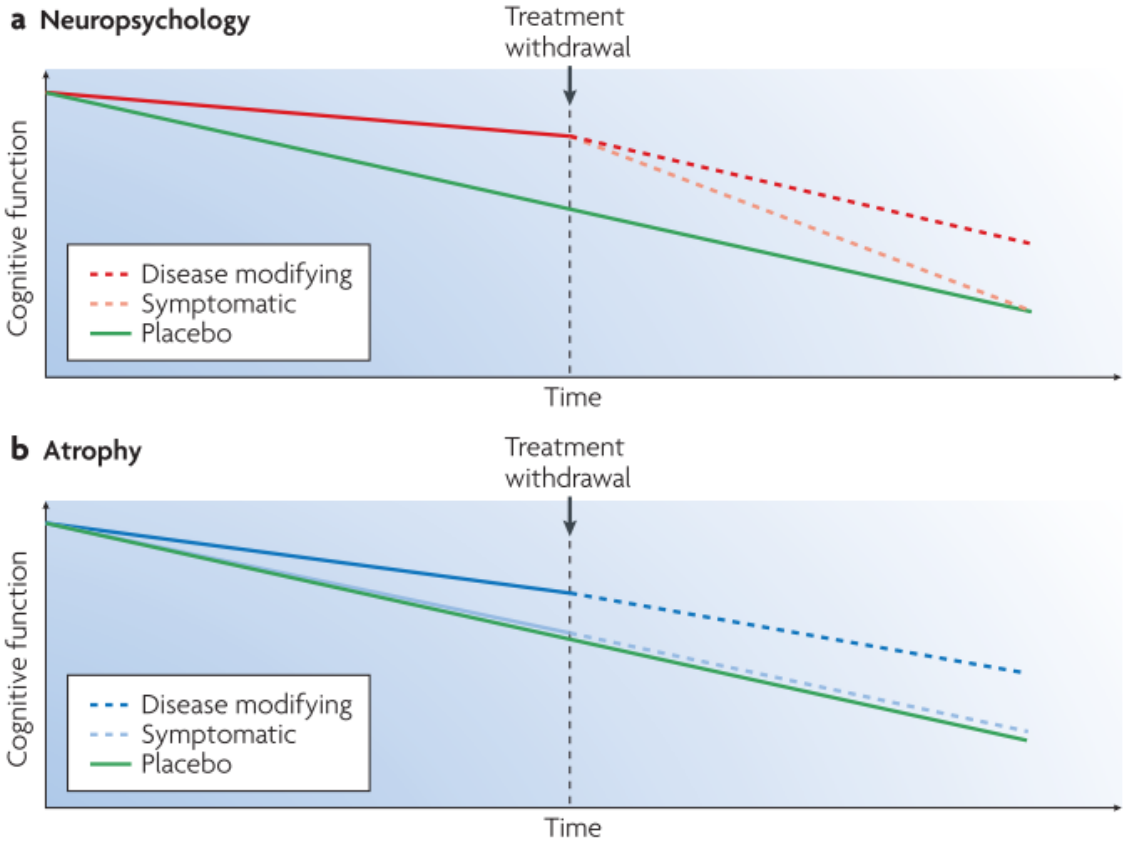


Figure 1.5: Detection of disease-modifying treatment effects. In Graph a, the red solid line depicts the cognitive dicline with the disease-modifying or symptomatic treatment. The red and orange dashed lines indicate the cognitive decline after the withdrawal of treatment. The green solid line demostrates the cognitive decline with placebo. In Graph b, the upper blue line describes the cognitive decline with disease-modifying treatment. The lower blue line shows the it with symptomatic treatment. The green line characterize the atrophy in the brain with a treatment of placebo. These figures are generated by Hampel et al[23].

biomaker of AD. The most commonly used ROI is hippocampus, whose volumes are measured by visual inspection or manual drawing on MRI slices. However, manual drawing is labour intensive. People have already proposed some automated techniques to handle it and several automatic techniques have been put forward. For instance, Wolz et al[60] proposed an automatic hippocampus segmentation method.

1.4 Thesis Outline

Research presented in this work makes contributions to the image-based classification of AD vs CN, pMCI vs CN, as well as pMCI vs sMCI. In particular, a framework of multiple instance learning, which could be used to detect the mis-labelled instances in the training data, is presented.

In Chapter 2, two machine learning techniques will be introduced in detail. The first one is the support vector machine (SVM), which is a supervised learning method. It will be introduced in three stages: linear, soft-margin, and kernel SVM forms. The soft-margin SVM will be wide-

ly used in our following experiments. The second machine learning method is multiple instance learning. It is a semi-supervised learning method. Because of this, it could be based on both supervised and unsupervised learning methods. In terms of multiple instance learning, we will employ supervised-based and unsupervised-based techniques, which are Citation-kNN and Diverse Density, respectively. Both of them will be used in Chapter 6.

In the past five years, numerous papers on the medical image analysis focused on two of the main issues on AD. One is feature extraction and the other one is classifier selection. Whole brain images typically consist of several million voxels, many of which may contain redundant information that may mislead global classification as well as adding to the computational borden to process them. However, if very few simple features are extracted from the raw data, ensuring the computation efficiency, much useful information may be lost, which may also mislead the overall classification. Therefore, there is a trade-off between feature extraction and classifier selection. In Chapter 3, many relevant papers about feature extraction and classification are reviewed.

In Chapter 4, we will describe the data used in this work. Firstly, there is a brief introduction on Alzheimer's Disease Neuroimaging Initiative, which is the source of our data. Then we will discuss how the data was pre-processed, including registration, segmentation, and intensity normalization. In the third stage, the whole data will be divided in different groups, including AD, CN, pMCI and sMCI. According to different experiments, different features will be extracted, which includes the voxels of whole brains and around the hippocampus and the volumes of hippocampus.

In Chapter 5, statistical analysis in terms of the ADNI data is performed. In this chapter, different training algorithms, cross-validation strategies, and comparison groups are assessed. It is clear that in this case, the voxels of hippocanpus towers over the whole brain voxels in terms of classification between both AD and CN and pMCI and sMCI. Also, the classification error rate of scans from different clinical sites is assessed, which results in that instances from most clinical sites are fairly accurate. However, there are few clinical sites from which instances tend to have a high classification error rate.

Chapter 6 details a novel framework of multiple instance learning, which stands between supervised learning and unsupervised learning. Multiple instance learning copes with the situation where not all the labels from the training instances can be completely trusted. Under this assumption, a small number of instances whose labels are definitely true are recognized as negative while others are regarded as positive. Then we can put several instances into a bag. For each bag, if it contains no positive instance, its label is negative; otherwise it is positive. As a result, binary classification could be conducted on bags. It is easy to test the label of an instance by manually assigning it to bags and testing the labels of those bags. Based on our own model, a small number of mis-labelled ADs and CNs are discovered.

Finally, our overall conclusion, contributions of this work and possible future work are presented in Chapter 7.

2 Machine Learning Methods

2.1 Support Vector Machine

In 1980s, neural network was one of the most popular supervised learning techniques [2]. It was based on the idea that certain kinds of functions, which are or can be derived from basic elementary functions, are able to represent the target mapping. According to it, neural networks were used in regression, classification and etc. However, under the circumstance of binary classification, there was not sufficient theoretical support for neural networks. In 1992, Vapnik et al [59] developed a state-of-the-art classification technique called support vector machine, known as SVM. It was a milestone in supervised learning because it combines theoretical and practical worlds together. Practically, SVM is committed to find the maximum margin between two classes. Theoretically, it is a optimal programming procedure. As a result, SVM became the most popular classification method instead of neural networks. In addition, the research following Vapnik et al improved SVM significantly such that it is able to deal with the misclassification situations. Further, multi-class SVM and kernel tricks were developed.

It is a common knowledge that in supervised learning, each instance has a given label and we need to bridge a mapping between the instances and their labels. In this paper, we define a set of training data consisting of N points,

$$(\mathbf{x}_i, y_i), i = 1, ..., N$$

$\mathbf{x}_i$ is the feature vector in n dimensions that describes the data point and y_i is the corresponding label of $\mathbf{x}_i$. We only take binary classification into account. Therefore if $\mathbf{x}_i$ is positive, its y_i is $+1$; otherwise, y_i is -1. In supervised learning, we have to find a function $f(\mathbf{x})$, which is able to predict the y for each unseen $\mathbf{x}$.

2.1.1 Linear SVM

From very beginning, the SVM was to classify two classes. In addition, it was assumed that there should be a linear function that could reach the goal. That means there exists a hyperplane in n dimensions, which could linearly separate the two classes. In this case, we define the equation as below:

$$f(\mathbf{x}) = \mathbf{w}^{\mathrm{T}}\mathbf{x} + b = 0 \tag{2.1}$$

In Equation (2.1), $f(\mathbf{x})$ is the mapping that we have to find. $\mathbf{w}$ is the normal vector to the hyperplane. b is the bias. To explain more simply, if the dimension is 1, then $f(\mathbf{x})$ is a straight line in 2D Euclidian space. $\mathbf{w}$ is its slope and b is the intercept on y-axis. Two classes locate at opposite sites of the hyperplane. Therefore we may use the hyperplane as the discriminant of the two classes.

For a unseen data points, if $f(\mathbf{x}_i) > 0$ then it locates at the upper side of the hyperplane. Thus its predicted label should be 1. One the other hand, if $f(\mathbf{x}_i) < 0$ the data point locates at the lower side the the hyperplane such that it should be labelled as -1. Hence $f(\mathbf{x})$ in equation (2.1) will be the decision rule.

However, there is no wonder that there should be more than one hyperplanes meeting the requirements mentioned above. The obvious motivation to solve it is to find the only one hyperplane which is able to distinguish the two classes most significantly. Specifically, we have to find out the largest margin between the two classes and the hyperplane in the middle the the maximum margin will be the right one we need. Mathematically, in order to use the hyperplane for classification, we have to use a constant to scale $\mathbf{w}$ and tune b properly to make

$$\begin{aligned} \mathbf{w}^{\mathrm{T}}\mathbf{x}_i + b \geq +1 \quad \text{then} \quad y_i = +1 \\ \mathbf{w}^{\mathrm{T}}\mathbf{x}_i + b \leq +1 \quad \text{then} \quad y_i = -1 \end{aligned} \tag{2.2}$$

These equations represent parallel bounding hyperplanes that separate the data. Geometrically, the perpendicular distance between the parallel bounding hyperplanes is $2/\|\mathbf{w}\|$. To simplify Equation (2.2), it is possible to combine them together into one equation as

$$y_i(\mathbf{w}^{\mathrm{T}}\mathbf{x}_i + b) \geq 1 \tag{2.3}$$

Therefore, to find the maximum margin is to maximize the perpendicular distance between the bounding hyperplanes, which is to minimize $\|\mathbf{w}\|/2$. We can summarize it as a particular in quadratic programming

$$\begin{aligned} \min \quad & \frac{1}{2}\mathbf{w}^{\mathrm{T}}\mathbf{w} \\ \text{s.t.} \quad & y_i(\mathbf{w}^{\mathrm{T}}\mathbf{x}_i + b) \geq 1 \quad i = 1, ..., N \end{aligned} \tag{2.4}$$

Intuitively, the two parallel bounding hyperplanes must go through some of the data points; otherwise there would be more space for them to make the margin larger. In this way, those data points with $f(\mathbf{x}) = \pm 1$ are called support vectors. SVM therefore is the machine finding the support vectors because once support vectors are found, it becomes easy to find the largest margin between the two classes. Hence the discriminant hyperplane becomes surfaced. That means finding support vectors is the key problem in this case. Therefore this method was named after the support vectors. Figure 2.1 describes the core idea of the algorithm. In the figure, the margin between two dashed lines are the fattest. When predicting new points, if it locates in the left of the solid line, it belongs to the blue class; otherwise it belongs to the red class.

Equation (2.5) cannot be solved directly. It is a constrained optimization problem. We may use Lagrangian multipliers to transform it into unconstrained form. The optimization problem could then be re-expressed as

$$\begin{aligned} \min_{\mathbf{w},b} \max_{\alpha} \quad & \left\{ \frac{1}{2}\mathbf{w}^{\mathrm{T}}\mathbf{w} - \sum_{i=1}^{N} \lambda_i [y_i(\mathbf{w}^{\mathrm{T}}\mathbf{x}_i - b) - 1] \right\} \\ \text{s.t.} \quad & \lambda_i \geq 0 \end{aligned} \tag{2.5}$$

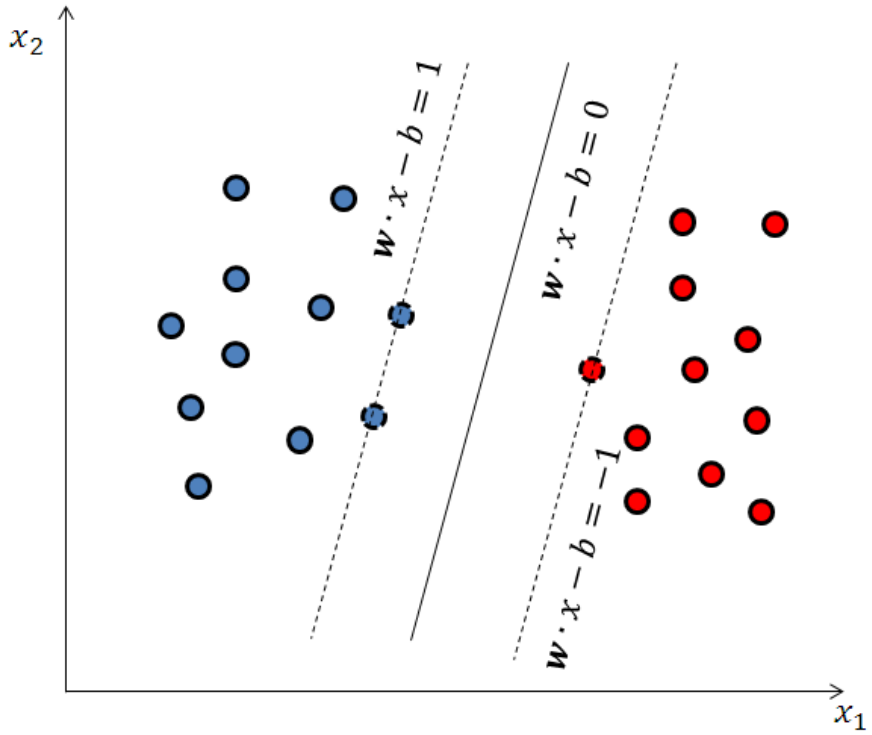


Figure 2.1: The illustration of SVM. Red circles and blue circles represent data points belonging to two different classes. The circles with dashed contours are support vectors. The margin between two parallel dashed hyperplanes is the largest. The solid line dipicts the discriminant hyperplane.

According to Equation (2.5), the normal vector $\mathbf{w}$ could be derived as

$$\mathbf{w} = \sum_{i=1}^{m} \lambda_i y_i \mathbf{x}_i$$

which is a linear combination of the feature vectors. According to the theory of quadratic programming, the optimizers could only be reach at the boundary of the feasible region. That means the λ_i will be zero if the corresponding data point $\mathbf{x}_i$ is not the support vector. Suppose there are overall N_{sv} support vectors in this case, then the b could be found by

$$b = \frac{1}{N_{sv}} \sum_{i=1}^{N_{sv}} (\mathbf{w}^{\mathrm{T}} \mathbf{x}_i - y_i)$$

It is the average error between all the $\mathbf{w}^{\mathrm{T}} \mathbf{x}_i$ and y_i.

The primal form of the Lagrangian $L(\mathbf{w}, b, \lambda)$ may be equivalently written in dual form by substituting the above expression for $\mathbf{w}$. The dual form,

$$\begin{aligned} \max_{\lambda} \quad & \tilde{L}(\lambda) = \max_{\lambda} \left\{ \sum_{i=1}^{N} \lambda_i - \frac{1}{2} \sum_{i,j} \lambda_i \lambda_j y_i y_j \mathbf{x}_i^{\mathrm{T}} \mathbf{x}_j \right\} \\ \text{s.t.} \quad & \lambda_i \geq 0 \text{ and } \sum_{i=1}^{N} \lambda_i y_i = 0 \end{aligned} \tag{2.6}$$

expresses the optimization criterion in terms of inner products of the feature vectors. This is an important property for the creation of non-linear SVM classifiers.

2.1.2 Soft-margin SVM

Under some practical circumstances, the real data may be corrupted by the noise, which results in that there is no linear hyperplane that can separate the data. In this case, we have to find a new hyperplane, which can minimally mis-classify the data. Therefore, we introduce a slack variable ξ, which can measure the degree of misclassification of the feature vectors. The optimization becomes a trade-off between maximizing the margin and minimizing the degree of misclassification. A penalty parameter C is required to control this trade-off. Consequently, the constrained optimization should be expressed as

$$\begin{aligned} \min_{\mathbf{w},\xi,b} \quad & \left\{ \frac{1}{2}\mathbf{w}^{\mathrm{T}}\mathbf{w} + C\sum_{i=1}^{N}\xi_i \right\} \\ \text{s.t.} \quad & y_i(\mathbf{w}^{\mathrm{T}}\mathbf{x}_i - b) \geq 1 - \xi_i \text{ and } \xi_i \geq 0 \end{aligned} \tag{2.7}$$

Similarly, we use Lagrangian Multiplier Algorithm to transform the constrained problem into unconstrained form as

$$\begin{aligned} \min_{\mathbf{w},\xi,b} \max_{\lambda,\beta} \quad & \left\{ \frac{1}{2}\mathbf{w}^{\mathrm{T}}\mathbf{w} + C\sum_{i=1}^{N}\xi_i - \sum_{i=1}^{N}\lambda_i[y_i(\mathbf{w}^{\mathrm{T}}\mathbf{x}_i - b) - 1 + \xi_i] - \sum_{i=1}^{N}\beta_i\xi_i \right\} \\ \text{s.t.} \quad & \lambda_i, \beta_i \geq 0 \end{aligned} \tag{2.8}$$

Additionally, its dual form is

$$\begin{aligned} \max_{\lambda} \quad & \tilde{L}(\lambda) = \max_{\lambda} \left\{ \sum_{i=1}^{N}\lambda_i - \frac{1}{2}\sum_{i,j}\lambda_i\lambda_j y_i y_j \mathbf{x}_i^{\mathrm{T}}\mathbf{x}_j \right\} \\ \text{s.t.} \quad & 0 \leq \lambda_i \leq C \text{ and } \sum_{i=1}^{N}\lambda_i y_i = 0 \end{aligned} \tag{2.9}$$

It is clear that the only difference between linear SVM and the soft-margin SVM is that the λ_i has an upper bound C in soft-margin SVM.

2.1.3 Kernel SVM

In cases where the training data are not linearly separable but they can be non-linearly separate, seen in the left part in Figure 2.2, there may exist a non-linear function $\phi(x)$ that can map the original data into a higher-dimensional space where the data could be linear separate, seen in the right part in Figure 2.2. The non-linear function *phi* is called the kernel function. By using kernel tricks, the non-linear classification problem is able to be transformed into linear classification problem, which could be solve by SVM.

As we mentioned above, we may use the dual form of SVM to add the kernel tricks into it by

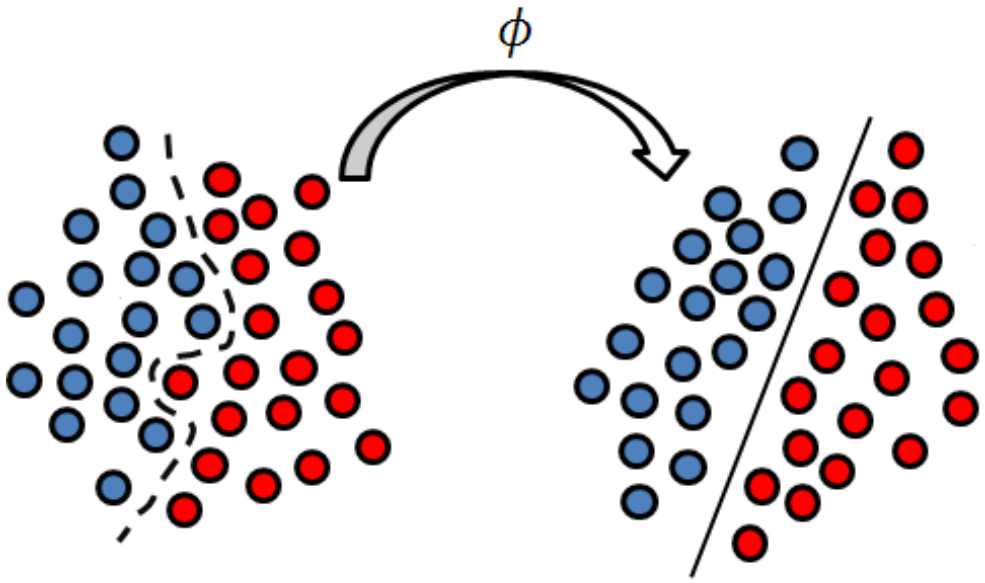


Figure 2.2: Illustration of kernel tricks. The orignial data is non-linearly separable. After the mapping of a trick function ϕ, they can be linearly separate.

replacing the inner product of original feature vectors with the inner product of feature vectors with kernels. It could be expressed as below:

$$\tilde{L}(\lambda) = \sum_{i=1}^{N} \lambda_i - \frac{1}{2} \sum_{i,j} \lambda_i \lambda_j y_i y_j \phi(\mathbf{x}_i)^{\mathrm{T}} \phi(\mathbf{x}_j) \tag{2.10}$$

The optimization criterion is thus expressed in terms of inner products of the transformed feature vectors. Different non-linear mapping ϕ could map the vectors into different non-linear spaces in terms of a kernel function $\mathbf{k}(\mathbf{x}_i, \mathbf{x}_j) = \phi(\mathbf{x}_i)^{\mathrm{T}} \phi(\mathbf{x}_j)$. It provides us with an explicit way to perform the mapping.

