Automated MRI Prediction of Alzheimer's Disease Development by Machine Learning Methods

by

Meng Meng

(Under the direction of Khaled Rasheed)

Abstract

Alzheimer's is an irreversible brain disease that impairs memory, thinking and behavior and leads ultimately to death. Research has shown that individuals with MCI (mild cognitive impairment), the pre-stage of Alzheimer's, have an increased risk of developing Alzheimer's over the next few years [1]. It is useful and important to diagnose and predict MCI's conversion to Alzheimer's as early as possible for appropriate treatment. In our study, we use numerous machine learning, feature selection as well as clustering methods for this prediction purpose. High precision of prediction is observed for both 10-fold and 2-fold cross-validation. We also use L1 and L2-norm shrinkage terms to control the model complexity. As a result, the prediction error is reduced. These findings illustrate that machine learning methods accurately and reliably predict MCI's conversion, and potentially provide a great assistance to medical diagnosis.

Index words: Alzheimer's disease, MRI, Mild cognitive impairment, SVM, Logistic, Lasso, Loss function, Regularization

Automated MRI Prediction of Alzheimer's Disease Development by Machine Learning Methods

by

Meng Meng

B.S., Zhengzhou University, Zhengzhou, Henan, China, 2002

M.S., Shanghai Jiaotong University, Shanghai, China, 2005

A Thesis Submitted to the Graduate Faculty

of The University of Georgia in Partial Fulfillment

of the

Requirements for the Degree

Master of Science

Athens, Georgia

2011

© 2011

Meng Meng

All Rights Reserved

Automated MRI Prediction of Alzheimer's Disease Development by Machine Learning Methods

by

Meng Meng

Approved:

Major Professor: Khaled Rasheed

Committee: Tianming Liu
Hamid R. Arabnia

Electronic Version Approved:

Maureen Grasso
Dean of the Graduate School
The University of Georgia
November 2011

Dedication

This paper is dedicated to my son, Linus, who is the sunshine of my life.

Acknowledgments

First and foremost, I would like to express my sincere appreciation to my advisor, Dr. Rasheed, who gives me valuable instructions and helpful suggestions for my research. The thesis would not have been possible without his helps.

Secondly, I'd like to thank Dr. Liu for his well appreciated support and assistance. I would also like to thank Dr. Arabnia for being a member of my thesis committee.

Last but not least, I'd like to thank my wife, Siyan Hu, for her constant support and beliefs.

Table of Contents

List of Tables

List of Figures

Chapter 1

Introduction

1.1 Background

According to the Alzheimer's Association, the global voluntary health organization in Alzheimer's care and support, Alzheimer's is a type of dementia that causes problems with memory, thinking and behavior. This incurable, degenerative and terminal disease was first described by German psychiatrist and neuro-pathologist Alois Alzheimer in 1906 and was named after him [15]. People with Alzheimer's usually get worse in their cognitive behavior over time and the deterioration often gets severe enough to affect their daily tasks. Alzheimer's is the sixth leading cause of death in the United States. The average living time for patients with Alzheimer's is only eight years. However, AD is not always a disease of old age; many people younger than age 65 can also develop the disease. It is irreversible, and progressively destroys memory cells which results in decline of memory and mental function. Although current Alzheimer's treatments cannot stop Alzheimer's from progressing, they can temporarily slow the worsening of dementia symptoms and improve quality of life for those with Alzheimer's and their care givers [1].

Scientists estimate that up to 4 million people now have AD in the USA alone and for every 5-year age group beyond 65 the percentage doubles. According to the Alzheimer's Association there are 500,000 Americans younger than 65 with Alzheimer's and other dementias, of which 40 percent are estimated to have Alzheimer's [17]. If no preventive measures or treatments are taken, it is estimated that 14 million older Americans will have Alzheimer's disease in 50 years [18]. AD and other forms of dementia have huge impact on the US economy.

As a result, early detection of Alzheimer's is of great importance to patients and medical group. The pre-stage of Alzheimer's is called as MCI, which stands for mild cognitive impairment. People with MCI typically have problems with memory, language, and/or another mental function severe enough to be noticeable to other people, but not serious enough to interfere with daily life. Research has shown that individuals with MCI have an increased risk of developing Alzheimer's over the next few years, especially when their main problem is memory [1].