The Gaussian radial basis function [3], known as the RBF, is one of the most commonly used kernel functions. The RBF kernel on two samples $\mathbf{x}_i$ and $\mathbf{x}_j$, represented as feature vectors in the training data set, is defined as

$$\mathbf{k}(\mathbf{x}_i, \mathbf{x}_j) = \exp\left(-\frac{\|\mathbf{x}_i - \mathbf{x}_j\|_2^2}{2\sigma^2}\right) \tag{2.11}$$

$\|\mathbf{x}_i - \mathbf{x}_j\|_2^2$ may be recognized as the squared Euclidean distance between the two feature vectors. σ is a free parameter. An equivalent, but simpler definition involves a parameter $\gamma = -\frac{1}{2\sigma^2}$. Hence, the kernel function can be re-expressed as

$$\mathbf{k}(\mathbf{x}_i, \mathbf{x}_j) = \exp(\gamma \|\mathbf{x}_i - \mathbf{x}_j\|_2^2) \tag{2.12}$$

2.1.4 Application to AD Diagnosis

In AD diagnosis, clinicians used to wonder whether a patient is AD or not based on the MRI. They commonly compared the patient's data with those in the database. The data in the database should be trained to generate a classifier. By using the trained classifier, the patient could be predicted that whether he/she is AD or not. In addition, for MCI patients, they should be traced checking whether they have processed to pMCI or even to AD. In this case, it is also necessary to classify between sMCI and pMCI.

Practically, features such as clinical ROI are extracted from patients' MRIs. One or more than one kind of features are organized into a vector $\mathbf{x}$. A number of those vectors consist of the training dataset. A SVM could be trained based on the dataset and the labels of following instances can be predicted by it.

2.2 Multiple Instance Learning

In machine learning, techniques could be divided into supervised, unsupervised, semi-supervised and reinforcement learning based on whether the training data have the given labels. For supervised learning, all the instances have the corresponding given labels while for unsupervised learning, all the instances do not have their labels. For example, SVM mentioned above is one of the classical supervised learning method and Gaussian Mixed Model is one of the unsupervised learning method. For reinforcement learning, the instances have labels but the labels are delayed. Particularly, for semi-supervised learning, some of the instances have labels but the others do not. As a result, algorithms solving semi-supervised problems can be on a basis of both supervised and unsupervised learning. Not surprisingly, there are various semi-supervised algorithms. Multiple instance learning is a class of semi-supervised learning. It was firstly introduced by Dietterich et al [11] in 1997.

In multiple instance learning, the training set is composed of many bags each contains several instances. A bag is positively labelled of it contains at least one positive instance; otherwise it is labelled as negative. The task is to learn some concepts from the training set for correctly labelling unseen bags. It has a number of different algorithms, including Diverse Density [41], Citation-kNN [58], eMIL [32], MIForests [33], multi-instance ensembles [36], Bayesian-kNN [58], mi-SVM [12], mi-DS [47], as well as IL-SMIL [29]. In fact, most of those algorithms are modified from supervised learning. Based on different conditions, researched may develop different strategies on multiple instance learning to fit their problems properly. Therefore multiple instance learning does not have a sole expression like SVM.

Generally, it is extremely difficult to use multiple instance learning to discriminate the label of an unseen instance. If an unseen bag is predicted as negative, then all the instances in the bag are negative. However, if the bag is predicted as positive, the instances in it could never been predicted. In addition, it is possible to regard each instance as a bag because not all the instances were assigned a label.

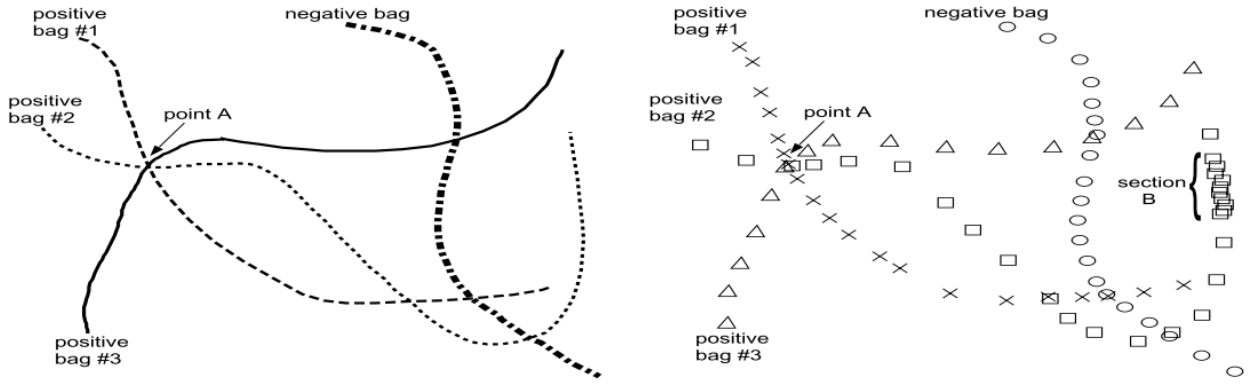
Multiple instance learning has achieved a notable success in various fields. From very begin-

ning, Dietterich et al [11] investigated it in the problem of drug activity prediction. Then it was widely used. Ding et al [12] used multiple instance learning in breast ultrasound image classification. Leung et al[35] handled the label noise in video classification via multiple instance learning. Maron et al [40] classified the natural scenes with multiple instance learning. Raykar et al [52] automatically selected the features and transferred them inductively by using multiple instance learning.

2.2.1 Diverse Density

The Diverse Density algorithm [41] is one if the most renowned multiple instance learning algorithm, which regards each bag as a manifold. Intuitively, in negative bags the diverse density at each point is low while in positive bags it is high particularly where there are a number of both positive and negative instances. It is assumed that a positive bag intersects all positive feature manifolds but it never intersect any negative feature manifolds. As a result, in the feature space, diverse density at a point describes the density of different bags near that point [63]. Specifically, the more the different positive bags are near that point, the higher the diverse density is. The further the negative instances are from that point, the higher the diverse density is. In this way, searching for a point in the feature space whose diverse density is high is the key task of this algorithm. In front of it, we need to define the diverse density in formal mathematics.

Diverse density is derived from the idea of a molecular example by Maron et al [41]. Assume that a feature vector can represent the shape of a candidate molecule. Therefore, a point in n-dimensional feature space indicates a instance of the molecule. Since the shape of the molecule usually changes rigidly and non-rigidly in n-dimensional space, the change of the shape may trace out a manifold. In Figure 2.3a, there are four molecules changed their shapes and it shows their paths. Particularly, the three positive bags intersect at one point.



(a) The different shapes that a molecule can take on are represented as a path. The intersection point of positive path is where they took on the same shape.

(b) Samples taken along the paths. Section B is a high density area, but point A is a high diverse density area.

Figure 2.3: The heuristics of diverse density. Both of the two sub-figures were generated by Maron et al[41]. They are definitely classic figures to clarify the core ideas of diverse density.

According to the biology [41], if a molecule is labelled as positive, then there is at least one place along the manifold, taking the right shape for it to fit into the target protein. Otherwise,

if no points on the manifold may be right for it to fit into the target protein, it will be labelled as negative. Suppose that only one shape that will bind the molecule to the target protein, all positive manifolds will intersect at the point corresponding to that shape, without intersecting any other negative manifolds. In Figure 2.3a, Point A is the right shape point and all the three positive manifolds intersect at it.

However, in the practical science discovery, including drug research, we are never accessible to the right result at the beginning. We have to do large amount of statistics and try to discover potential regular patterns behind the data we collected. Therefore, we may only sampled a small number examples from our experiments and use them to estimate the distribution of the whole data. If we do sampling on the model in Figure 2.3a, we may get the result shown in Figure 2.3b, which indicates the fact that we have to study the location at Point A in Figure 2.3a by approximating it in in Figure 2.3b. The reason is in in Figure 2.3b, there may not be an exact Point A since different positive manifold may not exactly intersect with each other after discretization. As a result, instead of finding an intersection point, we are motivated to find an area where the density of positive points is high and meanwhile that of negative points is low. That means we are trying to find an area where the diverse density is high. To illustrate more clearly, for example in Figure 2.3b, the density of an area of Section B is high because there are many negative points in that area. However, all the instances are negative. Therefore the diverse density is low. On the other hand, in the area near Point A, the diverse density is high due to that there are both positive and negative instances around there.

To formally define the algorithm of diverse density. We define B_i^+ as the i-th positive bag, B_{ij}^+ as the j-th point in that bag, and B_{ijk}^+ as the value of the k-th feature of that point. Similarly, B_{ij}^- represents a negative point. Suppose that the true concept is a single point c, we can find it by maximizing

$$\Pr(x = c|B_1^+, \ldots, B_n^+, B_1^-, \ldots, B_m^-)$$

over all points x in feature space. According to Bayesian rule, we may use an uninformative prior knowledge over the concept location to transform the problem into maximizing the likelihood

$$\Pr(B_1^+, \ldots, B_n^+, B_1^-, \ldots, B_m^-|x = c)$$

. In addition, assume that the bags are conditionally independent given the target concept c, the best hypothesis is

$$\arg\max_x \prod_i \Pr(B_i^+|x = c) \prod_i \Pr(B_i^-|x = c)$$

If we use Bayesian rule and an uniform prior concept location again, this is equivalent to

$$\arg\max_x \prod_i \Pr(x = c|B_i^+) \prod_i \Pr(x = c|B_i^-) \tag{2.13}$$

Equation (2.13) is the general definition of maximum diverse density. In [39], Maron gave out the numerical solutions to it. However, we have to instantiate terms in the products. Maron et al [41] indicated a noisy-or model to handle it. Their model suggested that the probability that not all

points missed the target is

$$\Pr(x = c|B_i^+) = \Pr(x = c|B_{i1}^+, B_{i2}^+, \dots) = 1 - \prod_j (1 - \Pr(x = c|B_{ij}^+))$$

and similarly,

$$\Pr(x = c|B_i^-) = \prod_j (1 - \Pr(x = c|B_{ij}^-))$$

The model defined

$$\Pr(x = c|B_{ij}) = \exp(-\|B_{ij} - x\|^2)$$

as the causal probability of an individual instance on a potential target. It is based on the Euclidean distance since the diverse density is related to the distance between bags. Specifically, if the diverse density at a point x is high, it should be in close of an instance in a positive bag. If the diverse density at a point x is low, it may be near an instance in a negative bag. Particularly, based on the definition of the causal probability, diverse density at an intersection of n bags is exponentially higher than it is at an intersection of $n - 1$ bags. However, all it takes is one well placed negative instance to drive the diverse density down.

In Euclidean space, Euclidean distance is commonly used to measure the length from one instance to another based on the features. It regards all the features are relevant or equally weighted. However, there are also some circumstances where not all the features are relevant or some of them overweight others. Using the same framework, not only the best location in feature space, but also the best weighting of the features could be found. Finding the best weighting of the features under the same framework is to find the best scaling of the individual features. The diverse density could be the best weights for each feature [41].

2.2.2 Citation-kNN

As we mentioned above, there are various algorithms that can classify an unseen bag. Dietterich et al [11] proposed the Diverse Density method, which has been described before. It is a distinctive multiple instance learning technique. However, there are also a number of methods based on classic machine learning methods, including supervised learning, unsupervised learning, as well as inductive logic programming [44]. Many of them are eager learning techniques.

However, Wang et al [58] adapted the lazy learning to multiple instance learning problems. In lazy learning, the predicted label of an unseen example will only depend on its "neighbours", which are most similar to it to some extent. In k-nearest-neighbour (kNN) [44] is one of the most famous lazy learning algorithm. The similarity defined in kNN is mainly the distance and usually it is Euclidean distance or modified Euclidean distance. It trains all the training examples and selects the corresponding "neighbours". However, in multiple instance learning the training instances are bags, which is composed of a number of instances. As a result, the elementary kNN cannot adapt to multiple instance learning directly. In the first stage, we have to define the distance between bags. It is one of the key problems.

Wang et al [58] proposed that the Hausdorff distance could be used to measure the distance between bags in multiple instance learning because it provides a metric function between subsets of a metric space. According to the definition of Hausdorff distance, two sets A and B are within Housdorff distance d of each other iff every point of A is within distance d of at least one point of B, and every point of B is within distance d of at least one point of A. Formally, provided that two sets of points $A = \{a_1, \ldots, a_m\}$ and $B = \{b_1, \ldots, b_n\}$, the Housdorff distance is defined as

$$H(A, B) = \max\{h(A, B), h(B, A)\}$$

where

$$h(A, B) = \max_{a \in A} \min_{b \in B} \|a - b\|$$

However, Wang et al [58] also mentioned that the Hausdorff distance is very sensitive to even a single outlying point of A or B. For example, consider $A = \{1, 2, 3\}$ and $B = \{4, 5, 20\}$, where the 20 is some large distance away from every point of A. In this case, $H(A, B) = 17$, which means that the distance is solely determined by this outlying point. As a result, we cannot adapt original Hausdorff distance directly in our case. To improve the robustness of this distance with respect to noise, they modified the original Hausdorff distance. The modification focused on replacing the global maximum-ranked distance by the k-th ranked distance, which is

$$h_k(A, B) = \operatorname*{kth}_{a \in A} \min_{b \in B} \|a - b\|$$

where $h_k(A, B)$ is the k-th ranked value. Particularly, when $k = m$, $h_k(A, B) = H(A, B)$, which is the maximum value. On the other hand, when $k = 1$ the global distance between A and B is dependent on the minimal distance between points in the two sets. Since

$$h_1(A, B) = \min_{a \in A} \min_{b \in B} \|a - b\| = \min_{b \in B} \min_{a \in A} \|a - b\| = h_1(B, A)$$

in this case,

$$H(A, B) = h_1(A, B) = h_1(B, A)$$

They call this measure the minimal Hausdorff distance.

In fact, using the Hausdorff distance was still not sufficient to adapt kNN to the multiple instance learning [58]. Conventional kNN algorithm cannot handle the situation where the numbers of instances in positive and negative bags are not balanced. The reason is the positive and negative bags cannot be selected in the nearest neighbours equally. As a result, it is more valid to predict the unseen instance, which belongs to the class who has more training instances by chance. For example, we have 3 positive bags and 23 negative bags and we choose $k = 3$. The current nearest neighbours are two positive bags and one negative bag. According to the algorithm, we should set the new bag as positive. However, if it is predicted as negative, the overall classification accuracy will be higher.

In multiple instance learning, there are both negative and positive instances in positive bags

while there are only negative instances in negative bags. Assume that the numbers of positive and negative bags are balanced and there are equal number of instances in each bag. It is obvious that the overall number of negative instances is more than positive ones. Therefore, the negative instances in positive bags may affect negative bags. For example, Wang et al [58] drew a simple figure of Figure 2.4 to illustrate the particular case. In Figure 2.4, the instances in negative bags are represented as round dots and the instances in positive bags as square blocks. Given $\{P_1, P_2, N_1\}$ as three training bags, N_2 will be classified as positive. Given $\{N_1, N_2, P_1\}$ as three training bags, P_2 will be classified as negative.

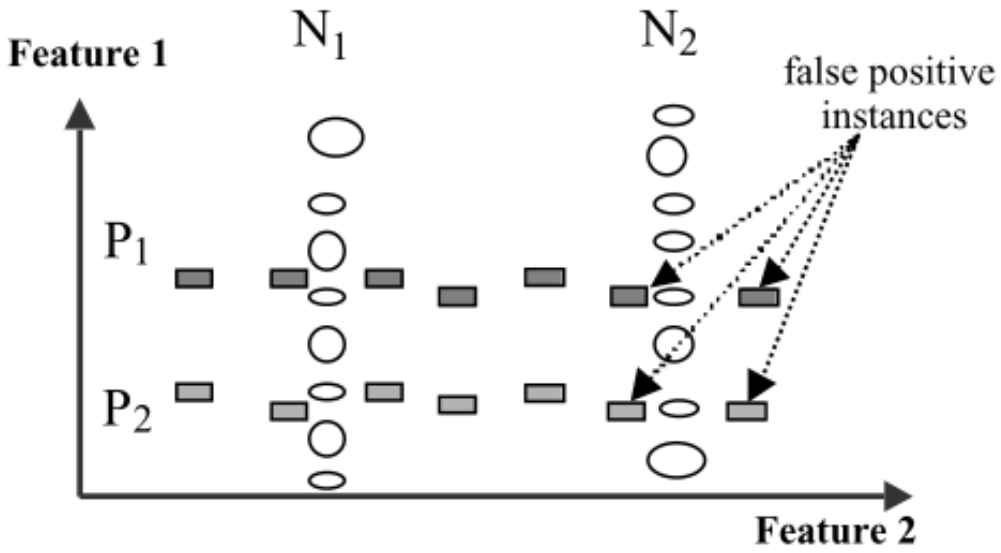


Figure 2.4: Illustration of kNN in multiple instance learning. The figure was generated by Wang et al [58]. It dipicts a 2-dimensional feature space. N_1 and N_2 are two negative bags and P_1 and P_2 are two positive bags. Round circles are negative instancs and square blocks are positive instances.

To handle the problem, Wang et al [58] developed a citation approach to adapt kNN algorithm to multiple instance learning more properly. They were motivated by the notion of citation from library and information science. In library and information science, document indexing is one the the most significant research topic as it is difficult for a researcher to find a specific paper that he/she wants among masses of papers. One of the most popular method is based on references and citers. In terms of reference, a paper is related to the reference of another paper if it is cited by that paper. Likewise, a paper is related to the citer of another paper if it cites that one. As a result, any paper are related to both references and citers.

In this case, we have to consider the "neighbours" of a bag b according to the Hausdorff distance and meanwhile take all the other bags into account, which recognize b as a "neighbour". It is suggested that we may use both references and citers of an unseen instance to predict its label. In terms of reference, we should consider the R-nearest neighbours of b, which is simple. However, defining the citers of the bag is complex. Set n be the number of all example bags $B = \{b_1, \ldots, b_n\}$, then for an example $b \in B$, all other examples $B \backslash b = \{b_i | b_i \in B, b_i \neq b\}$ can be ranked according to the similarity to the example b. We use $Rank(a, b)$ to represent the rank of the example $a \in B \backslash b$.

Particularly, set $Rank(b, b)$ to be ∞. In this way, the C-nearest citers of b could be defined as

$$Citers(b, C) = \{b_i | Rank(b_i, b) \leq C, b_i \in B\}$$

In addition, the C-nearest citers may also be defined based on distance [58].

To combine R-nearest references and the C-nearest citers together to predict the label of an unseen bag [58], assume that there are R_p positive bags and R_n negative bags for the R-nearest references and there are C_p positive bags and C_n negative bags for the C-nearest citers. There are overall p positive bags and n negative bags. Therefore, $p = R_p + C_p$, and $n = R_n + C_n$. p and n could be computed by the Hausdorff distance. If $p > n$, then the class of the bag b is predicted as positive; otherwise, it is negative. Until now, the prediction is completed. It should be noted that when a tie happens, the class of b is set to be negative.

3 Literature Review

Since 2008 there were a number of studies involving in techniques of AD diagnosis. Papers we reviewed are the most highly cited since then. The main work focused on feature extraction as well as classifiers choosing and optimization. We listed a table (Table 3.1) to conclude and compare between different studies in four aspects, including the data, the features, the classifier, as well as the accuracy. In terms of the data, we are concerned about the number of participants that each paper used, whether they are from Alzheimer's Disease Neuroimaging Initiative (ADNI), and what kind of comparison group did they focus on. In addition, we consider what kind the features were extracted, what kind of classifiers they selected, and how was their best classification accuracy.

3.1 Data

The data of the studies was typically from professional organizations and institutes, such as ADNI and Harvard Medical School. Data from these places are widely used in the world, which means they are proved to be valid by numerous experiments.

In most of the reviewed papers, the classification concentrated on AD patients and normal controls. It is a basic start point to develop a system to distinguish between definite patients and healthy individuals. More importantly, our final goal should be distinguishing MCI subjects and normal people and distinguishing AD patients and MCI individuals. Once one has been definitely diagnosed as AD or healthy, there is no need to diagnose them by computer techniques. On the other hand, it is more meaningful to detect whether a "healthy" one is on the way to MCI or not and whether an MCI patients is going to be AD or not. For example, if we are confident one who is suffering on MCI will be an AD patient, a series of medical cares could be provided for him/her in order to slow down the progress of the disease.

In terms of the number of participants involving in those papers, it ranges from 30 to more than 600. In our opinion, participants less than 100 might not be valid. The classification methods used in the papers tended to be machine learning technologies. They needed to perform cross-validation to evaluate their algorithms. 10-fold-cross-validation is one of the most popular method to do the evaluation. According to Mitchell's book [44], the statistically-based methods call for large numbers of samples in learning and validation stages. He pointed out that at least 30 samples are needed in each fold when performing k-fold cross-validation. Therefore, at least 300 examples are required when training classifiers. For those who performed leave-one-out cross-validation, it also calls for at least 100 samples. As a result, to be more convincing, many studies, including, [14], [10] and [19] should take more samples.

Table 3.1: Literature Review. The first column shows the authors of each work. The second column represents the number of participants used. The third and fourth columns indicate the classifiers they selected and the features they extracted, respectively. The fifth column demonstrates the best accuracy achieved by every work. The sixth column records whether the data comes from ADNI. The last colomn marks the comparison group of each work.