The data used in this study is obtained from ADNI database, which is an abbreviation for Alzheimer's Disease Neuro-imaging Initiative. ADNI was launched by the National Institute on Aging (NIA), the National Institute of Biomedical Imaging and Bioengineering (NIBIB), and the food and Drug Administration (FDA) in 2004. It was renewed in 2010 and started in 2011 with the name "ADNI GO". Its goal is to determine the characteristics of AD as the pathology evolves from normal aging to mild symptoms, to MCI, to dementia. It uses neuro-imaging and biomarker measures to track the changes in the brains of the study subjects to diagnose AD at an initial stage [1]. ADNI recruited 800 subjects and 200 were cognitively healthy, 400 with MCI and 200 with early AD. The subjects are being followed for about two to three years. ADNI collects the data of the subjects during the study and maintains a database which keeps track of measurements, tests and assessment of the subject. During this time they track the changes of the subjects' brain structure, function, and activities that might help in understanding the progression of MCI to AD.

Machine learning techniques have been extensively applied to clinical and research study. A machine learning model is built based upon a set of training instances. Certain rules or mathematical expressions can be formed during the training phase. To predict, an unlabeled instance is fed to the machine learning model and a class label will be assigned to it according to the classification rules. During the training process, the input is a set of labeled instances and then through the machine learning techniques a set of classification rules are produced from the input data.

The objective of this study is to use machine learning techniques for predicting MCI's future conversion. We use MRI features at month 6 of subjects with MCI to predict their future health conditions (MCI or AD) at month 12, month 24 and month 36. We find that Support Vector Machines and Logistic model give very good results in predicting MCI subjects' future health condition. In particular, by 10-fold cross-validation, more than 97% precision are achieved for both sensitivity and specificity for subjects' conversion types at month 12. Based on the information at month 6, to predict the health condition at month 12 is the most meaningful and indicative thing for medical treatment purposes.

We also use 2-fold cross-validation to estimate the prediction error. By using L1 and L2-norm regularization terms, out model can predict MCI's conversion with a sensitivity /specificity of 82.8%/93.7% under 2-fold cross-validation.

Besides that, as a thorough investigation, we also include clinical features to see if that can boost the prediction accuracy. The results indicate that the combinations of MRI and clinical features doesn't give us a big edge in prediction.

The rest of the thesis is organized as follows. Chapter 2 describes the machine learning methods and concepts used in this study. Chapter 3 gives details of how experiments are conducted and results. Chapter 4 summarizes the project, concludes the study, and suggests future research direction on this project.

Chapter 2

Methods

2.1 Machine Learning Concepts

2.1.1 Binary Classification

Our problem is a typical binary classification problem with labels 0 and 1. It can be expressed in the following general form:

1. Given N samples: $(x_1, y_1), (x_2, y_2) \cdots (x_N, y_N)$, where $x_i \in R^d$, and $y_i \in 0, 1$ is the class label, for $i = 1, 2, \cdots, N$.

2. Our task is to find a classifier $f(x, \theta)$, which represents our prediction of class label for x_i. Here θ is an unknown parameter set to be determined in the fitting process.

3. The performance of the classifier is measured by using the misclassification error defined as the following:
$$E(y, f(x, \theta)) = \begin{cases} 0 & \text{if y=f(x,}\theta\text{)} \\ 1 & \text{o.w.} \end{cases}$$

4. So, the error rate for the dataset of x_i, where $i = 1, 2, \cdots, N$, is: $\frac{1}{N} \sum_1^N E(y_i, f(x_i, \theta))$

5. We need to find appropriate $f(x, \theta)$, i.e. the right set of θ and the right form of f, such that the above error rate is minimized.

2.1.2 Sensitivity and Specificity

For binary classification, accuracy, sensitivity and specificity are always used to measure the model's predictive power. They are defined as follows:

Accuracy: the fraction of correctly classified instances;

Sensitivity: the fraction of the real positives that are correctly classified as positives;

Specificity: the fraction of the real negatives that are correctly classified as negatives;

$$\begin{aligned} accuracy &= \frac{TN + TP}{TN + TP + FN + FP} \\ sesitivity &= \frac{TP}{TP + FN} \\ specificity &= \frac{TN}{TN + FP} \end{aligned} \tag{2.1}$$

where TN stands for true negatives; FN for false negatives; FP for false positives; and TP for true positives.

In medical diagnosis, to correctly classify one who has a disease as such is more important than to classify one who is normal as normal. So sensitivity is more important than specificity in this kind of study. So in our research, we focus more on achieving a good sensitivity.