Author	# Participants	Classifier	Features	Accuracy (%)	ADNI	Comparison group
Vemuri et al [57]	380	SVM	STAND-score	89.3	No	AD vs CN
Fan et al [14]	30	SVM	ROI	100	No	AD vs MCI
Davatzikos et al [10]	30	SVM	GM, WM, CSF	90	No	AD vs MCI
Klöppel et al [31]	90	SVM	Whole brain	96	No	AD vs CN
Gerardin et al [18]	71	SVM	SPHARM coefficients	94	No	AD vs CN
Magnin et al [38]	38	SVM	ROI	94.5	No	AD vs CN
Misra et al [43]	103	SVM	ROI	90.38	Yes	AD vs CN
Ramírez et al [50]	104	SVM	ROI	93.2	No	AD vs CN
Zhang et al [62]	202	SVM	ROI	93.2	Yes	AD vs CN
Illán et al [27]	79	SVM	Image factorization	96.91	No	AD vs CN
Graña et al [19]	45	SVM	Pearson's correlation	100	No	AD vs CN
Filipovych et al [15]	359	SVM	Diagnostic information	82.91	Yes	AD vs CN
Colliot et al [7]	74	Boostrap	Hippocampi	84	No	AD vs MCI
Ecker et al [13]	44	SVM	GM	86	No	AD vs CN
Querbes et al [49]	382	NTI	Cortical thickness	85	No	AD vs CN
Sun et al [56]	72	Logistic regression	Cortical GMD maps	86.1	Yes	AD vs CN
Rao et al [51]	129	Logistic regression	GM volume	85.26	No	AD vs CN
Iglesias et al [26]	120	KNN	SSO similarity	89	No	AD vs CN
Aggarwal et al [1]	60	Decision tree	Statistical coefficients	89.58	Yes	AD vs CN
Gray et al [20]	287	Random forest	MRI ROI, PET voxels	90	No	AD vs CN
Sateesh et al [55]	198	PBL-McRFN	VBM	77.56	Yes	AD vs CN
Liu et al [37]	643	Sparse coding	ROI	90.5	No	AD vs CN
Natarajan et al [46]	297	Bagging	ROI	76	Yes	AD vs CN
Morra et al [45]	70	AdaBoost	Hippocampi	84	Yes	AD vs CN

3.2 Feature

There are various of features could be extracted from the raw image. Some papers like [31] use just the whole brain voxel intensities, which means they do not do too much to cope with the data. However, most studies extracted different features, including the ROI, the hippocampus and the cortical thickness. The region of interest tends to be the GM, the WM, as well as the CSF. These features are the common favourites.

In addition, there are also some studies developing their own features. For example, Gerardin et al [18] extracted their own features, called spherical harmonics (SPHARM) coefficients. They used the SPHARM coefficients to model the shape of the hippocampus. It means they considered the geometrical information of the hippocampus, rather than the intensity or the volume.

Further more, general studies only take one modality of features such as MRI or PET into account. However, considering multi-modality features seems more significant because each modality has it advantage to detect some part of information. If we combine more than one modalities together, it might be more convincing. For instance, Gray et al [20] combined MRI features and PET features together. Later, in [21], they added CSF features and genetic features into the previous feature combination. Their results showed a notable success.

However, according to Cuingnet et al's study [9], the use of feature selection did not improve the performance but substantially increased the computation times. The idea of their work is that they thought it is difficult to compare between different papers because different paper use different data. Therefore they used the uniform data to test different works and drew the conclusion.

There is no doubt that feature extraction is computationally cost. Besides, if we only extract extremely abstract features, we may lose useful information. For example, if we chose the volume of hippocampus as the feature. We have the right part volume and the left part volume, which is a 2-dimensional vector. Compared with the whole brain, which contains millions of voxels, it is obvious that most information has been lost. In addition, for some popular features such as hippocampus, there are many papers studying at how to segment it from the brain. It is fair to state that there is a standard strategy of extraction although the extracting algorithms may differ. For those features which are not very popular, there is not be a mature extracting method. As a result, the following classification might be affected. Also, feature extraction should fit with the classifier selected. Specifically, some high dimension features are suitable for the efficient classifiers; otherwise it could be time-consuming.

3.3 Classifier

According to Table 3.1, it is clear that SVM is the most popular classifier while the remaining is consisted of other supervised learning methods, some ensemble learning algorithms, as well as some classifiers developed by the authors. Figure 3.1 shows the percentages that each classifier accounts for among all the papers we reviewed.

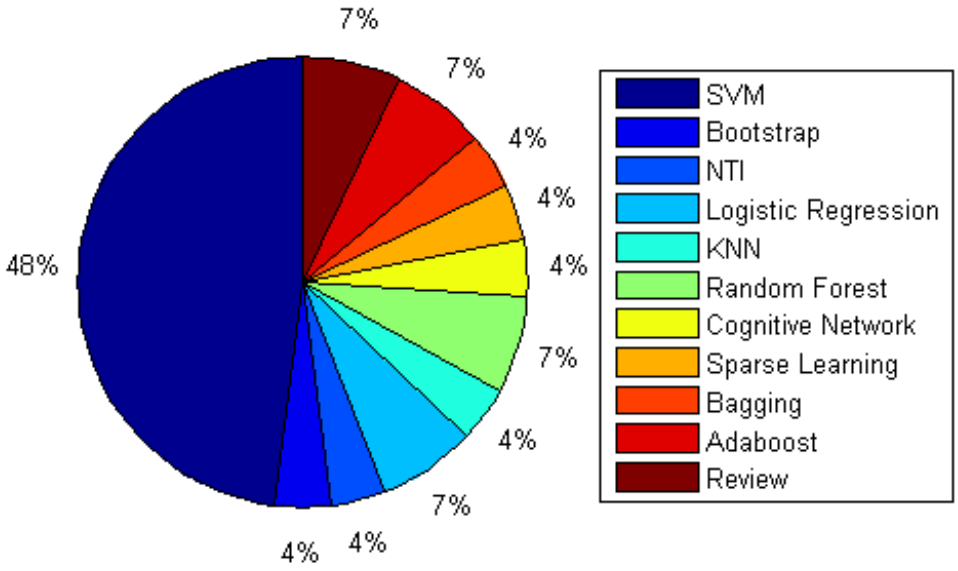


Figure 3.1: Classifiers. This pie chart shows the percentage that each paper reviewed accounts for. There are overall 11 kinds of classifiers we have reviewed. It is clear that the SVM is the most popular one.

Figure 3.1 shows that there are 48% papers utilizing SVM as the classifier among all the papers we reviewed. It is the binary classification in this case, which only needs to classify between AD and CN or AD and MCI. SVM is expert in this kind of classification. It could find out the furthest distance between two classes. In addition, kernel tricks in SVM can help to handle the condition where the gap between two classes is non-linear. Some studies not only used the standard SVM, they also did some modifications with it. The other reason why the SVM is popular is likely to be its efficiency. Mathematically, the SVM classification problem could be transformed into quadratic programming problems, which could be solved by mature numerical algorithms. The solutions do not need too many iterations and the computational complexity is not as high as other methods.

There are also some other supervised learning algorithms, including KNN, logistic regression, as well as decision tree, being used in classification. In our opinion, they are greatly limited. One of the most obvious shortages is that the corresponding features extracted from the raw data are commonly abstract. For instance, Rao et al [51] employed grey matter volume as the features, which is in very low dimension. As mentioned before, these sort of features cannot represent all the information of the raw data. However, these classifiers are difficult to handle the high dimensional problems. Concerning KNN, it is a lazy learning algorithm, which only considers other samples in certain distance. It can never cope with the effect of noise. Also, it requires large amount of memory space to run KNN. In terms of decision tree, the main problem is that it is easy to over-fit the data and the tree-pruning process is not simple to deal with. Similarly, the logistic regression cannot perform well in high dimensional features.

To reach the trade-off between the dimension of features and the computation of classifiers, some studies like [7], [46], and [45] combined some weak classifiers together, which is called ensemble learning. Its advantage is it is not easy to be misled by one classifier and combine many classifier

together to improve the final results. However, when combining different classifiers, there should be more space to store the examples compared with individual classifiers. In addition, ensemble classifiers like Adaboost call for numerous iterations [16], which is significantly time-consuming.

In addition, there are some papers, such as [49] and [55], developed their own classifiers. For Querbes et al, they measured the cortical thickness on the baseline MRI volume for each subject. The resulting cortical thickness map was parcellated into 22 regions and a normalized thickness index was computed using the subset of regions that optimally distinguished stable MCI from progressive MCI. Sateesh Babu et al [55] presented their novel approach with Voxel-Based Morphometry (VBM) detected features using a proposed "Projection-Based learning for Meta-cognitive Radial Basis Function Network (PBL-McRBFN)" classifier. McRBFN emulates human-like meta-cognitive learning principles. As each sample was presented to the network, McRBFN uses the estimated class label, the maximum hinge error and class-wise significance reduce the computational effort used in training. The similarity between those two classifiers is that both of them only can train the features extracted corresponding to themselves. It is difficult to build up an adaptive system for the whole process.

3.4 Accuracy

As we mentioned above, we picked up the best classification accuracy in Table 3.1 for each paper. According to the table, most accuracies are above 85%.

For classification between AD and CN, it usually around 90%. However, it is unbelievable that Fan et al [14] and Graña et al [19] achieved 100% accuracy. It is a common knowledge that there must be some trivial elements like noise will slightly effect the data, particularly in large dataset. Although they may not deeply affect the classification, people cannot obtain a perfect accuracy without any misclassification. One of the possibilities is that the number of participants in both of the two papers are fairly small. In [14], there were only 30 participant and in [19], there were only 45 participants. Few participants may not represent the global scene.

Except those two particular cases, the fact is to achieve high accuracy, there is no need to extract abstract features or develop complex classifiers. For instance, in [55], the authors developed a very complicated system to extract features and do the classification but their result was surprisingly low, which never reached 80%. For other studies like Zhang et al [62], they just use the ROI and SVM classifier but got 93.2% accuracy. Note that their had 202 participants, which is not fairly few.

It is interesting to note that feature selection plays a pivotal role in ensemble learning. Among our reviewed papers, only [20] and [21] got good accuracies because it combines multi-modalities and all the features extracted from different modalities are common. However, other ensemble methods, including [7], [46], and [45]. They employed simple features but never got encouraging results.

3.5 Conclusion

In conclusion, it cannot be fair if the number of training data is less than 300. In addition, features and classifiers influence with each other. Both of them have to depend on the original data. In our story, if the feature is extracted as voxels, the SVM and modified SVM are the most efficient classifiers. Not surprisingly, the classification accuracy between AD and CN could range from 85% to 90%. If more features are extracted and more complicated classifiers are selected, the accuracy may rise up to 95%. For the comparison group of sMCI and pMCI, the SVM classification accuracy would only be around 80% based on voxels.

4 Imaging Data

Data used in the preparation of this article were obained from the ADNI database (`http://adni.loni.ucla.edu/`). The ADNI was launched in 2003 by the National Institute on Aging (NIA), the National Institute of Biomedical Imaging and Bioengineering (NIBIB), the Food and Drug Administration (FDA), private pharmaceutical companies and non-profit organisations as a $60 million, five-year public-private partnership. The primary goal of ADNI has been to test whether serial MRI, PET, other biological markers, and clinical and neuropsychological assessment can be combined to measure the pregresion of MCI and early AD. Determination of sensitive and specific markers of very early AD progression is intended to aid researchers and clinicians to develop new treatments and monitor their effectiveness, as well as lessen the time and cost of clinical trials. The Principal Investigator of this initiative is Michael W. Weiner, M.D., VA Medical Center and University of California - San Francisco. ADNI is the result of efforts of many co-investigators from a broad range of academic institutions and private corporations, and subjects have been recruited from over 50 sites across the U.S. and Canada. The initial goal of ADNI was to recruit 800 adults, ages 55 to 90, to participate in the research - approximately 200 cognitively normal older individuals to be followed for three years, 400 people with MCI to be followed for three years, and 200 people with early AD to be followed for two years. Further up-to-date information, including detailed eligibility criteria is available on the ADNI information website (`http://www.adni-info.org`).

4.1 Participants

In this study, we will use a subset of the 796 MRI data from ADNI, which are all baseline MRI. There are more baseline MRI in ADNI. In addition, according to the latest data, we got some further information on the original data, especially for the pMCIs. As we know, the data of subjects collected by ADNI is traced every 6 to 12 months. Later, they would also be invited to the screening for several times. The pMCI subjects are assigned different labels according to the time point when they firstly assessed as AD. Specifically, one was assessed as baseline soon after his screening visit. His label is "bl". That means he converted to AD between the screening visit and the baseline visit. Since the time between screening and baseline is quite short, these patients were borderline-AD at screening. We have 4 "bl" subjects. Individuals who were detected that have converted to AD in 6 months were labelled as "m06". Similarly, we also obtained "m12", "m18", "m24", "m36", and "m48" subjects. Note that we only have 4 "bl" subjects and 2 "m48" subjects. To make each group balanced, we combine "bl"s and "m06"s together and that means subjects converting within 6 months. Similarly, we combine "m36" and "m48" cases together, meaning subjects converting within 3 to 4 years. Table 4.1 shows the number of participants in each group in detail. It is clear that there are 157 pMCI subject among the overall 796 individuals. However, there are 162 individuals in "m06", "m12", "m18", "m24", and "m36" Groups since there are a

small number of sMCIs being assessed as "m36" or "m48".

Table 4.1: Number of participants in each group.

Group	Number	Group	Number
AD	189	m06	26
pMCI	157	m12	46
sMCI	230	m18	32
CN	220	m24	32
		m36	26

4.2 MRI Acquisition

We downloaded the pre-processed baseline from the LONI Archive in NIfTI format. These had been acquired according to a standard protocal described in [28], involving two scans per subject that were based on a 3D MPRAGE imaging sequence and acquired using Siemens, GE and Philips MRI scanners.

4.3 Pre-processing

In this study, all the images used were skull-stripped using multi-atlas segmentation [34] and the intensity was normalized at a global scale using a piecewise linear function [48]. Intensity normalization was carried out following an iterative scheme, where all images are normalized to a common template/subject, then the template was changed and all the images were re-normalized to the new template. This was repeated N times, where N is the number of subjects to aid in removing normalization bias [8]. In this work, we set the subject whose ID is ADNI_ 002_ S_ 0295 as the common subject. The original image of it could be seen in Figure 4.1. Figure 4.2 shows the ADNI_ 002_ S_ 0295 that has been skull-stripped. Also, all images were transformed to a common space, the MNI152 template, which is shown in Figure 4.3, and hence re-sliced and re-sampled to and isotropic voxel size of 1mm. A coarse free-form-deformation [53], using a control point spacing of both 2.5mm and 10mm, was carried out to remove gross anatomical variability while aligning anatomical structures in order to focus on more local variation. Figure 4.4 displays the images that the subject of ADNI_ 002_ S_ 0295 registered to MNI template.

4.4 Feature Extraction

In our research, we chose voxels of the whole brain and around the hippocampus and the volumes of the hippocampus as our features. To extract the hippocampus, we projected a hippocampus mask to every image to obtain the hippocampus data of each subject. The mask, which is shown in Figure 4.5 was obtained by adding all ADNI segmentations in MNI space, then dilating the resulting mask. The segmentation of hippocampus of the ADNI images was described by Wolz et al[60], using LEAP, which is short for Learning Embeddings for Atlas Propagation. Figure 4.6

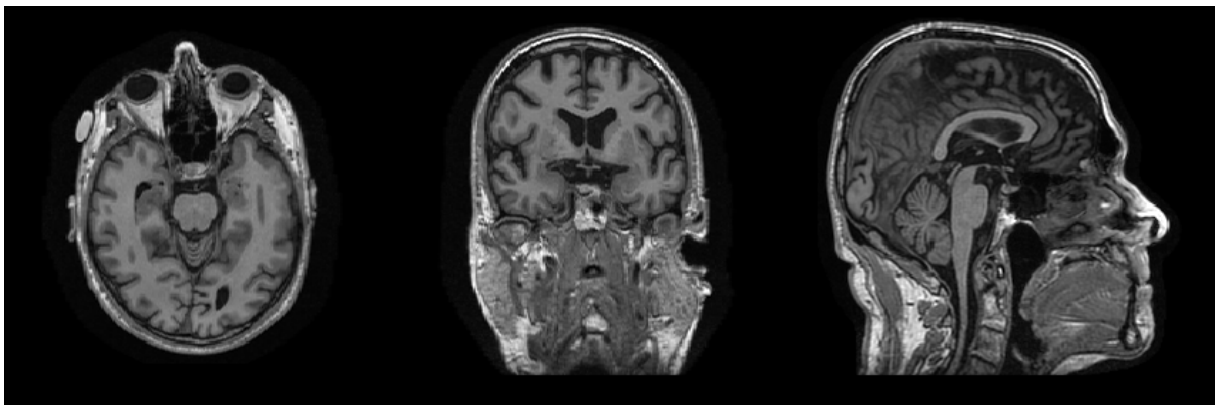


Figure 4.1: The original images of common subject ADNI_ 002_ S_ 0295. The subject is before skull-stripping and registration. The images from left to right are slices in axial, sagittal and coronal direction, respectively.

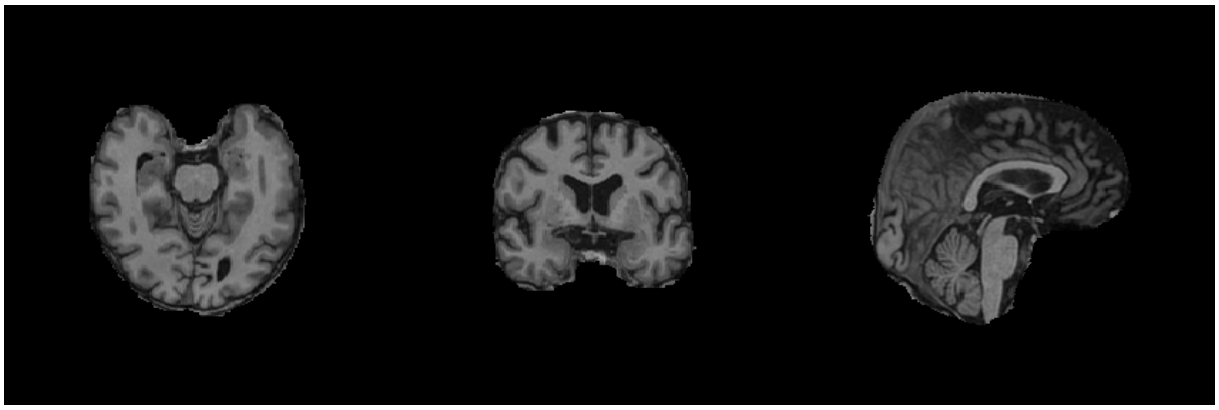


Figure 4.2: The skull-stripped images of subject ADNI_ 002_ S_ 0295. The images from left to right are slices in axial, sagittal and coronal direction, respectively.

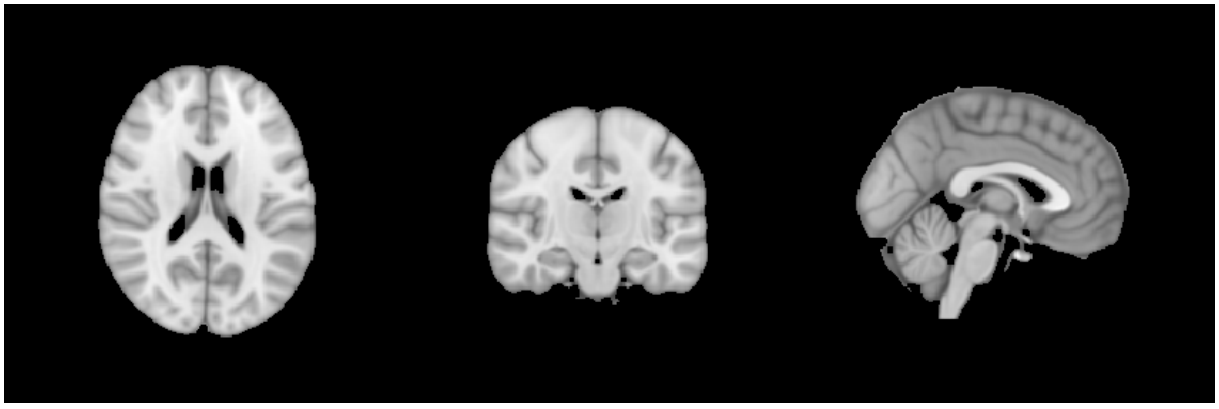


Figure 4.3: The MNI template. The images from left to right are slices in axial, sagittal and coronal direction, respectively.

shows the images of subject ADNI_ 002_ S_ 0295 with an overlay of the hippocampus segmentation in native space. That means the image has not been registered. After registration to MNI template, the images look like Figure 4.7. In addition, we also extract the volumes of hippocampus, which consists of the left and right halves of the brain according to LEAP. As a result, we employ both intensities and volumes of hippocampus of each image.

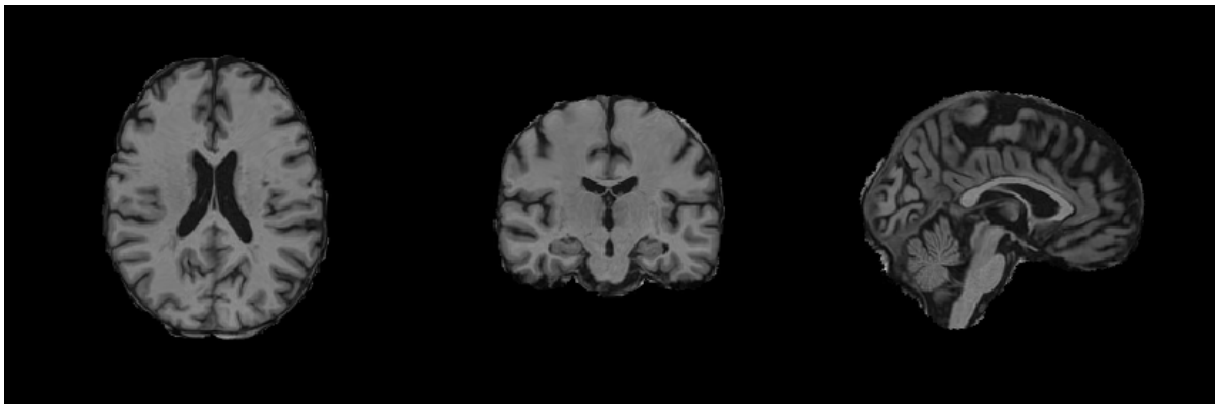


Figure 4.4: The subject of ADNI_ 002_ S_ 0295 registered to MNI template. The images from left to right are slices in axial, sagittal and coronal direction, respectively.

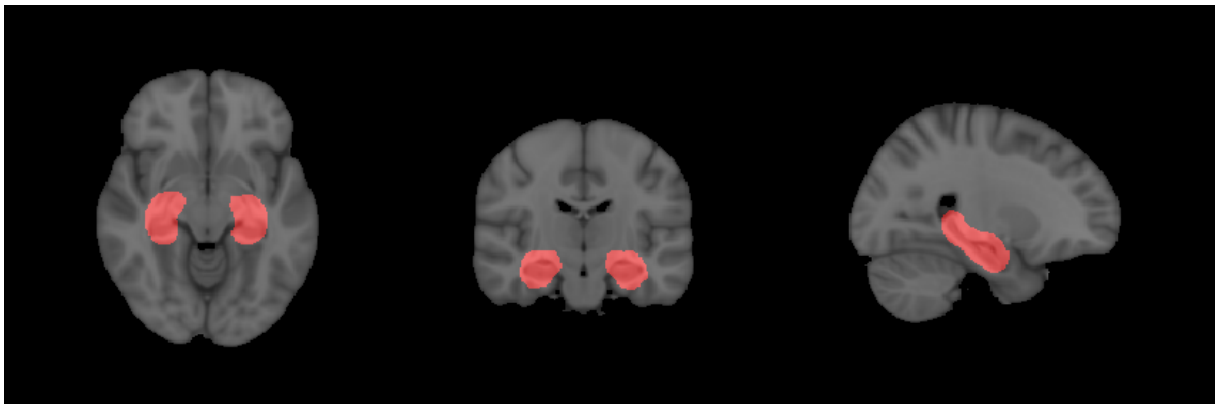


Figure 4.5: The mask with an overlay of the MNI template and the hippocampus region of interest. The red regions shown in images are the hippocampus. The images from left to right are slices in axial, sagittal and coronal direction, respectively.

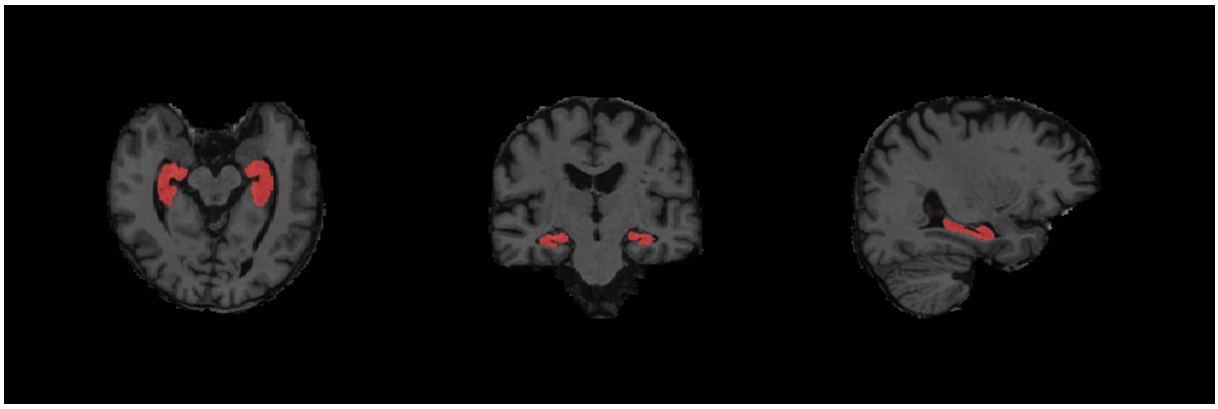


Figure 4.6: The images with an overlay of the hippocampus segmentation in native space. The red regions shown in images are hippocampus. The images from left to right are slices in axial, sagittal and coronal direction, respectively.

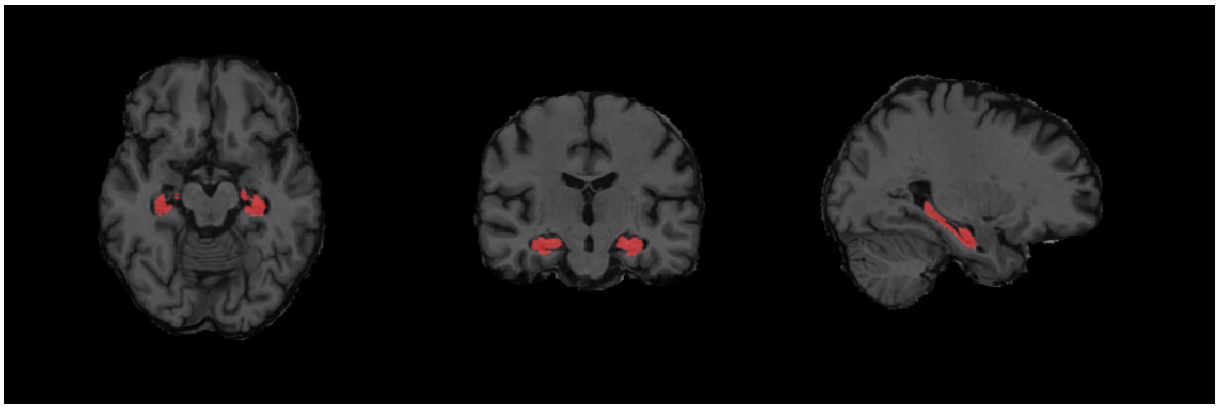


Figure 4.7: The images with an overlay of the hippocampus segmentation in MNI space. The red regions shown in images are hippocampus. The images from left to right are slices in axial, sagittal and coronal direction, respectively.

5 Statistical Data Analysis

In this chapter, we will do large-scale statistics on ADNI data. The features are voxels from whole brain and the ROI around hippocampus. The classifiers are linear and RBF SVM. Our final goal is to find whether there are a small number of instances, which are definitely diagnosed correctly. We will also compare the features and kernels of SVM. In addition, performance of different clinical sites will be assessed based on the results of statistics.

5.1 Feature Comparison

In this section, we use the voxels from both the whole brain and a ROI around the hippocampus as two kinds of features. According to our literature review in Chapter 3, SVM is the most popular classifier, which indicates that SVM is suitable for the classification between different data groups in AD MRI. Therefore, we employ SVM as our classifier in this part. In addition, two kinds of kernels of SVM are considered, which are linear and RBF kernels. Concerning the parameter γ in RBF kernel, we will manually tune it using cross-validation. In this section, leave-one-out cross-validation technique is used. The results of the classification focus on the sensitivity, the specificity, and the accuracy. Note that the number of examples in individual pMCI group is usually smaller than that in sMCI and CN groups. To handle the situation of unbalance, we sampled instances from the larger group to keep it balanced. These processes were repeated for 100 times and the results in the tables below are the mean numbers.

In the first stage, we compare between sMCI group and different pMCI groups, including the whole pMCI, m06, m12, m18, m24, and m36. The results are shown in Figure 5.1. According to the bar chart, it is obvious that the earlier the patient transferred to AD, the easier to classify them with sMCI. That means the faster their disease progressed, the more significant they distinguished from sMCI patients. In addition, the overall pMCIs can also distinguish with sMCIs at a high accuracy compared with individual pMCI groups. However, all the classification indexes are below 60% and some of them are even below 50%. For example, the m36 group versus sMCI group, all the three indexes are below 50%. That means there is meaningless to do this classification because it cannot be better than by chance. At this point of view, it is not significant to use linear SVM to classify the whole brain voxels to distinguish pMCIs and sMCIs.

Then we continue using linear SVM to classify CN group and different pMCI groups and AD group. The feature is still the whole brain voxels. The resulting bar chart are Figure 5.2. According to the results, the accuracy of AD vs CN exceeds 80% and the specificity even exceeds 90%. That means it is possible to use linear SVM to classify ADs and controls within respect of whole brain voxels. However, the classification results between CN and individual pMCI groups are still low, with tend to be between 60% and 70%. The worst results are still in m36 versus CN comparison,

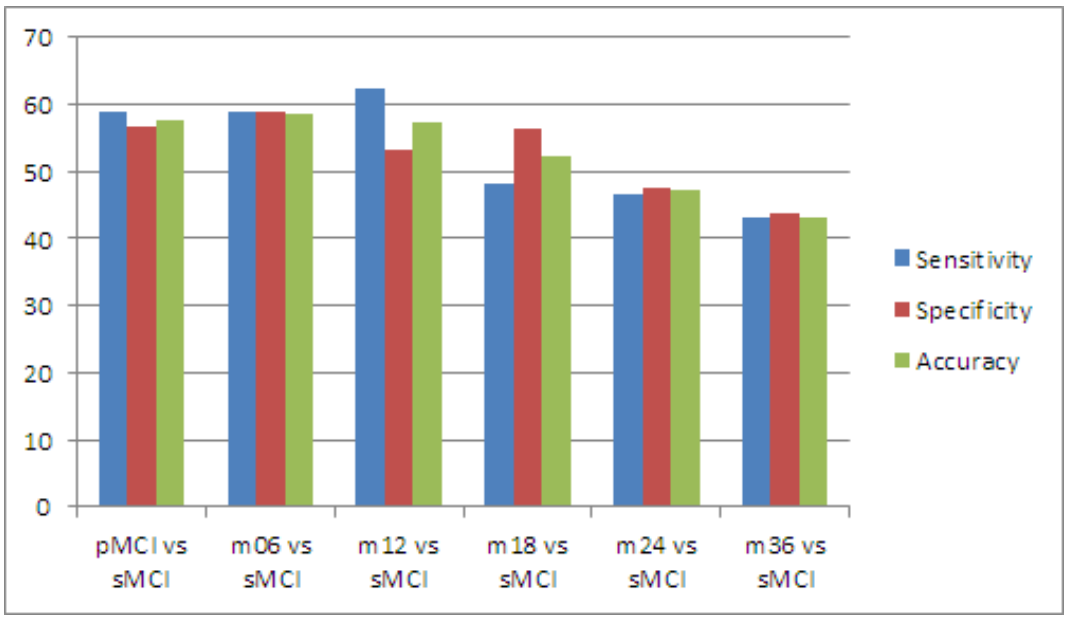


Figure 5.1: Linear SVM classification on sMCI group and pMCI groups in terms of whole brain voxels. The blue, red, and green bars stand for sensitivity, specificity and accuracy, respectively.

which even never reach 60%. It confirms the fact that the longer the patients converted to AD, the more difficult to recognize them.

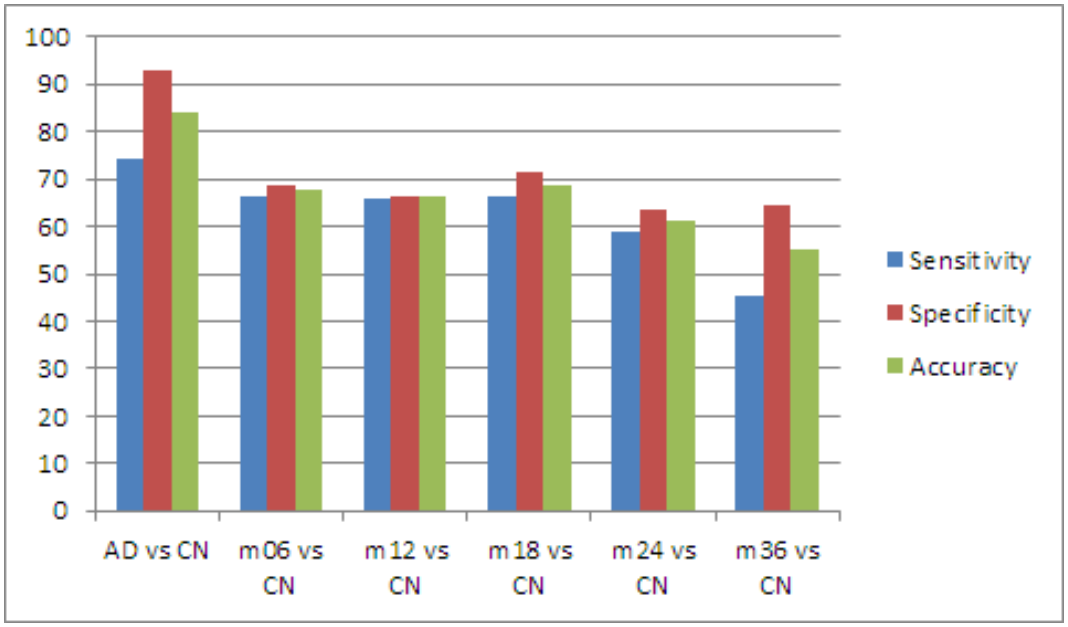


Figure 5.2: Linear SVM classification on CN group and pMCI groups and AD group in terms of whole brain voxels. The blue, red, and green bars stand for sensitivity, specificity and accuracy, respectively.

In the next step, we repeat the experiment above by using SVM classifier with RBF kernel instead of linear kernel. Figure 5.3 shows the results in terms of pMCI groups versus sMCI group. Compared with that classified by linear SVM, the overall trend of the results are similar, but the all the sensitivity, specificity and accuracy degrade faster when the conversion takes longer. Particularly, the accuracy of the m36 versus sMCI decreases to no more than 30%, which is absolutely

unacceptable. Likewise, RBF SVM is not suitable for classifying between sMCI group and pMCI group, either.

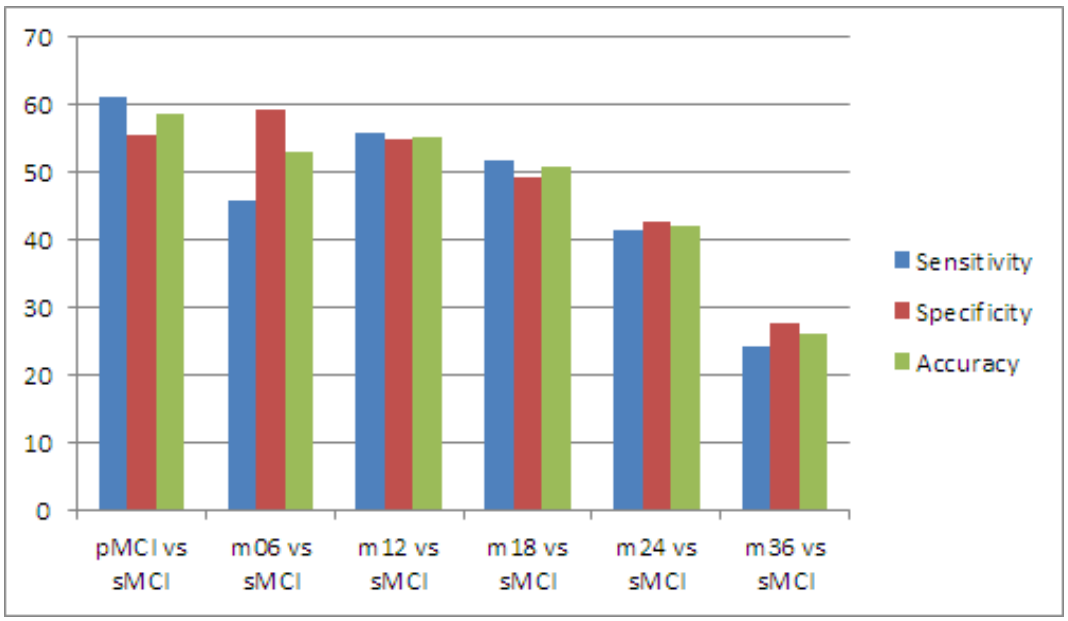


Figure 5.3: RBF SVM classification on sMCI group and pMCI groups in terms of whole brain voxels. The blue, red, and green bars stand for sensitivity, specificity and accuracy, respectively.

Figure 5.4 describes the results within respect of pMCI groups versus controls. Different with it in linear SVM, the accuracy of AD and CN is not as high as that in linear SVM, which is only 74.33%. However, results from other comparison groups are approximately flat with that in linear SVM except the last comparison group. The accuracy of m36 versus CN falls down below 50%, which is worse than chance.

Comparing the four groups of results, it is easy to find that the sensitivities, specificities and accuracies from pMCI groups and AD group versus CN group is slightly higher than they versus sMCI group. Intuitively, the more obvious to discover the difference between two classifying groups, the higher the results are. The difference between pMCIs and controls is more significant than sMCIs. The difference between ADs and controls is the most obvious such that this comparison group could achieve the highest results.

In the second stage, we repeated all the experiments above with the voxels of hippocampus, rather than those of whole brain. The classification results in terms of pMCI groups and sMCI group are in Figure 5.5. The trend of each index is similar to the experiments before. However, it is notable that sensitivities, specificities and accuracies are higher than before which typically are around 60%. Even the worst group, which is m36 versus sMCI, the specificity is over 50%. Although linear SVM also cannot be used in this case, there is a significant improvement when extracting hippocampus as the feature.

Not surprisingly, the results of CN group versus pMCI groups and AD group are better than before, which are shown in Figure 5.6. Almost all the accuracies have exceeded or are approaching

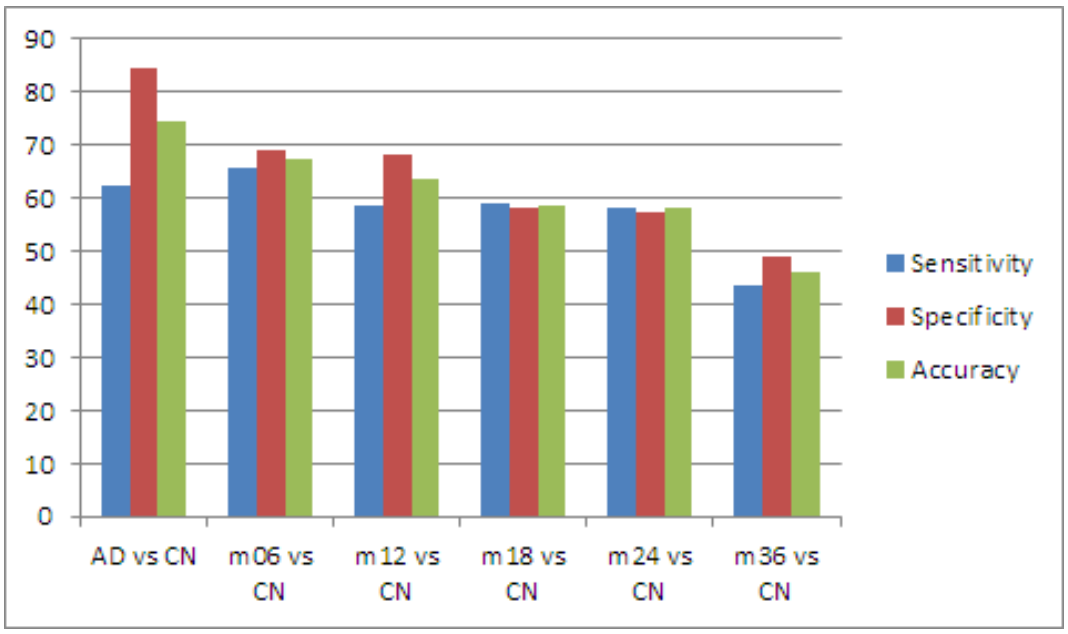


Figure 5.4: RBF SVM classification on CN group and pMCI groups and AD group in terms of whole brain voxels. The blue, red, and green bars stand for sensitivity, specificity and accuracy, respectively.

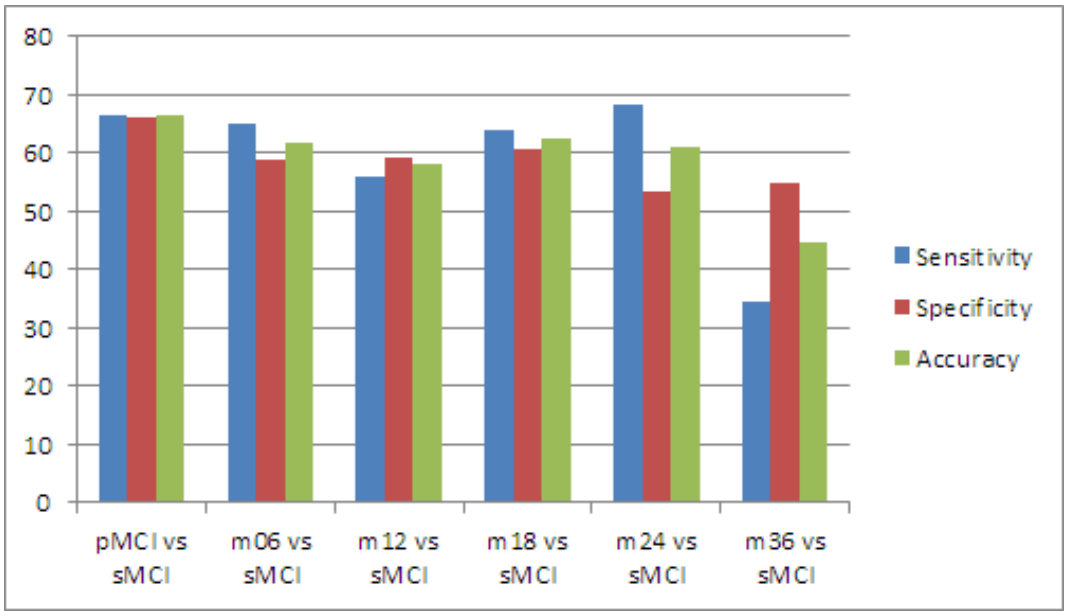


Figure 5.5: Linear SVM classification on sMCI group and pMCI groups in terms of hippocampus voxels. The blue, red, and green bars stand for sensitivity, specificity and accuracy, respectively.