2.1.3 Cross-validation

Cross-validation is a way of measuring the prediction accuracy of models. For K-fold cross-validation, it partitions one dataset into K parts of equal size (or roughly equal), with the first K-1 parts to fit the model, and then calculate the prediction error of the fitted model by using the left kth part of the data. To reduce the variability, multiple rounds of cross-validation are performed using different partitions, and the validation results are averaged over the rounds. That is, one do this for $k = 1, 2, ..., K$ and combine the K estimates of prediction error [11]. Cross-validation can be used to estimate any quantitative measure of fit that is appropriate for the data and model. In our study, to predict MCI's conversion

type is a binary classification problem, the model either predicts correctly or incorrectly. So the overall accuracy, sensitivity and specificity rate can be used to summarize the fit. Cross-validation can be used to compare the performances of different models. In our study, we use cross-validation results to compare different models.

2.1.4 Loss Functions

In many machine learning tasks, given observations X, we seek a function $f(X)$ to predict outcomes Y. Then a Loss function $L(Y, f(X))$ is required to penalize prediction errors. There are various kinds of Loss functions. For the binary classification purpose, let's assume the response is $y = \pm 1$; the class prediction is $sign(f)$, where f is the prediction by the model. We may have the following losses [9]:
exponential: $e^{-y \cdot f}$; binomial deviance: $log(1+e^{-2y \cdot f})$; squared error: $(y-f)^2$; and SVM hinge: $(1 - y \cdot f)_+$
In our study, we use SVM, Logistic regression and Lasso as predictive models. They use SVM hinge, binomial deviance and squared error Loss function respectively. For classification purposes, the squared-error Loss function is not as robust as the other ones. This is because when $y \cdot f \geqslant 1$, it increases quadratically. In other words, for those observations that are (correctly) classified with great confidence, instead of assigning a loss of 0, it penalizes them even more heavily. So it is not a good choice of Loss function with robustness as a concern.

2.1.5 Feature Selection

Feature selection is the technique for selecting a subset of relevant features for building learning models. Since in many cases we are able to obtain the dataset with a large set of features, we have to face the problem of selecting most informative and predictive features from them. By removing those irrelevant and redundant attributes from the original data, feature selection helps build up a good predictive model in the following aspects:

1. Avoiding the curse of dimensionality.

2. Facilitating the computation procedure.

3. Improving the model interpretability.

The curse of dimensionality refers to various issues that arise when a high-dimensional dataset is analyzed. For example, in sampling, the volume increases exponentially with dimension such that the need of instances increases extremely fast compared to increase of attributes; in optimization, one has to search in a high-dimensional space and the objective function is fed with a larger set of parameters. This brings great obstacle to the computation process.

There are various ways of doing feature selection. Supervised and Unsupervised feature selection methods are used in practice. The former make use of class labels and try to tell the correlation between the class and the feature. The larger the correlation is, the more relevant to the class the feature is. The correlation can be defined in many different ways. Unsupervised feature selection is based on the features internal structure and properties, and then identifying the most promising ones. In our study, we use SVM-RFE, a supervised feature selection method, to eliminate redundant features. Details of SVM-RFE is provided in Section 2.2.4.

2.1.6 Clustering

Clustering is not a specific algorithm, but a task to be resolved. It assigns a group of subjects into clusters based on their similarities. The ones with greater similarities are usually assigned into one group and the ones with less similarities are scattered into different groups. There are various ways of clustering. It can be achieved by various algorithms that differ greatly by the definition of similarity, the notion of what define a cluster and the way how to find them.

2.2 Machine Learning Methods

2.2.1 Support Vector Machine

A support vector machine is a powerful mathematical tool that performs binary classification by constructing a hyper-plane that optimally separates the data into two categories [3]. A good separation is achieved by the hyper-plane that has the largest distance to the neighboring data points of both classes. The data points closest to it are known as support vectors. The object of SVM is to find an optimal hyper-plane and optimize it so that all the data should be easily separated and classified correctly. The number of features in the training data does not affect the complexity of the SVM and therefore SVMs are well suited for the datasets that have a large number of features. When the data are not linearly separable, they are often mapped onto a higher dimensional space and there a hyper-plane is defined. The SVM uses a kernel function to map the data into a transformed feature space where they can be linearly separable. Mathematically, this problem can be conveniently expressed as:

$$\begin{aligned} &Find\ the\ function\ \ h(x) = w^T x + b\ that \\ &\qquad\qquad minimizes\ \frac{1}{2}\|w\|^2 \\ &subject\ to\ y_i(x_i^T\omega + b) \geq 1, i = 1, ..., N, \end{aligned} \tag{2.2}$$

As shown in the following graph, the goal is to find the optimal separating hyper-plane $h(x) = 0$ to maximize the margin. Here the margin is defined as $\frac{2}{\|w\|}$.