80%, which seems to be a threshold in weighing whether the classification makes sense or not. Even the accuracy of comparison group of m36 versus CN is more than 60%, which is much higher than before which cannot reach 50%. Apart from it, the difference between early conversion groups and later conversion groups and CN group becomes less significant. Specifically, the classifier can also perform well on later conversion groups and controls.

In terms of the RBF SVM classification between sMCI group and pMCI groups on voxels of hippocampus, although the results are significantly improved than on whole brain voxels, which

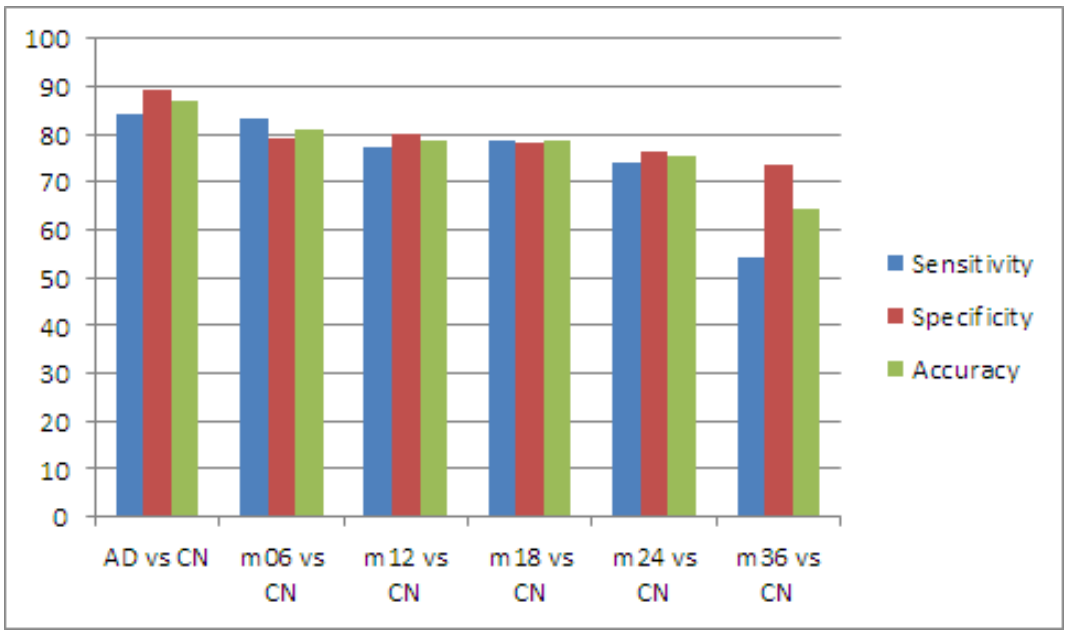


Figure 5.6: Linear SVM classification on CN group and pMCI groups and AD group in terms of hippocampus voxels. The blue, red, and green bars stand for sensitivity, specificity and accuracy, respectively.

is shown in Figure 5.7, the accuracies have not yet reached 70%. Among all the accuracies, the best one is 63.01% on whole pMCI group versus sMCI group. The worst one is still the m36 group versus sMCI group, whose accuracy even never reaches 50%. Likewise, the RBF SVM is also not feasible in classification between pMCI groups and sMCI group.

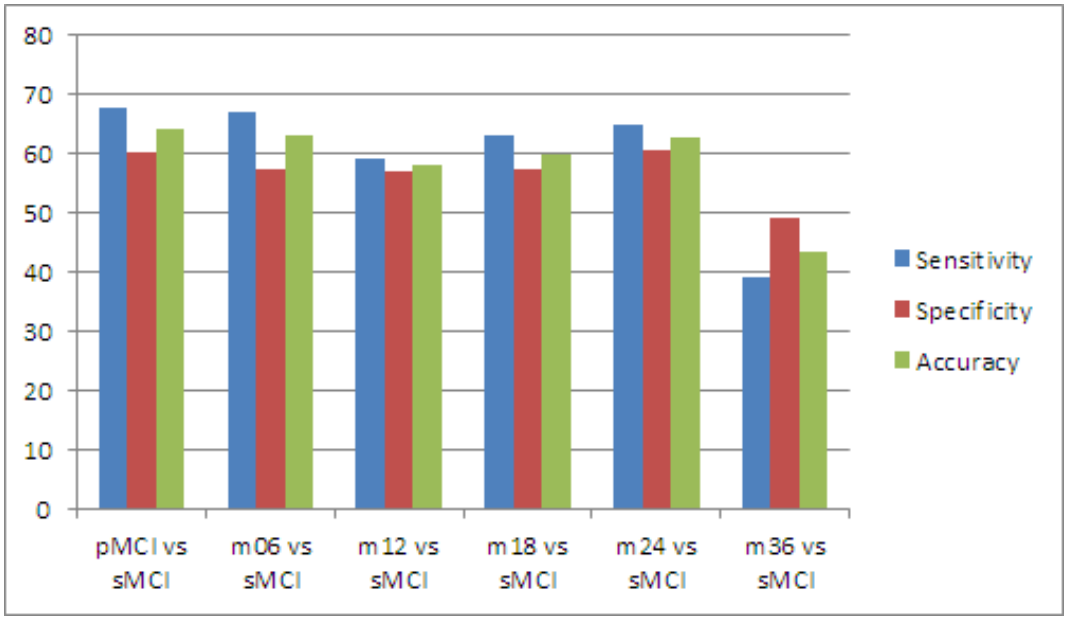


Figure 5.7: RBF SVM classification on sMCI group and pMCI groups in terms of hippocampus voxels. The blue, red, and green bars stand for sensitivity, specificity and accuracy, respectively.

The last series of classification is based on CN group and pMCI groups and AD group in terms of voxels of hippocampus. The results obtained are the best among these series of classification

experiments. For the best condition where the comparison group is AD versus CN, the sensitivity, specificity and accuracy are all over 80%. For other pMCI groups, including m06, m12, and m18, versus CN, the accuracies tower over 70% and even approach 80%. In spite of the worst comparison group, m36 versus CN, its accuracy is slightly more than 60%.

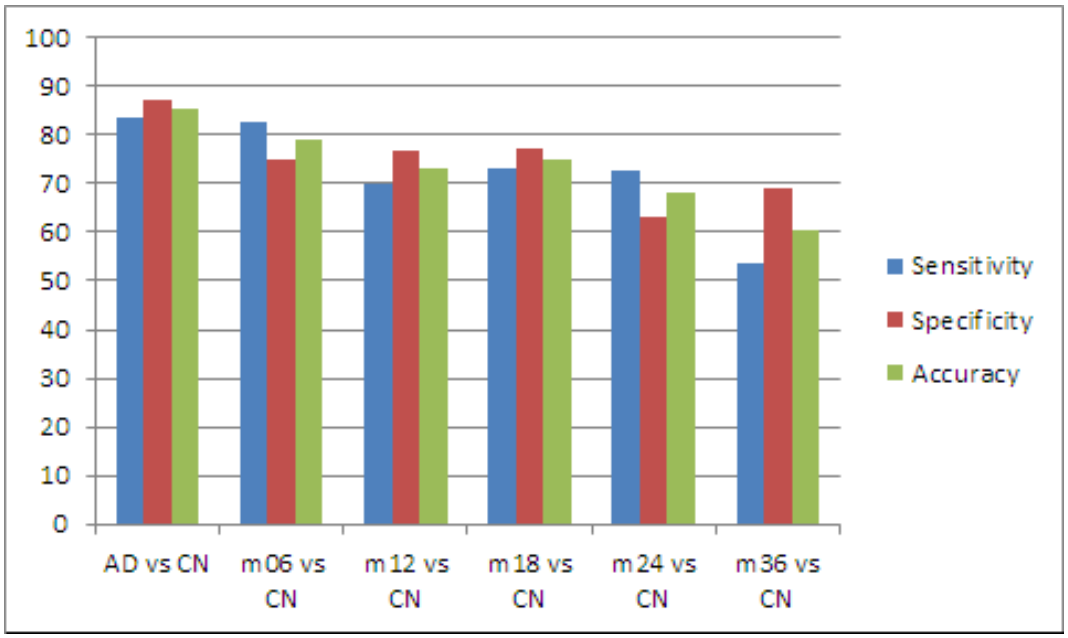


Figure 5.8: RBF SVM classification on CN group and pMCI groups and AD group in terms of hippocampus voxels. The blue, red, and green bars stand for sensitivity, specificity and accuracy, respectively.

To sum up, voxels of hippocampus perform better under all circumstances of classification than that of whole brain. It is expected that voxels in other parts of the brain may not play a pivotal role in AD. On one hand they are likely to act as noise in classification. On the other hand, one example is a vector whose length is nearly 2 million when it features as a whole brain but if it is from a pair of hippocampus, the length would be more than 30 thousands. The higher the dimension of training vectors, the more difficult to reach the trade-off in soft-margin SVM.

In terms of classification results, there is no significant advantages for RBF SVM. As we mentioned before, RBF kernel increases the amount of computation because we need to map the original data into higher space. Besides, the parameter γ in RBF is necessary to be tuned manually. If it cannot tower over linear SVM, it is not necessary to pay more to do it. However, whether RBF kernel is better or worse than linear kernel, it should have more evidence.

5.2 Kernel Comparison

In this section, we will study on the correct rate of individual instances. The comparison groups will be AD and CN and sMCI and pMCI. To achieve the balance between two classes, we will still sample numbers of examples from two classes. For each iteration, we will divide the training examples into 2 and 10 folds randomly, one of which will be used as the validation set. That means we adopt two strategies, including leave-10%-off and leave-50%-off. We will overall run 1000

iterations. As a result, for leave-10%-off method, each instance could be tested around 100 times and for for leave-50%-off method, each instance could be tested around 500 times. Therefore, for each instance, we may calculate its classification correct percentage. We would like to use the histograms to compare linear and RBF kernel. Specifically, we would divided the accuracy region, which is from 0 to 1, into 20 parts. In each small region, such as 0 to 0.05, we count the number of instances whose correct accuracies are in this small region. As a result, we could obtain a global distribution of correct accuracy.

We use histograms to show the distribution of the correct accuracies in terms of different classification groups. Figure 5.9 is the histogram, which focuses on the classification between AD and CN groups on whole brain voxels. The strategy is leave-10%-out. It indicates that most instances are absolutely classified as correctness. There are a small number of instances have never been correctly classified. Extremely few examples are hardly to be correctly classified or mis-classified. Through this figure, we can see that RBF kernel increases the frequency of instances that never been correctly classified and meanwhile decreases the frequency of instances been perfectly classified.

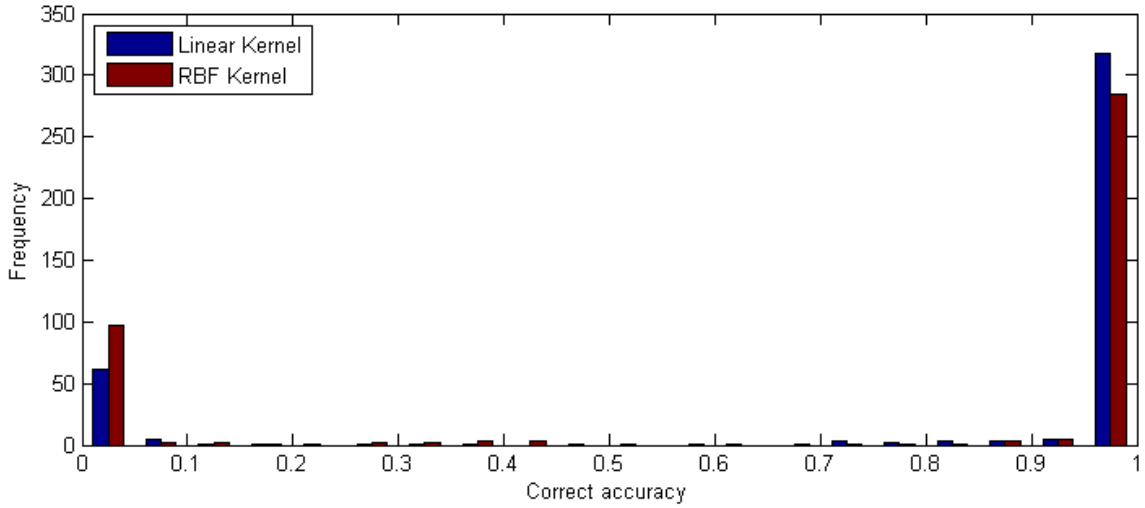


Figure 5.9: Histogram of classification on individual data point between AD and CN groups. 10% of the data points are set as validation. The feature voxels come from the whole brain. The blue bar stands for the results of linear kernel and the red bar represents that of RBF kernel.

If we adopt leave-50%-out strategy, seen in Figure 5.10, the overall trend never changed but it is interesting to note that the frequencies of instances whose accuracies are between 0 to 1 increase significantly. Intuitively, if more data points are left as validation, there must be more points being tested at one iteration. Once all iterations are completed, the testing times of each instance will significantly increase. Therefore the corresponding frequencies grow up significantly.

In terms of the comparison group of sMCI versus pMCI under the condition of leave-10%-out, seen in Figure 5.11, there are more instances that have never been correctly classified. Accordingly, there are less instances could be perfectly classified. In addition, more instances are classified with accuracies that slightly more than 0 and less than 1. The reason is likely to be that the difference between pMCIs and sMCI is not sufficiently obvious, which also results in that more instances have

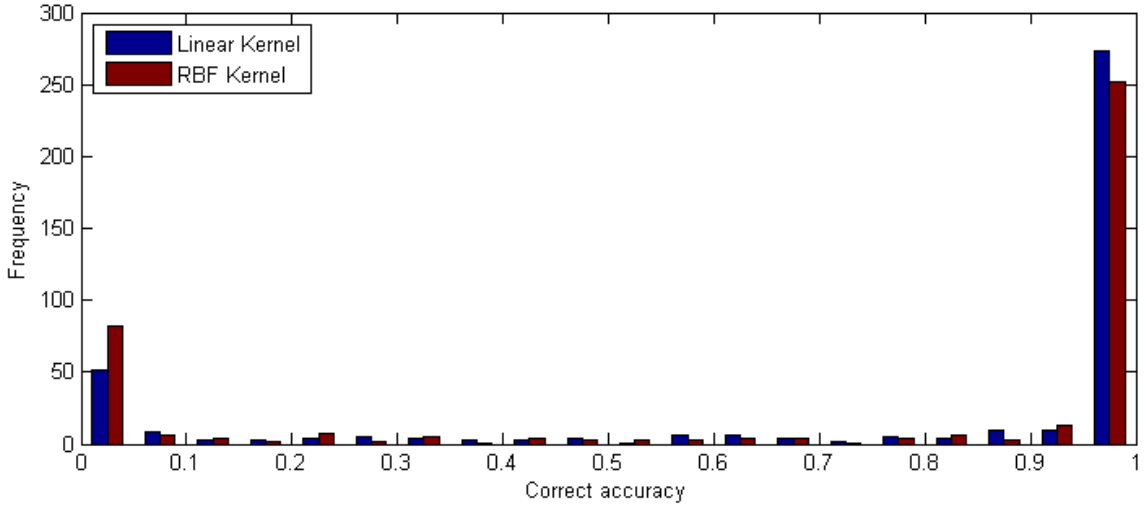


Figure 5.10: Histogram of classification on individual data point between AD and CN groups. 50% of the data points are set as validation. The feature voxels come from the whole brain. The blue bar stands for the results of linear kernel and the red bar represents that of RBF kernel.

accuracies between 0.1 to 0.9. In this scenario, RBF kernel still cannot perform better than the linear kernel.

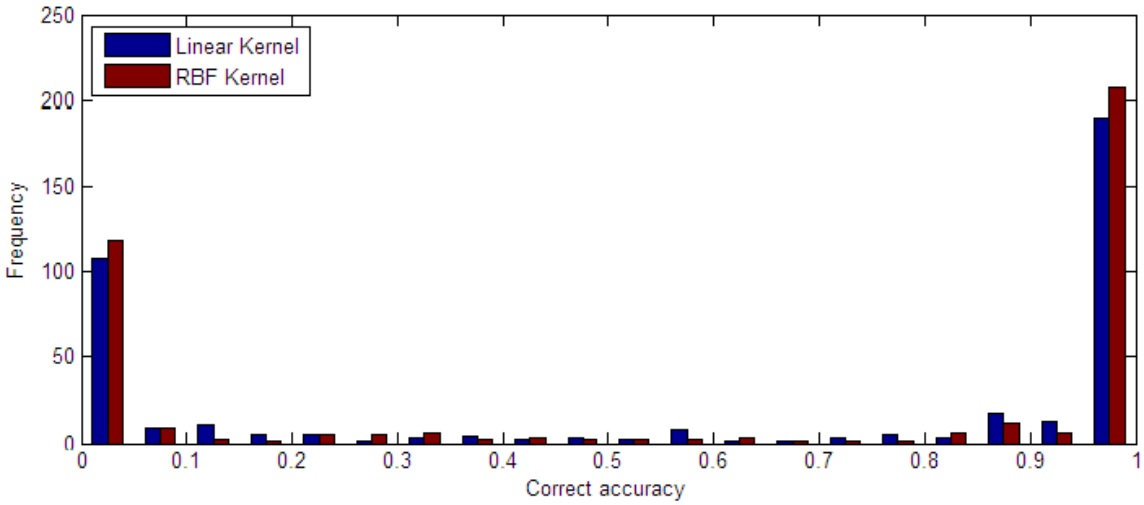


Figure 5.11: Histogram of classification on individual data point between sMCI and pMCI groups. 10% of the data points are set as validation. The feature voxels come from the whole brain. The blue bar stands for the results of linear kernel and the red bar represents that of RBF kernel.

As it is shown in Figure 5.12, there more instances being classified with 100% accuracy by RBF kernel than by linear kernel but there are also more examples being classified with 0% accuracy by RBF kernel than by linear kernel. In addition, when using leave-50%-out strategy, there would be a number of instances being classified with accuracies ranging from 0.1 to 0.9, which may also result from the unclear difference between sMCIs and pMCIs.

When using voxels of hippocampus, the fact changes significantly. As it is shown in Figure 5.13,

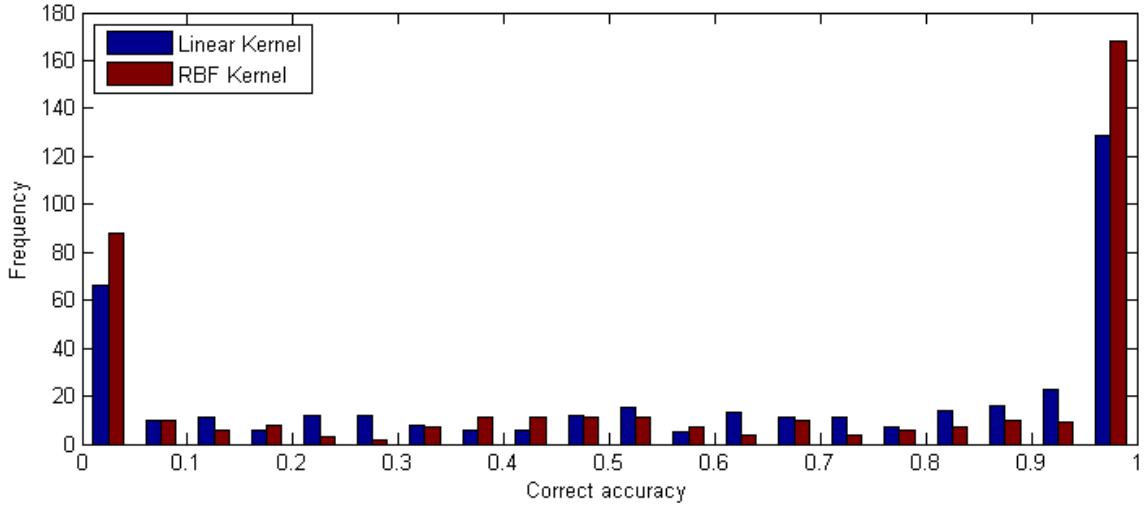


Figure 5.12: Histogram of classification on individual data point between sMCI and pMCI groups. 50% of the data points are set as validation. The feature voxels come from the whole brain. The blue bar stands for the results of linear kernel and the red bar represents that of RBF kernel.

almost all the instances are classified with 100% accuracy, where we perform leave-10%-out on AD and CN. There also a smaller number of instances that have never been classified correctly. Hardly any instances are classified with accuracies ranging from 0.1 to 0.9. Similarly, RBF kernel never takes more advantages than the linear kernel.

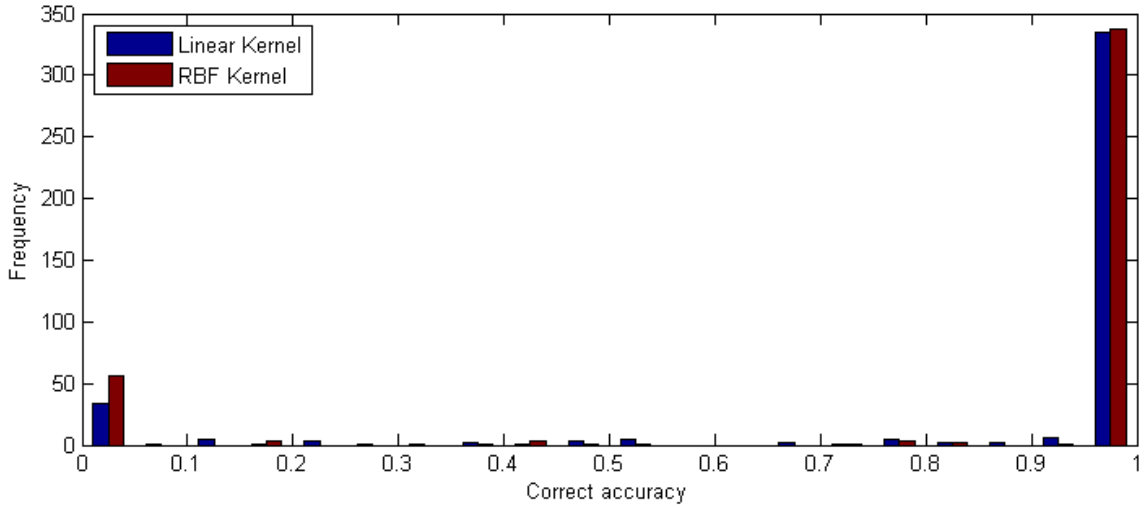


Figure 5.13: Histogram of classification on individual data point between AD and CN groups. 10% of the data points are set as validation. The feature voxels come from the hippocampus. The blue bar stands for the results of linear kernel and the red bar represents that of RBF kernel.

If we take leave-50%-out method to repeat the experiments shown in Figure 5.13, we could obtain the results in Figure 5.14. The distribution never changes significantly. The increase of the testing times of each instance only results in that less instances are classified with 100% accuracy and correspondingly the number of instances with other accuracies grows up. The RBF kernel still never show its advantages.

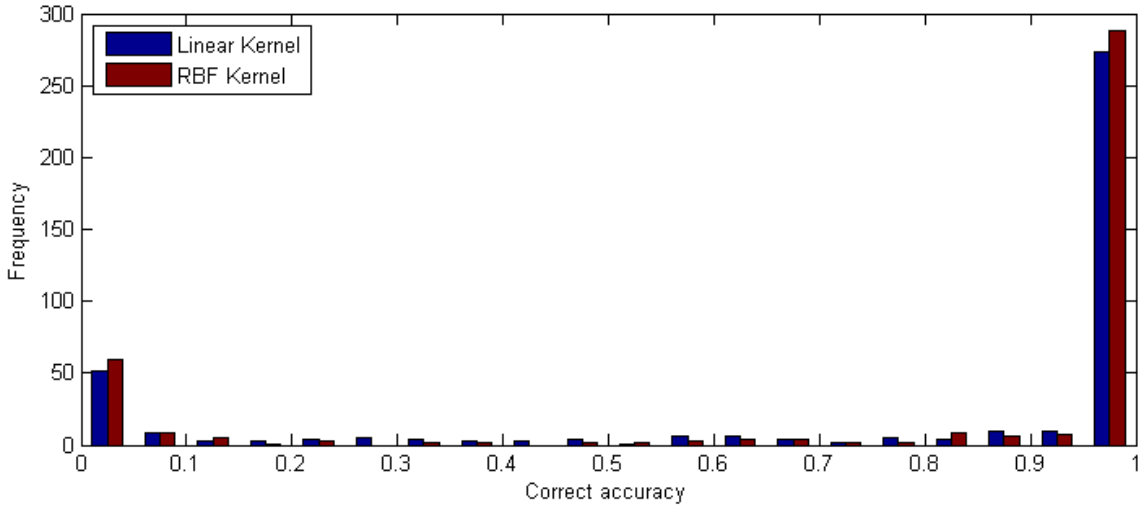


Figure 5.14: Histogram of classification on individual data point between AD and CN groups. 50% of the data points are set as validation. The feature voxels come from the hippocampus. The blue bar stands for the results of linear kernel and the red bar represents that of RBF kernel.

In terms of sMCI and pMCI groups, it is similar to that in Figure 5.11 where the number of absolutely mis-classified examples increases significantly while that of definitely classified examples decreases manifestly. Apart from it, there still more instances with accuracy of 0 by RBF kernel than by linear kernel.

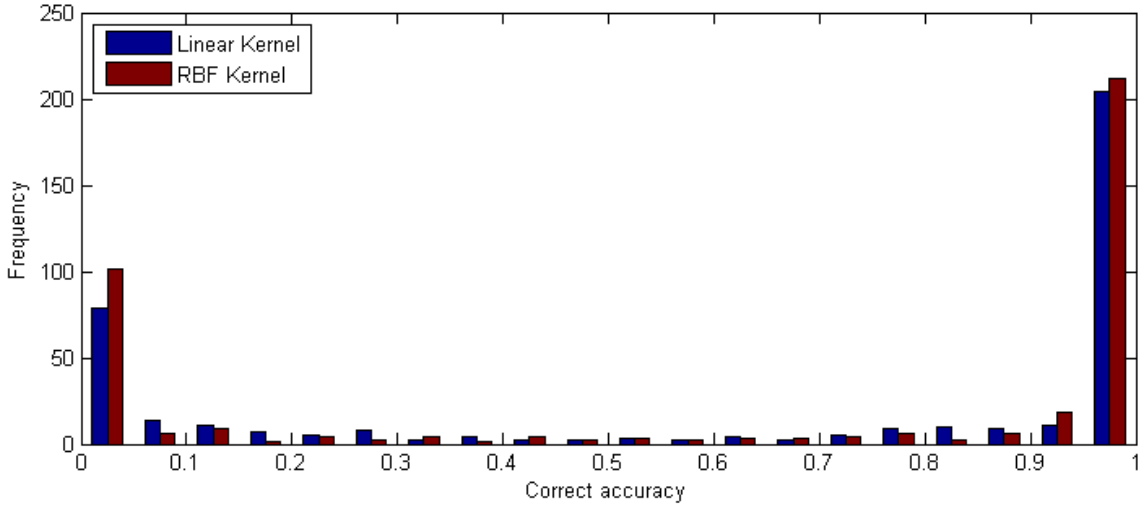


Figure 5.15: Histogram of classification on individual data point between sMCI and pMCI groups. 10% of the data points are set as validation. The feature voxels come from the hippocampus. The blue bar stands for the results of linear kernel and the red bar represents that of RBF kernel.

For the last series of experiments, represented by Figure 5.16, the disadvantage of RBF kernel appears more apparently. There are nearly 60 instances never being classified correctly by RBF kernel, which is approximately twice more than by linear kernel in case of sMCI versus pMCI with leave-50%-out strategy. However, for the instances never been mis-classified, RBF kernel performs

only slightly better than linear kernel.

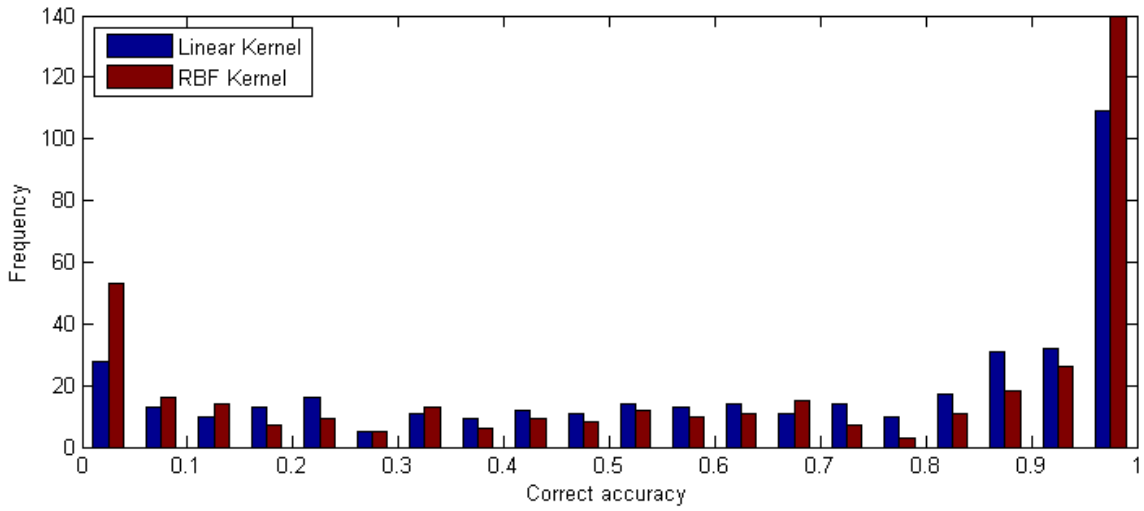


Figure 5.16: Histogram of classification on individual data point between sMCI and pMCI groups. 50% of the data points are set as validation. The feature voxels come from the hippocampus. The blue bar stands for the results of linear kernel and the red bar represents that of RBF kernel.

To sum up, admittedly RBF kernel may perform slightly better than linear kernel but in most cases, there is not any significant evidence that can show the advantage of RBF kernel compared with linear kernel. It might be resulted from the manual adaptation of RBF parameter because it is difficult to reach a global optima and even a local optima. On the other hand, the manual adaptation boosts the computation cost. Therefore in the following experiments, we will only consider the linear SVM as our classifier.

5.3 Assessment on Clinical Sites

In this section, we will combine the achievements obtained from the above two sections. Firstly, the hippocampus voxels will be the features in the following experiments, which is also been confirmed in Section 5.2. Secondly, we will only use linear SVM as the classifier. Thirdly, both of the two sections above suggest that the comparison group AD versus CN is better than pMCI and sMCI for us to do assessment on different clinical sites because the overall classification accuracy in group AD versus CN is averagely higher. However, there is a trade-off between two testing strategies, which are leave-10%-out and leave-50%-out. In this scenario, we will still take them into account.

According to the numerous results achieved in Section 5.2, we can do statistics to assess the performance of various clinical sites. Firstly, we order the correct accuracies of individual AD and CN groups in terms of the codes of the clinical sites. It is easy to count the number of scans from each site. Concerning the examples in each site, if the correct accuracy of one example is lower than 20%, which is a threshold we set, then it is confident for us to state this example may not be classified correctly. That means the diagnostic label may be wrong. Otherwise, it is acceptable. Therefore, it is not difficult to calculate the acceptance rate of each clinical site. Specif-

ically, we can calculate how do the acceptable scans account for all the scans from one site. We do this in cases of both leave-10%-out and leave-50%-out testing. The results are shown in Figure 5.17.

According to Figure 5.17, most sites achieve 100% accuracies in both leave-10%-out and leave-50%-out testing strategies. Particularly, for sites like 941, 011 and 018, the difference of two strategies is significant. It may due to that the over-fitting occurs in case of leave-10%-out or leave-50%-out testing. Another reason is likely to be the numbers of instances from those sites are small such that the error may occur. For example, for clinical site Number 941, there are only 4 scans. In case of leave-10%-out testing, 2 of them are acceptable while in case of leave-50%-out testing, 3 of them are acceptable. However, for clinical site Number 035, both of the two testing strategies do not provide satisfactory results. Although it only has 4 scans, it is necessary for it to improve its scanning quality.

5.4 Conclusion

In conclusion, we have found that there are a number of ADs and controls that have never been mis-classified. In addition, voxels from ROI around hippocampus are more useful to distinguish between patients and healthy individuals. Based on the voxels the linear SVM performs more efficient than the RBF SVM. Moreover the difference between the AD vs CN group is more significant than that between the pMCI vs sMCI group. Besides, the images from almost all clinical sites are highly reliable.

Figure 5.17: Assessment on clinical sites. The abscissa lists the code of clinical site and the ordinate shows the accuracy of each site. The blue bars stand for the results obtained by leave-50%-out testing strategy and the red bars suggest that in case of leave-10%-out strategy.

6 ADNI MRI Label Correction via Multiple Instance Learning

6.1 Motivation

It still is a long time for AD to progress in terms of a patient. There is no doubt that the earlier the disease is detected, the better it is for the patient. Although there is no cure for AD patients currently, psychological or some related treatments can be conducted on them to slow down the progress. In AD diagnosis, empirical diagnosis is widely used because of the difficulties to distinguish between CN and AD or even MCI. For some advanced diagnosis, clinicians may compare the candidate's data with that in the database from some professional organizations like ADNI, which probably improves the accuracy of diagnosis. However, the data from the ADNI is also from the clinical sites, which cannot be definitely sure that there is not any mistakes. As a result, it is meaningful to inspect the data in ADNI dataset.

In this chapter, we will employ techniques of multiple instance learning to detect whether there are some data from ADNI being mis-labelled or not. Particularly, it will be the welfare for the potential AD patients, who are mis-labelled as CNs. They and their doctors could pay more attention for the potential disease.

6.2 Synthetic Data

The data used in this chapter is still the MRI from ADNI. We selected the volumes of the hippocampus as the feature. According to Chapter 5, the training data will be those who never been mis-classified in cases of both leave-10%-out and leave-50%-out in terms of linear SVM classification on voxels from hippocampus between AD and CN. We overall have 409 AD and CN examples. We obtain hippocampus volumes of 403 among them, which could be divided into two groups. One is the training group, which consists of 90 ADs and 96 CNs. The other one is the testing group composed of 217 instances.

Because of the small number of the training data, we cannot test the capability of models of multiple instance learning. Before the formal experiments on the real data. We need to simulate large amount of synthetic data to test the algorithms. On one hand, we could select a better one between the two candidates based on the synthetic data. On the other hand, for the target algorithm, we could use the synthetic data to find out whether it is able to detect the mis-labelled instances and discover the robustness of the algorithm.

Intuitively, to generate the synthetic data, we have to ensure that it obeys the same distribution with the real data in order that it is consistent to the real data. Therefore in the first stage, we guess that the training data is submitted to a 2-dimensional normal distribution. To prove it, we plot the QQ-plot in terms of the training data, which is shown in Figure 6.1. In statistics, a normal QQ-plot compares a set of data on the vertical axis to a statistical population on the horizontal axis, which provides a graphical view of how similar or different are two distributions. As both of the two dimensions of the data are well fitted according to the figure, it is confident for us to claim that the training data obey for a 2-dimensional normal distribution.

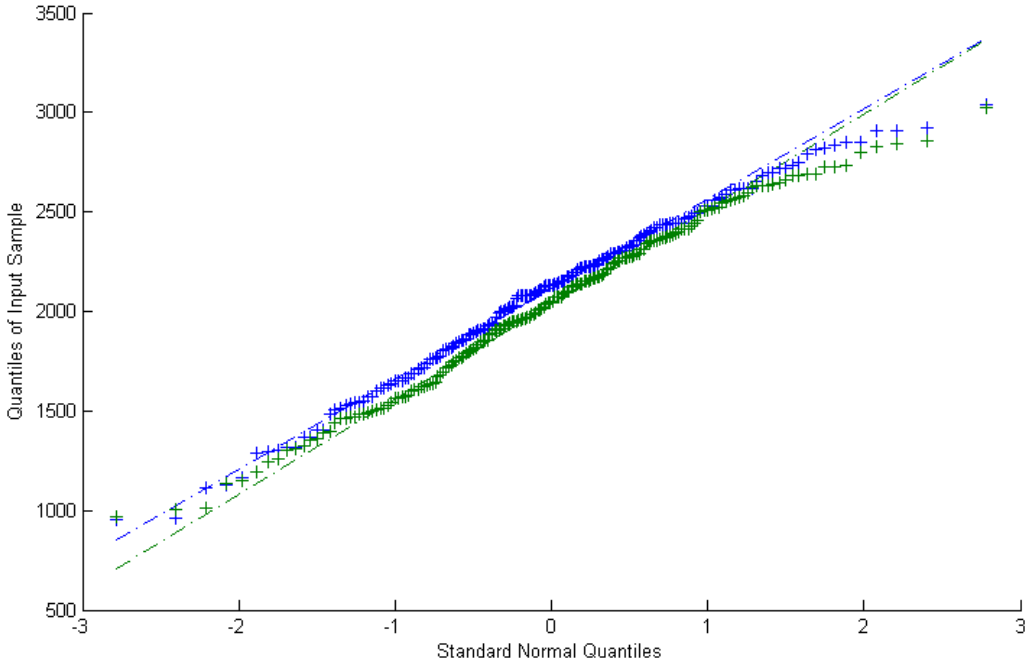


Figure 6.1: QQ plot of training data. The blue and green crosses stand for the data in two dimensions respectively. The blue and green dashed lines are the linear fitting lines to the two dimensions of data crosses respectively. Intuitively, the better the crosses are fitted by the dashed lines, the more they are submitted to normal distributions.

We firstly calculate the means and the covariance matrix of the AD and CN data respectively, which are

$$\begin{aligned} \mu_{AD} &= \begin{bmatrix} 1865.04 \\ 1770.73 \end{bmatrix} \quad \Sigma_{AD} = \begin{bmatrix} 173223.32 & 145861.97 \\ 145861.97 & 161376.55 \end{bmatrix} \\ \mu_{CN} &= \begin{bmatrix} 2318.17 \\ 2274.86 \end{bmatrix} \quad \Sigma_{CN} = \begin{bmatrix} 97459.08 & 80168.96 \\ 80168.96 & 89301.20 \end{bmatrix} \end{aligned} \tag{6.1}$$

According to the parameters calculated, it is possible to figure out the graphs of the probability density functions (PDF) of AD and CN, respectively. Figure 6.2 shows the two PDFs in the same graph. In the figure, we can see that hippocampus volumes of AD and CN obey similar normal distributions, whose shape is narrow and long, which could be reflected by the large covariance between the volumes from left and right parts of the brain. It indicates that the two parts of hippocampus are linked closely. On the other hand, it is obvious that hippocampus from healthy

individuals are larger than the patients in average, which could be seen by the fact that the peak belongs to CN is higher than that belongs to AD in Figure 6.2. In addition, the figure also reveals that the two distributions overlaps with each other manifestly, which may bring about interference to our following experiments.

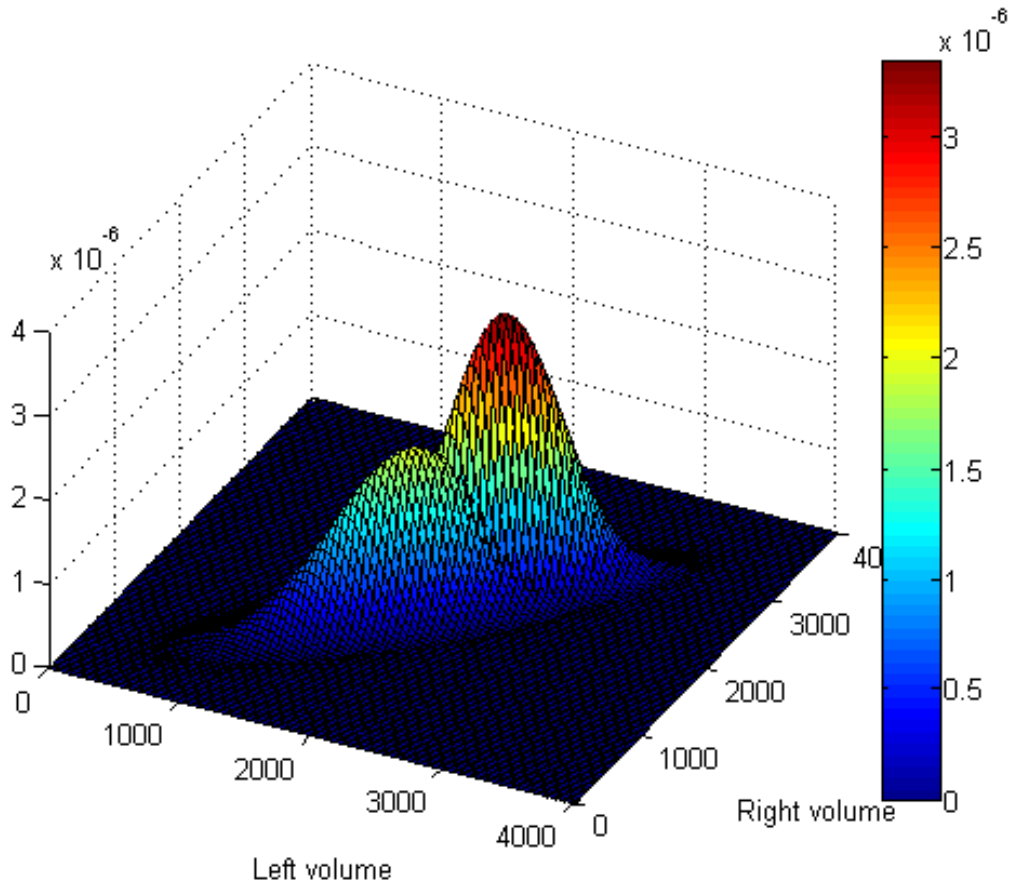


Figure 6.2: Probability density distribution of hippocampus volumes of AD and CN. Both of AD and CN obey for normal distributions whose shape is narrow and long. Obviously, the distribution with higher peak is density of CN because the average hippocampus volumes of healthy people are larger than the patients.

6.3 Model Comparison

Once obtaining the synthetic data, we can simulate a sufficient number of data in the experiments. In this chapter, two different algorithms are selected as candidates, which are Diverse Density and Citation-kNN. They have been introduced in Chapter 2. As we know there is not any machine learning technique that is able to cater for all the situations. A successful model has to be designed according to the data. In our case, we firstly have to test whether the two algorithms can work properly on our synthetic data or not.

In this part, we focus on testing the algorithms in two conditions. One is the best condition where there is no negative instances in positive bags. It means that positive bags will never affect the negative bags. The other one is the normal situation where there are both negative and

positive instances in positive bags. We wonder how do the two algorithms work under these two circumstances. However, in multiple instance learning, it is difficult to deploy the number of bags and that of instances. To achieve an overview, we will test various structures. The number of bags ranges from 10 to 50 and the number of instances is from 5 to 50. Therefore the minimum number of all instances is 50 and the maximum is 2500. In addition, we believe that the number is instances cannot be more than the number of bags. Note that both hippocampus volumes of AD and CN are submitted to similar normal distributions. It is believed that there is no significant difference in deciding setting AD or CN as negative. All the experiments in this section will set AD as negative.