In the case that the classes overlap in feature space, one can "soften" the margins by introducing "slack" variables in the constraints while still searching a hyper-plane in the

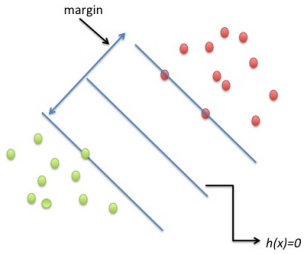

Figure 2.1: Support vector machine (linearly separable)

original space [3]. So the new problem can be formulated as:

$$\begin{aligned} &Find\ the\ function\ \ h(x) = w^T x + b\ and\ \xi\ \ that \\ &\qquad minimizes\ \frac{1}{2}\|w\|^2 + C\sum_{i=1}^{N}\xi_i \\ &subject\ to\ y_i(x_i^T\omega + b) \geq 1 - \xi_i\ and\ \xi_i \geq 0, i = 1, ..., N, \end{aligned} \tag{2.3}$$

The following graph shows us, in the non-separable (overlap) case, some instances come across the border and even appear in the other class domain.

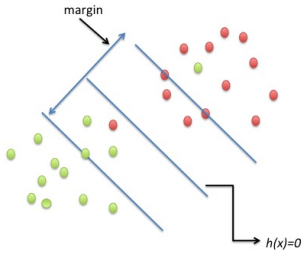

Figure 2.2: Support vector machine (with classes overlap)

SVM can be made more flexible by enlarging the feature space to achieve better training-class separation. Typically, one can consider using some functions to map the original feature space to an enlarged space, and then find linear boundaries in the enlarged space. We list a few kernel functions we use in our experiments here.

We use Chih-Jen Lin's LIBSVM [4] and LIBLINEAR tools [5] for numerical computations. Specifically, LIBSVM is an integrated software for support vector classification, regression and distribution estimation. It also supports multi-class classification. LIBLINEAR is also an open source library. It is for large-scale linear classification and supports logistic regression and linear support vector machines.

LIBSVM supports different kernels which include:

1. linear: $K(x_i, x_j) = x_i^T x_j$.

2. polynomial: $K(x_i, x_j) = (\gamma x_i^T x_j + r)^d, \gamma > 0$.

3. radial basis function (RBF): $K(x_i, x_j) = exp(-\gamma\|x_i - x_j\|^2), \gamma > 0$.

4. sigmoid: $K(x_i, x_j) = tanh(\gamma x_i^T x_j + r)$.

LIBLINEAR supports different combinations of loss energy and regularization terms. Here we only list a few of them. They are actually also used in our study and the one with best prediction accuracy is shown in Chapter 3.

1. L2-regularized L1-loss SVC:

$$\min_w \frac{1}{2} w^T w + C \sum_{i=1}^{N} (max(0, 1 - y_i w^T x_i)) \tag{2.4}$$

2. L2-regularized L2-loss SVC:

$$\min_w \frac{1}{2} w^T w + C \sum_{i=1}^{N} (max(0, 1 - y_i w^T x_i))^2 \tag{2.5}$$

3. L1-regularized L2-loss SVC:

   L1 regularization generates a sparse solution w.

$$\min_w \|w\|_1 + C \sum_{i=1}^{N} (max(0, 1 - y_i w^T x_i))^2 \tag{2.6}$$

   where $\| \ . \ \|_1$ denotes the 1-norm.

2.2.2 Logistic Model

Logistic regression is used for prediction of the probability of an event by fitting a logistic function to the data. It is used extensively in the medical and social sciences, marketing applications. The model has the form:

$$\begin{aligned}
\log \frac{Pr(G=1|X=x)}{Pr(G=K|X=x)} &= \beta_{10} + \beta_1^T x \\
\log \frac{Pr(G=2|X=x)}{Pr(G=K|X=x)} &= \beta_{20} + \beta_2^T x \\
&\cdot \\
&\cdot \\
&\cdot \\
\log \frac{Pr(G=K-1|X=x)}{Pr(G=K|X=x)} &= \beta_{(K-1)0} + \beta_{K-1}^T x
\end{aligned} \tag{2.7}$$

Then the probability of assigning each class value to one individual is:

$$\begin{aligned}
Pr(G=k|X=x) &= \frac{exp(\beta_{k0} + \beta_k^T x)}{(1 + \sum_{l=1}^{K-1} exp(\beta_{l0} + \beta_l^T x))}, \\
&\qquad k = 1, ..., K-1 \\
Pr(G=K|X=x) &= \frac{1}{(1 + \sum_{l=1}^{K-1} exp(\beta_{l0} + \beta_l^T x))}, \\
&\qquad k = 1, ..., K-1
\end{aligned} \tag{2.8}$$

In our study K=2 because we only have MCI and AD, two possible outcomes of MCI's conversion. Logistic regression has good interpretability, and widely used in binary classification applications such as disease diagnosis/prediction.