Under the first condition where there is no negative instances in positive bags, the classification results are shown in Table 6.1 and Table 6.2, respectively. Leave-one-out cross-validation is used to validate the learning process. Table 6.1 reflects the classification performed by Citation-kNN. Almost all the accuracies are over 90%. Table 6.2 presents the results achieved from Diverse Density. The best accuracy is 80%, which is acceptable while the worst is just 48%, which is totally unsatisfactory. In this case, Citation-kNN is more suitable than Diverse Density for our data.

Table 6.1: Classification accuracies under different configurations by Citation-kNN. In this case, there is no negative instances in positive bags. The results are shown in percentage.

	5 instances	10 instances	20 instances	30 instances	50 instances
10 bags	100	100	-	-	-
20 bags	95	95	100	-	-
30 bags	96.67	96.67	93.33	90	-
50 bags	94	96	94	98	86

Table 6.2: Classification accuracies under different configurations by Diverse Density. In this case, there is no negative instances in positive bags. The results are shown in percentage.

	5 instances	10 instances	20 instances	30 instances	50 instances
10 bags	80	80	-	-	-
20 bags	85	85	55	-	-
30 bags	70	60	53.33	50	-
50 bags	62	50	50	48	50

In the second case where there are both positive and negative instances in positive bags, it is the normal structure in multiple instance learning. We randomly generate positive and negative instances for each positive bag but we ensure that there are at least one positive instance in each positive bag such that it satisfies the requirement of the definition. The results achieved by Citation-kNN and Diverse Density are displayed in Table 6.3 and Table 6.4, respectively. According to Table 6.3, the results in all kinds of structures are still exciting, which are even slightly better than the first condition. Only the structure of 50 bags, each of which contains 50 instances does not reach 80% accuracy. It is still normal while the other structures which may not contain too many instances is likely to be occurred by over-fitting. In addition, the overlap between AD and CN may also lead to it. On the other hand, the performance of Diverse Density is always discouraging. When the situation becomes worse, the accuracies it obtained also fall down significantly. According to Table 6.4, the best accuracy is only 60%. In spite of it, it occurs in the case of there

are only 10 bags, each of which only contain 5 instances, where over-fitting may occur.

Table 6.3: Classification accuracies under different configurations by Citation-kNN. In this case, positive bags contains random positive and negative instances. In addition, there is at least one positive instance in each positive bag. The results are shown in percentage.

	5 instances	10 instances	20 instances	30 instances	50 instances
10 bags	100	100	-	-	-
20 bags	100	95	90	-	-
30 bags	100	100	86.67	80	-
50 bags	100	100	100	80	78

Table 6.4: Classification accuracies under different configurations by Diverse Density. In this case, positive bags contains random positive and negative instances. In addition, there is at least one positive instance in each positive bag. The results are shown in percentage.

	5 instances	10 instances	20 instances	30 instances	50 instances
10 bags	60	50	-	-	-
20 bags	50	55	50	-	-
30 bags	43.33	53.33	50	50	-
50 bags	52	50	48	48	46

To sum up, Citation-kNN performs significantly better than Diverse Density in this case. Admittedly Diverse Density has its advantages in multiple instance learning but it never fits our data well. Therefore, we will choose Citation-kNN as our model in the following work.

6.4 Experiments on Synthetic Data

Before applying Citation-kNN to the real data to detect the potential mis-classified data, we have to test it more deeply because the results from previous experiment are not sufficient to clarify the its feasibility.

In the first stage, we continue the experiments in the section above. We wonder how does Citation-kNN work under the worst circumstance where there is only one positive instance in each positive bag. The classification results are displayed in Table 6.5. It is no wonder that the accuracies cannot be as good as before. Particularly, under the situation where there are overall 2500 instances, there are only 64% accuracy, which is unacceptable in reality. Fortunately, the number of our real data is small and the results shown in Table 6.5 reflect the high accuracy where there are not too many instances in bags, such as 5 instances and 10 instances. To this extent, Citation-kNN is feasible to the task.

In the second stage, we have to test whether Citation-kNN is possible to detect the mis-labelled instances. To handle this problem, we have to consider how to predict the label of one instance by using multiple instance learning. One idea may be put the instance in a bag and do the prediction for it. However, all the training bags contain a number of instances, which may lead to the unbalance if the testing bag only contains one instance. Our idea is to put the unseen instance

Table 6.5: Classification accuracies under different configurations by Citation-kNN. In this case, each positive bag only contains one positive instance. The results are shown in percentage.

	5 instances	10 instances	20 instances	30 instances	50 instances
10 bags	90	90	-	-	-
20 bags	100	75	75	-	-
30 bags	100	100	73.33	80	-
50 bags	100	100	100	70	64

into each negative bag and then classify it again. If the predicted label changes, then the unseen instance could be labelled as positive; otherwise it is negative. The experiments in this section above confirm that Citation-kNN has achieve a notable success in coping with the situation where there is only one positive instance in positive bags. When there is many negative bags, we can set a threshold ξ. Specifically, if the number of negative bags who changed their labels once being assigned the unseen instance is greater than ξ, then the unseen instance is positive; otherwise it is negative. For example, there are overall 10 bags, 5 of which are negative. Now an unseen instance needs to be classified and it is assigned to each negative bag respectively. For each negative bag, once being assigned of the unseen data, it is left out. Then this negative bag is to be classified based on the other bags. We set $\xi = 1$. That means if one negative bag changes its label when being assigned the unseen instance, the unseen instance is regarded as positive. If $\xi = 5$, then the unseen instance is negative unless all the negative bags change their labels when being assigned.

Considering the small number of our real data, we prefer the number of instances in each bag to be 5. Firstly, all the experiments based on Citation-kNN provide us with perfect accuracies where there are 5 instances in each bag. Secondly, for different numbers of bags, the overall number of instances is maximumly 250, which never exceeds the number of real data significantly. If it exceeds significantly, there must be several same instances in the bags, which may affect the classification. On the other hand, there is no significant difference in using different numbers of bags. As a result, combining classification results of different bags together may improve the global results further. According to our task, each instance to be classified has a previous label, which could be regarded as a reference. Once the instance having been classified, we could compare the new labels which are from where there are different numbers of bags with the reference. If all the four new labels are consistent with the previous one, then it is confident for us to state that this instance has never been mis-classified. Otherwise, we recognize it was mis-classified.

To set the experiments in this stage, we let each bag contain 5 instances and we will try 10 bags, 20 bags, 30 bags, as well as 50 bags. Under each situation, we firstly utilize all the synthetic data to train a Citation-kNN classifier. Then flip 10% of the overall number of instances out and set their label to the opposite. Thirdly, we adopt the classification strategy mentioned above to detect whether the classifier could sense the wrong labels. In this case, we set AD and CN as negative, respectively. The results are presented in Table 6.6. When AD is set as negative, it is 93.5% correct in detecting the mis-labelled CNs while the accuracy is only 12% accuracy in mis-labelled AD detection. It is similar when CN is set as negative. Therefore, we may set AD as negative to detect the instances whose previous label is CN and vice versa.

Table 6.6: Detection of mis-labelled instances. The first and second rows show the condition where AD and CN are set to negative respectively. In terms of the first column, we set all the instance to be detected as AD originally. In the second column, we set all the instances to be detected as CN in truth. The results are presented as mean accuracy (standard variance). All the numbers are in percentage.

	AD detection	CN detection
AD as negative	12 (17.43)	93.5 (7.45)
CN as negative	97.25 (8.96)	11.5 (12.58)

6.5 Experiments on Real Data

Experiments on synthetic data are successful and we are encouraged to apply the strategy mentioned in Section 6.4 to the real data. When applying it to the real data, it is helpful by considering the statistical results obtained in Chapter 5. As we mentioned in the beginning in this chapter, the training data are those who never been mis-classified in Chapter 5. It is also believed that the mis-classified probability of those who only achieved less than 20% accuracies in Chapter 5 under both testing strategies is high. As a result, we only consider those instances as potential mis-classified ones.

Applying our proposed method to the real data, we overall find 20 CNs and 10 ADs that might be mis-classified. In terms of the number, it makes sense since it is widely known that there about 10% mis-diagnosis in clinical. In our case, the percentage is 7.33%. Nevertheless, we should investigate the patients' clinical information, which is shown in Table 6.7. It is clear that there are 5 CNs converting to MCI at some time points, which could be regarded as mis-labelled subjects. Unfortunately, there are 11 individuals withdrawing the screening, which results in that is there no evidence to state whether they were mis-labelled or not. Particularly, we even cannot chase the following progress for the withdrawn CNs. More depressingly, some individuals suffered from other diseases like strokes and depression, which may interference the AD monitoring. There is one individual leaving no information to ADNI although he/she never withdraws the screening. The other one subject was diagnosed with mild confidence.

By way of conclusion, although our multiple instance model achieves notable success on synthetic data, its performance on real data still cannot be assessed objectively because of the missing of many patients' clinical information. In addition, if a person labelled as CN converted to MCI or even AD at some time point, then it is confident for us to claim that he/she might be in early stage of disease, which could not be realized. However, if one was diagnosed as AD, whose symptoms are not brought about by AD (he/she is mis-labelled), it is extremely difficult for us to verify the truth of prediction.

6.6 Discussion

In this section, we will discuss and try to analyse some interesting aspects in our work. Someone may propose that we have to normalize the hippocampus volumes. It is because the volumes of human brains vary between different individuals. Someone who has a larger brain, he/she may

Table 6.7: Mis-labelled instances detected. The first column presents the subject IDs. The second column displays their original labels. The third column shows the changes in diagnosis. The fourth column indicates when the candidates withdrew the screening. The last column gives some other comments.

Subject ID	Original label	Change in diagnosis	Withdrawal time point	Other comments
ADNI_002_S_0685	CN			
ADNI_003_S_0907	CN			
ADNI_007_S_0316	AD		Month 24	
ADNI_007_S_1304	AD		Month 24	Stroke(s) at month 12
ADNI_010_S_0419	CN			
ADNI_013_S_0592	AD		After baseline	
ADNI_013_S_1276	CN		Month 36	
ADNI_023_S_0963	CN		Passed away after month 36 (cancer)	
ADNI_023_S_1289	AD			No information after month 6
ADNI_027_S_0850	AD			Some depression
ADNI_029_S_0824	CN			Some depression
ADNI_029_S_0845	CN			
ADNI_032_S_1169	CN	Converted to MCI at month 60		
ADNI_033_S_0889	AD		Month 24	
ADNI_033_S_0920	CN			
ADNI_033_S_1098	CN			
ADNI_035_S_0048	CN		Month 36	
ADNI_035_S_0156	CN	Converted to MCI at month 84		
ADNI_036_S_0576	CN		Month 24	
ADNI_036_S_0759	AD			
ADNI_067_S_0059	CN	Converted to MCI at month 84		
ADNI_067_S_0812	AD			
ADNI_068_S_0210	CN	Converted to MCI at month 48		
ADNI_068_S_1191	CN		Month 6	
ADNI_098_S_0172	CN			
ADNI_098_S_0896	CN			Only mild confidence in diagnosis at month 6 and 12
ADNI_126_S_0606	AD			
ADNI_128_S_0740	AD		Month 12	
ADNI_137_S_0283	CN		Month 36	
ADNI_941_S_1202	CN	Converted to MCI at month 24		

have a larger hippocampus. Despite he/she suffers from AD and there is an atrophy on his/her hippocampus, it may still be larger than the normal ones'. Concerning this case, we study whether the normalized hippocampus volumes obey for normal distributions or not. We also draw the QQ plot for the normalized data, which is displayed in Figure 6.3. Obviously, after normalization the data is still submitted to normal distribution. Therefore we can use the proposed model to the normalized real data to detect the potential mis-classified subjects. However, the results are the same as before. Consequently, normalization does not affect the results in this case.

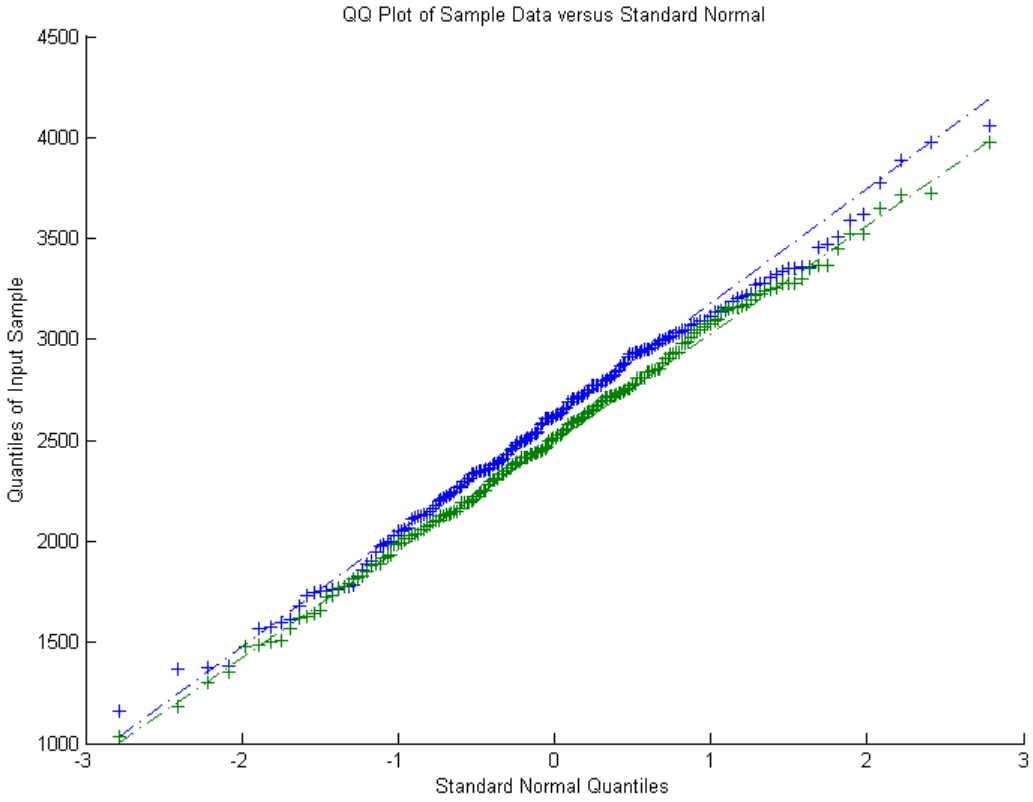


Figure 6.3: QQ plot of normalized training data. The blue and green crosses stand for the data in two dimensions respectively. The blue and green dashed lines are the linear fitting lines to the two dimensions of data crosses respectively. Intuitively, the better the crosses are fitted by the dashed lines, the more they are submitted to normal distributions.

In terms of model selection, we compared two algorithms, namely Diverse Density and Citation-kNN. According to our experiments on synthetic data, the Diverse Density failed to fit our data. Remember we mentioned in Chapter 2, the Diverse Density intends to find the intersections of positive bags, where they are believed to be the highest diverse density. For an unseen bag, if it comes across the intersections, it would be classified as positive; otherwise negative. However, Figure 6.2, there is a fairly overlap between hippocampus volumes of AD and CN, which may result in negative bags are able to come across the intersections. As a result, the negative bags are easier to be mis-classified. In addition, instances in our training data may not be viewed as a curved line as it is shown in Figure 2.3b because they are in 2 dimensions. Probably, there might be circles or crosses, which may also lead to mistakes.

On the other hand, Citation-kNN does not consider all the instances. Due to its property of lazy learning, it only take the most related bags into account. Particularly, unlike the elementary kNN algorithm, it cares all the instances related to the current bag. As shown in Figure 6.4, provided that the red circle is the current bag. If we do elementary kNN to it and $k = 3$, only the three blue ones will be considered. However, note that the red circle is also the nearest neighbour in terms of the yellow circle. Therefore, in Citation-kNN, the yellow circle is also taken into account. In this way, we are able to find all the instances that related to the current bag.

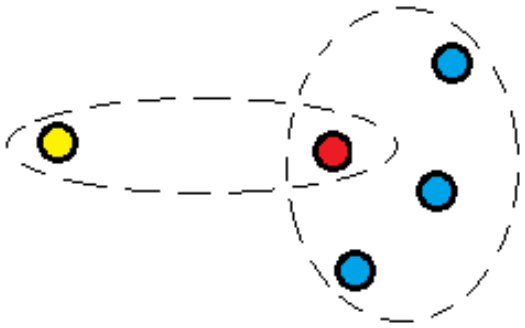


Figure 6.4: Sketch of Citation-kNN. Provided there are overall 5 bags, who are represented as circles. The red circle is the current one. The blue circles are the nearest neighbouts of the red one. The red circle is the nearest neighbour of the yellow circle.

However, although Citation-kNN has advantages mentioned above, it is not significantly robust. In kNN, it is difficult to decide the exact number of k. Similarly, in Citation-kNN, the two parameters ref and $cite$ cannot be fixed as both of them play pivotal roles in the whole algorithm. To handle the problem, we have to do grid search for each run. For example, we set the number is bags is 20 and the number of instances in each bag is 10 and generate the synthetic data to test the accuracies with different ref and $cite$. Table 6.8 shows the grid parameter search for the model. We can find that when $ref = 6$ and $cite = 15$ and $ref = 15$ and $cite = 17$, the classification accuracies are the highest, which is 95%. However, if $ref = 3$ and $cite = 16$ or $ref = 19$ and $cite = 14$, the accuracies is only 25%, which is the minimum. Additionally, there is not any significant patterns could be found according to Table 6.8. As a result, to achieve an acceptable classification result, it is compulsory to do the serious grid search each time.

Not surprisingly, when predicting the labels of unseen instances, we apply the idea of ensemble learning in our story. Specifically, we assign the unseen instances to each negative bag respectively and predict the label of each bag again. Finally we combine the results by voting strategy. Note that we set a threshold ξ in voting, which at our point of view, is also sensitive to the results. For example, we have 5 negative bags. In one case, there are 2 bags change their labels and the remaining do not. Under this condition, $\xi = 2$ or $\xi = 3$ is key to the prediction. If $\xi = 2$, the unseen instance will be predicted as positive while if $\xi = 3$, the prediction will be positive. There is no doubt that the smaller the ξ is, the more sensitive the classifier is and meanwhile the more mis-classification will occur. There is a trade-off between the number of ξ and the rate of accuracy. Hence, how to decide the number of ξ also needs careful consideration based on the real data.

In terms of the results, although we obtained higher accuracy in mis-classification detection on synthetic data, we cannot definitely confirm the subjects we found were mis-labelled yet. It is due to we do not have the absolutely accurate evaluation standard. For individuals who were labelled as CN, it is easier to examine our prediction. If he/she converted to MCI or even AD at the following screening, then our prediction is valid. However, if the subject withdrew the screening, there is no idea to validate it. Concerning the AD patients, although the previous diagnosis was wrong, it is difficult to validate our prediction because as he/she was detected as AD, he/she must have some symptoms related to AD, which may also obstruct our validation.

To apply our contribution to the real life, there is no doubt that it may assist the clinicians to avoid mistakes. Specifically, if a subject is diagnosed as CN but our model regards it as mis-classification, then the person himself/herself as well as his/her doctors should pay more attention to the following screening. It would be the best if the clinicians introduce more methods to synthetically diagnose the patient.

Table 6.8: Grid search for Citation-kNN. Each row presents one ref and each column corresponds to one $cite$, both of which ranges from 1 to 20 as we overall have 20 bags. There are 10 instances in each bag. All the numbers in the table are in percentage.

cite	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20
ref=1	65	45	45	50	40	45	55	70	65	80	55	75	70	60	60	65	65	80	55	60
ref=2	40	80	45	50	65	50	30	50	45	40	55	40	65	55	45	50	65	70	55	40
ref=3	60	65	65	50	70	60	60	30	70	40	45	60	85	65	75	70	60	70	55	65
ref=4	65	70	60	70	55	55	65	70	65	60	45	35	50	55	55	55	85	55	50	85
ref=5	50	70	60	40	30	50	65	55	60	50	50	85	75	70	70	80	90	60	80	90
ref=6	70	70	70	55	45	70	50	45	75	60	45	70	75	90	95	45	75	85	60	40
ref=7	65	55	45	65	45	65	80	50	60	45	75	50	55	60	65	45	70	40	50	55
ref=8	40	60	50	55	45	60	60	40	65	60	60	30	75	65	80	60	55	70	65	80
ref=9	55	75	55	70	75	55	65	65	60	75	85	75	45	85	60	70	80	75	80	70
ref=10	50	40	40	65	70	70	35	80	60	85	55	70	65	90	55	65	60	70	55	55
ref=11	60	65	55	55	50	75	65	50	75	70	70	45	60	50	50	75	45	75	90	50
ref=12	55	60	40	75	70	65	70	30	60	60	60	45	75	40	80	50	70	45	70	55
ref=13	75	60	55	50	55	45	55	75	80	70	75	65	70	35	65	65	70	65	55	70
ref=14	55	50	50	60	50	50	50	60	45	70	65	65	75	45	65	25	85	65	70	60
ref=15	55	60	50	75	90	60	50	55	80	70	55	65	65	70	70	65	95	50	70	55
ref=16	60	55	45	50	65	70	55	55	60	60	50	65	70	50	75	90	80	55	60	70
ref=17	35	65	55	55	75	45	65	60	85	50	60	65	60	70	50	40	50	60	75	70
ref=18	60	60	40	65	65	80	75	65	70	50	70	75	65	60	55	80	80	50	65	60
ref=19	60	40	25	70	65	65	60	70	60	50	85	50	50	55	65	60	65	60	75	65
ref=20	40	50	85	65	70	80	65	85	70	45	65	65	90	65	65	70	60	60	55	75

7 Conclusion

To draw an overall conclusion, we will firstly review what we prepared for this work and what we contributed to the problem. Then we will propose some recommendations for future work, which may solve the problem more robustly.

7.1 Contributions

From the beginning, we investigated the machine learning techniques and a number of papers concerning AD diagnosis in recent five years. According to the investigation, we determined the materials and general ideas of our work. These work are presented in Chapter 2 and Chapter 3.

In Chapter 5, we tested the 796 subjects from ADNI in terms of MRI. Based on the experiments, we found out that voxels of hippocampus play more significant roles than whole brain voxels. In addition, linear SVM is a feasible classifier in classification between AD and CN concerning the voxels of hippocampus. More importantly, we did large amounts of statistics to pick up a number of instances that may be definitely diagnosed as well as several potential mis-labelled ones. Besides, the statistical data helps up with the quantitative assessment on clinical sites who provide the patients data. The assessment results show that almost all the clinical sites scanned candidates with high quality. Only countable sites may not perform as well as expectation.

According to the statistical analysis done in Chapter 5, we tried to detect all the labels of instances who were potentially mis-labelled in Chapter 6. As the limited number of training data, we generate numerous synthesis data according to the distribution of the real data to validate the feasibility of the model. In this chapter, we compared the performance between two multiple instance learning algorithms, namely Diverse Density and Citation-kNN. The experiments confirmed Citation-kNN is more suitable for our data. Therefore we focused on testing Citation-kNN more deeply in the next step. We tested its classification capability under the worst circumstances, which gave out encouraging results. Further more, we flipped a small number of instances and reversed their labels. Then we combined Citation-kNN classifiers with different bags together to obtain an ensemble. The fact is the ensemble was able to detect almost all the mistakes, which proved that our model is useful in correction of wrong labels. Finally, we applied our method to the real data. It seems that the results are good. But because many subjects' clinical information is missing, we are unable to definitely evaluate the performance on the real data.

7.2 Future Work

This research has thrown up many questions in need of further investigation. First of all, the preprocessing, including segmentation, registration, and normalization needs to be put together. Specifically, it would be interesting to build up a comprehensive system where we input the raw images and we can obtain the features as the output. More importantly, the system should be sufficiently robust that we could set all the different parameters in one platform.

In addition, further investigation may focus on the user-friendly design. The current study did large scale statistics on the extracted features to collect some useful information, such as which individuals are the potential mis-classified ones. This work did not address the issue that the statistics could improve itself with the increase of the number of subjects. It is significant for clinicians to obtain the more accurate diagnosis when more patients attend the screening.

Moreover, features from more modalities worth our consideration. In this work, we only just volumes of hippocampus in the label correction, which is in single modality. PET imaging and biochemical markers from the blood and/or CSF may provide us with more useful information. Therefore, we can not only combine different structures of classifiers, but we can combine different feature modalities. [21] reported a notable success in AD and CN classification using multiple modalities. It is believed that multi-modality could play better than single modality.

Apart from that, it would be interesting to assess more algorithms to solve this problem. Although we have achieve satisfactory results, the model is not as robust as we expected. More research is required on the development of the model. On one hand, under the framework of multiple instance learning, it is possible to try more algorithms like mi-Graph [61]. Different multiple instance learning methods have different backgrounds. For instance, the Citation-kNN is based on kNN method. Similarly, the mi-Graph is based on graph models. Future trials should assess a number of other multiple instance learning methods and compare them with each other. A more robust algorithm or ensemble method might be found. On the other hand, more semi-supervised learning and weakly supervised learning methods should be considered. For example, it is interesting to assess the co-training method [4].

Bibliography

[1] Namita Aggarwal and R Agrawal. Computer Aided Diagnosis of Alzheimer's Disease from MRI Brain Images. *Image Analysis and Recognition*, pages 259–267, 2012.

[2] Yoshua Bengio. Learning deep architectures for AI. *Foundations and Trends in Machine Learning*, 2(1):1–127, 2009.

[3] Koji Tsuda Bernhard Scho olkopf and Jean-Philippe Vert, editors. *Kernel methods in computational biology*. Cambridge, MA: MIT Press, July 2004.

[4] A Blum and T Mitchell. Combining labeled and unlabeled data with co-training. In *Proceedings of the eleventh annual conference on computational learning theory*, pages 92–100, 1998.

[5] Ron Brookmeyer, Elizabeth Johnson, Kathryn Ziegler-Graham, and H Michael Arrighi. Forecasting the global burden of Alzheimer's disease. *Alzheimer's & dementia : the journal of the Alzheimer's Association*, 3(3):186–91, July 2007.

[6] Meena Chintamaneni and Manju Bhaskar. Biomarkers in Alzheimer's disease: a review. *ISRN pharmacology*, 2012, January 2012.

[7] Olivier Colliot, G Chételat, and M Chupin. Discrimination between Alzheimer Disease, Mild Cognitive Impairment, and Normal Aging by Using Automated Segmentation of the Hippocampus1. *Radiology*, 248(1):194–201, 2008.

[8] Pierrick Coupé, Simon F. Eskildsen, Jos V. Manjn, Vladimir S. Fonov, and D. Louis Collins. Simultaneous segmentation and grading of anatomical structures for patient's classification: Application to Alzheimer's disease. *NeuroImage*, 59(4):3736 – 3747, 2012.

[9] Rémi Cuingnet, Emilie Gerardin, Jérôme Tessieras, Guillaume Auzias, Stéphane Lehéricy, Marie-Odile Habert, Marie Chupin, Habib Benali, and Olivier Colliot. Automatic classification of patients with Alzheimer's disease from structural MRI: a comparison of ten methods using the ADNI database. *NeuroImage*, 56(2):766–81, May 2011.

[10] Christos Davatzikos, Yong Fan, Xiaoying Wu, Dinggang Shen, and Susan M Resnick. Detection of prodromal Alzheimer's disease via pattern classification of magnetic resonance imaging. *Neurobiology of aging*, 29(4):514–23, April 2008.

[11] TG Dietterich, RH Lathrop, and T Lozano-Pérez. Solving the multiple instance problem with axis-parallel rectangles. *Artificial Intelligence*, 89(1):31–71, 1997.

[12] Jianrui Ding, H D Cheng, Jianhua Huang, Jiafeng Liu, and Yingtao Zhang. Breast ultrasound image classification based on multiple-instance learning. *Journal of digital imaging*, 25(5):620–627, October 2012.

[13] Christine Ecker, Vanessa Rocha-Rego, Patrick Johnston, Janaina Mourao-Miranda, Andre Marquand, Eileen M Daly, Michael J Brammer, Clodagh Murphy, and Declan G Murphy. Investigating the predictive value of whole-brain structural MR scans in autism: a pattern classification approach. *NeuroImage*, 49(1):44–56, January 2010.

[14] Yong Fan, Susan M Resnick, Xiaoying Wu, and Christos Davatzikos. Structural and functional biomarkers of prodromal Alzheimer's disease: a high-dimensional pattern classification study. *NeuroImage*, 41(2):277–85, June 2008.

[15] Roman Filipovych and Christos Davatzikos. Semi-supervised pattern classification of medical images: application to mild cognitive impairment (MCI). *NeuroImage*, 55(3):1109–19, April 2011.

[16] Yoav Freund and RE Schapire. A decision-theoretic generalization of on-line learning and an application to boosting. In *Computational learning theory*, pages 23–37, 1997.

[17] G. B. Frisoni, R. Ganzola, E. Canu, U. Rub, F. B. Pizzini, F. Alessandrini, G. Zoccatelli, a. Beltramello, C. Caltagirone, and P. M. Thompson. Mapping local hippocampal changes in Alzheimer's disease and normal ageing with MRI at 3 Tesla. *Brain*, 131(12):3266–3276, May 2008.

[18] E Gerardin, G Chételat, and M Chupin. Multidimensional classification of hippocampal shape features discriminates Alzheimer's disease and mild cognitive impairment from normal aging. *Neuroimage*, 47(4):1476–1486, 2009.

[19] M Graña, M Termenon, a Savio, a Gonzalez-Pinto, J Echeveste, J M Pérez, and a Besga. Computer aided diagnosis system for Alzheimer disease using brain diffusion tensor imaging features selected by Pearson's correlation. *Neuroscience letters*, 502(3):225–9, September 2011.

[20] K Gray, Paul Aljabar, and R Heckemann. Random forest-based manifold learning for classification of imaging data in dementia. *Machine Learning in Medical Imaging*, pages 159–166, 2011.

[21] Katherine R Gray, Paul Aljabar, Rolf a Heckemann, Alexander Hammers, and Daniel Rueckert. Random forest-based similarity measures for multi-modal classification of Alzheimer's disease. *NeuroImage*, 65:167–175, January 2013.

[22] Katherine Rachel Gray. *Machine learning for image-based classification of Alzheimer's disease*. PhD thesis, Imperial College London, 2012.

[23] Harald Hampel, Richard Frank, Karl Broich, Stefan J Teipel, Russell G Katz, John Hardy, Karl Herholz, Arun L W Bokde, Frank Jessen, Yvonne C Hoessler, Wendy R Sanhai, Henrik Zetterberg, Janet Woodcock, and Kaj Blennow. Biomarkers for Alzheimer's disease: academic, industry and regulatory perspectives. *Nature reviews. Drug discovery*, 9(7):560–574, July 2010.

[24] Denise Harold, Richard Abraham, Paul Hollingworth, Rebecca Sims, Amy Gerrish, Marian L Hamshere, Jaspreet Singh Pahwa, Valentina Moskvina, Kimberley Dowzell, Amy Williams, Nicola Jones, Charlene Thomas, Alexandra Stretton, Angharad R Morgan, Simon Lovestone, John Powell, Petroula Proitsi, Michelle K Lupton, Carol Brayne, David C Rubinsztein, Michael

Gill, Brian Lawlor, Aoibhinn Lynch, Kevin Morgan, Kristelle S Brown, Peter a Passmore, David Craig, Bernadette McGuinness, Stephen Todd, Clive Holmes, David Mann, a David Smith, Seth Love, Patrick G Kehoe, John Hardy, Simon Mead, Nick Fox, Martin Rossor, John Collinge, Wolfgang Maier, Frank Jessen, Britta Schürmann, Hendrik van den Bussche, Isabella Heuser, Johannes Kornhuber, Jens Wiltfang, Martin Dichgans, Lutz Frölich, Harald Hampel, Michael Hüll, Dan Rujescu, Alison M Goate, John S K Kauwe, Carlos Cruchaga, Petra Nowotny, John C Morris, Kevin Mayo, Kristel Sleegers, Karolien Bettens, Sebastiaan Engelborghs, Peter P De Deyn, Christine Van Broeckhoven, Gill Livingston, Nicholas J Bass, Hugh Gurling, Andrew McQuillin, Rhian Gwilliam, Panagiotis Deloukas, Ammar Al-Chalabi, Christopher E Shaw, Magda Tsolaki, Andrew B Singleton, Rita Guerreiro, Thomas W Mühleisen, Markus M Nöthen, Susanne Moebus, Karl-Heinz Jöckel, Norman Klopp, H-Erich Wichmann, Minerva M Carrasquillo, V Shane Pankratz, Steven G Younkin, Peter a Holmans, Michael O'Donovan, Michael J Owen, and Julie Williams. Genome-wide association study identifies variants at CLU and PICALM associated with Alzheimer's disease. *Nature genetics*, 41(10):1088–93, October 2009.

[25] C Huang, L.-O Wahlund, T Dierks, P Julin, B Winblad, and V Jelic. Discrimination of Alzheimer's disease and mild cognitive impairment by equivalent EEG sources: a cross-sectional and longitudinal study. *Clinical Neurophysiology*, 111(11):1961–1967, November 2000.

[26] J Iglesias, Jiayan Jiang, CY Liu, and Zhuowen Tu. Classification of Alzheimer's Disease Using a Self-Smoothing Operator. *Medical Image Computing and Computer-Assisted Intervention-MICCAI 2011*, pages 58–65, 2011.

[27] I.a. Illán, J.M. Górriz, M.M. López, J. Ramírez, D. Salas-Gonzalez, F. Segovia, R. Chaves, and C.G. Puntonet. Computer aided diagnosis of Alzheimers disease using component based SVM. *Applied Soft Computing*, 11(2):2376–2382, March 2011.

[28] Clifford R Jack, Matt a Bernstein, Nick C Fox, Paul Thompson, Gene Alexander, Danielle Harvey, Bret Borowski, Paula J Britson, Jennifer L Whitwell, Chadwick Ward, Anders M Dale, Joel P Felmlee, Jeffrey L Gunter, Derek L G Hill, Ron Killiany, Norbert Schuff, Sabrina Fox-Bosetti, Chen Lin, Colin Studholme, Charles S DeCarli, Gunnar Krueger, Heidi a Ward, Gregory J Metzger, Katherine T Scott, Richard Mallozzi, Daniel Blezek, Joshua Levy, Josef P Debbins, Adam S Fleisher, Marilyn Albert, Robert Green, George Bartzokis, Gary Glover, John Mugler, and Michael W Weiner. The Alzheimer's Disease Neuroimaging Initiative (ADNI): MRI methods. *Journal of magnetic resonance imaging : JMRI*, 27(4):685–91, April 2008.

[29] Yangqing Jia and Changshui Zhang. Instance-level Semisupervised Multiple Instance Learning. In *Proceedings of the Twenty-Third AAAI Conference on Artificial Intelligence*, pages 640–645, 2008.

[30] ZS Khachaturian. Diagnosis of Alzheimer's disease. *Archives of Neurology*, 59(20):1203–1204, 1985.

[31] Stefan Klöppel, Cynthia M Stonnington, Carlton Chu, Bogdan Draganski, Rachael I Scahill, Jonathan D Rohrer, Nick C Fox, Clifford R Jack, John Ashburner, and Richard S J Frackowiak.

Automatic classification of MR scans in Alzheimer's disease. *Brain : a journal of neurology*, 131(3):681–689, March 2008.

[32] Gabriel Krummenacher, CS Ong, and JM Buhmann. Ellipsoidal Multiple Instance Learning. In *Proceedings of the 30th International Conference on Machine Learning*, volume 28, pages 73–81, 2013.

[33] Christian Leistner, Amir Saffari, and Horst Bischof. MIForests: multiple-instance learning with randomized trees. In *Computer VisionECCV 2010*, pages 29–42, 2010.

[34] K.K. Leung, J. Barnes, M. Modat, G.R. Ridgway, J.W. Bartlett, N.C. Fox, and S. Ourselin. Automated brain extraction using Multi-Atlas Propagation and Segmentation (MAPS). pages 2053 –2056, 2011.

[35] Thomas Leung, Yang Song, and John Zhang. Handling label noise in video classification via multiple instance learning. In *2011 International Conference on Computer Vision*, pages 2056–2063. Ieee, November 2011.

[36] Yan Li, David M.J. Tax, Robert P.W. Duin, and Marco Loog. Multiple-instance learning as a classifier combining problem. *Pattern Recognition*, 46(3):865–874, March 2013.

[37] Manhua Liu, Daoqiang Zhang, PT Yap, and Dinggang Shen. Tree-Guided Sparse Coding for Brain Disease Classification. *Medical Image Computing and Computer-Assisted Intervention-MICCAI 2012*, 15(61005024):239–247, 2012.

[38] Benoît Magnin, Lilia Mesrob, Serge Kinkingnéhun, Mélanie Pélégrini-Issac, Olivier Colliot, Marie Sarazin, Bruno Dubois, Stéphane Lehéricy, and Habib Benali. Support vector machine-based classification of Alzheimer's disease from whole-brain anatomical MRI. *Neuroradiology*, 51(2):73–83, February 2009.

[39] O Maron. *Learning from ambiguity*. PhD thesis, MASSACHUSETTS INSTITUTE OF TECHNOLOGY, 1998.

[40] O Maron and AL Ratan. Multiple-instance learning for natural scene classification. In *International Conference on Machine Learning*, volume 7, pages 341–349, January 1998.

[41] Oded Maron and T Lozano-Pérez. A framework for multiple-instance learning. *Advances in neural information processing systems*, pages 570–576, 1998.

[42] G. McKhann, D. Drachman, M. Folstein, R. Katzman, D. Price, and E. M. Stadlan. Clinical diagnosis of Alzheimer's disease: Report of the NINCDS-ADRDA Work Group* under the auspices of Department of Health and Human Services Task Force on Alzheimer's Disease. *Neurology*, 34(7):939–939, July 1984.

[43] C Misra, Y Fan, and C Davatzikos. Baseline and longitudinal patterns of brain atrophy in MCI patients, and their use in prediction of short-term conversion to AD: results from ADNI. *Neuroimage*, 44(4):1415–1422, 2009.

[44] Tom M. Mitchell. *Machine Learning*. McGraw-Hill, new edition edition, March 1997.

[45] Jonathan H Morra, Zhuowen Tu, Liana G Apostolova, Amity E Green, Arthur W Toga, and Paul M Thompson. Comparison of AdaBoost and support vector machines for detecting Alzheimer's disease through automated hippocampal segmentation. *IEEE transactions on medical imaging*, 29(1):30–43, January 2010.

[46] Sriraam Natarajan, Saket Joshi, Baidya N. Saha, Adam Edwards, Tushar Khot, Elizabeth Moody, Kristian Kersting, Christopher T. Whitlow, and Joseph a. Maldjian. A Machine Learning Pipeline for Three-Way Classification of Alzheimer Patients from Structural Magnetic Resonance Images of the Brain. In *2012 11th International Conference on Machine Learning and Applications*, pages 203–208. Ieee, December 2012.

[47] Dat T Nguyen, Cao D Nguyen, Rosalyn Hargraves, Lukasz a Kurgan, and Krzysztof J Cios. mi-DS: Multiple-Instance Learning Algorithm. *IEEE transactions on systems, man, and cybernetics. Part B, Cybernetics : a publication of the IEEE Systems, Man, and Cybernetics Society*, 43(1):143–154, June 2012.

[48] Lszl G. Nyl and Jayaram K. Udupa. On standardizing the mr image intensity scale. *Magnetic Resonance in Medicine*, 42(6):1072–1081, 1999.

[49] Olivier Querbes, Florent Aubry, Jérémie Pariente, Jean-Albert Lotterie, Jean-François Démonet, Véronique Duret, Michèle Puel, Isabelle Berry, Jean-Claude Fort, and Pierre Celsis. Early diagnosis of Alzheimer's disease using cortical thickness: impact of cognitive reserve. *Brain : a journal of neurology*, 132(8):2036–47, August 2009.

[50] J. Ramírez, J.M. Górriz, D. Salas-Gonzalez, a. Romero, M. López, I. Álvarez, and M. Gómez-Río. Computer-aided diagnosis of Alzheimers type dementia combining support vector machines and discriminant set of features. *Information Sciences*, May 2009.

[51] Anil Rao, Ying Lee, Achim Gass, and Andreas Monsch. Classification of Alzheimer's Disease from structural MRI using sparse logistic regression with optional spatial regularization. In *IEEE Engineering in Medicine and Biology Society. Conference*, volume 2011, pages 4499–502, January 2011.

[52] VC Raykar, B Krishnapuram, and J Bi. Bayesian multiple instance learning: automatic feature selection and inductive transfer. In *Proceedings of the 25th international conference on Machine learning. ACM*, pages 808–815, 2008.

[53] D Rueckert, L I Sonoda, C Hayes, D L Hill, M O Leach, and D J Hawkes. Nonrigid registration using free-form deformations: application to breast MR images. *IEEE transactions on medical imaging*, 18(8):712–21, August 1999.

[54] M Sano and C Ernesto. A controlled trial of selegiline, alpha-tocopherol, or both as treatment for Alzheimer's disease. *New England Journal of Medicine*, 336(17):1216–1222, 1997.

[55] G. Sateesh Babu, S. Suresh, and B. S. Mahanand. Alzheimer's disease detection using a Projection Based Learning Meta-cognitive RBF Network. In *The 2012 International Joint Conference on Neural Networks (IJCNN)*, pages 1–8. Ieee, June 2012.

[56] Daqiang Sun and TGM van Erp. Elucidating a magnetic resonance imaging-based neuroanatomic biomarker for psychosis: classification analysis using probabilistic brain atlas and machine. *Biological Psychiatry*, 66(11):1055–1060, 2009.

[57] Prashanthi Vemuri, Jeffrey L Gunter, Matthew L Senjem, Jennifer L Whitwell, Kejal Kantarci, David S Knopman, Bradley F Boeve, Ronald C Petersen, and Clifford R Jack. Alzheimer's disease diagnosis in individual subjects using structural MR images: validation studies. *NeuroImage*, 39(3):1186–97, February 2008.

[58] Jun Wang and JD Zucker. Solving multiple-instance problem: A lazy learning approach. pages 1119–1125, 2000.

[59] William T. Vetterling Wiiliam H. Press, Saul A. Teukolsky and Brain P. Flannery. *Numerical Recipes 3rd Edition: The Art of Scientific Computing*. Cambridge University Press, 3 edition, September 2007.

[60] Robin Wolz, Paul Aljabar, Joseph V. Hajnal, Alexander Hammers, and Daniel Rueckert. LEAP: Learning Embeddings for Atlas Propagation. *NeuroImage*, 49(2):1316–1325, 2010.

[61] Z.-H. Zhou, Y.-Y. Sun and Y.-F. Li. Multi-Instance Learning by Treating Instances. In *The 26th International Conference on Machine Learning*, pages 1249–1256, Montreal, 2009.

[62] Daoqiang Zhang, Yaping Wang, Luping Zhou, Hong Yuan, and Dinggang Shen. Multimodal classification of Alzheimer's disease and mild cognitive impairment. *NeuroImage*, 55(3):856–67, April 2011.

[63] Zhi-hua Zhou. Multi-instance learning: a survey. Technical report, National Laboratory for Novel Software Technology, Nanjing, 2004.