The following graph is a typical plot of logistic function.

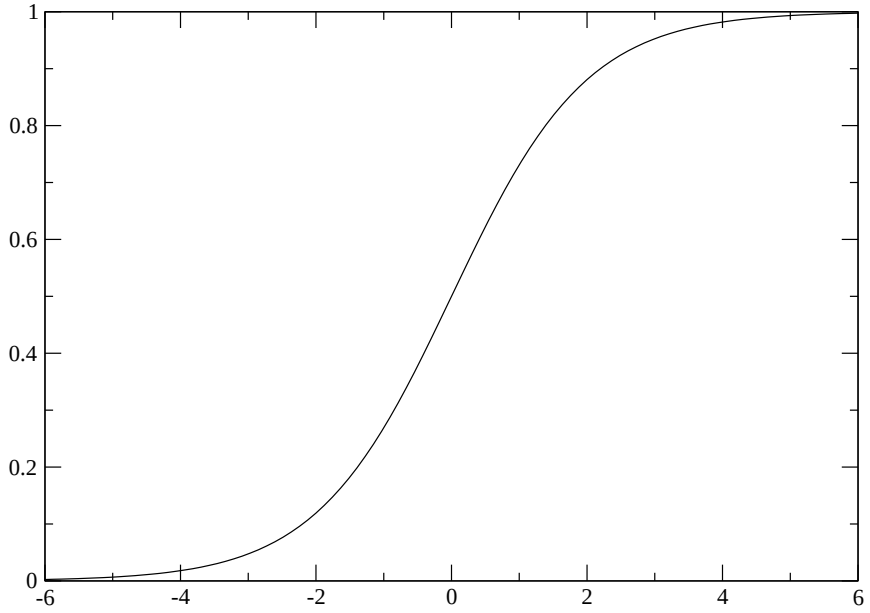

Figure 2.3: Logistic function

From the graph we know that the logistic function helps suppress the effects from too large or too small areas, and guarantees the major effect won't be ignored.

2.2.3 Lasso

Lasso is a linear regression model with a L1-norm regularization term [6]. With this regularization term as a constraint, it shrinks some coefficients down to 0. In this way, the most informative features are retained and those redundant ones are eliminated. Let $\hat{\beta} = (\hat{\beta}_1, \ldots, \hat{\beta}_p)^T$, the goal is to find $(\hat{\alpha}, \hat{\beta})$ such that:

$$(\hat{\alpha}, \hat{\beta}) = \arg\min \sum_{1}^{N} (y_i - \alpha - \sum_{j} \beta_j x_{ij})^2 \quad subject\ to\ \sum_{j} |\hat{\beta}_j| \leq t. \tag{2.9}$$

To express it in a penalization-regularization form, we have:

$$(\hat{\alpha}, \hat{\beta}) = \arg\min \sum_{1}^{N} (y_i - \alpha - \sum_{j} \beta_j x_{ij})^2 + \lambda \sum_{j} |\hat{\beta}_j|. \tag{2.10}$$

Regularization allows complex models to be trained on data sets of limited size without severe over-fitting, essentially by limiting the effective model complexity [9].

2.2.4 SVM-RFE

We have 776 MRI attributes in the very beginning. To select attributes to be used in our models, we use SVM-RFE [10] to pick up 100 features from them. The SVM-RFE algorithm returns a ranking of the features of a classification problem by training SVM with a linear kernel and removing the feature with smallest ranking criterion. This criterion is the weight value assigned by the SVM of the decision hyper-plane.

In particular, the iterative procedure is as following [10]:

1. Train the classifier.

2. Compute the ranking criterion for all features.

3. Remove the feature with smallest ranking criterion.

In essence, it is an instance of backward feature elimination.

Chapter 3

Experiments and Results

3.1 Introduction

We use MRI neuro-imaging data provided by ADNI [2]. Magnetic resonance imaging (MRI), is a medical imaging technique used to visualize the detailed internal structures. Specifically, an MRI machine takes use of magnetic field to systematically control the alignment of some atoms in the body [7]. Nuclei rotate at different speeds due to strong magnetic field gradients. By processing gradients in each direction, 3-D spatial image can be obtained. MRI has been applied to various fields. For example, MRI has widely been used to distinguish pathologic tissue from normal tissue.

ADNI uses neuro-imaging and biomarker measures to track the changes in the brains of the subjects to diagnose AD at an early stage. ADNI recruits people who are healthy but with a memory problem or diagnosed with late-stage MCI or with early Alzheimer's. Among the ADNI 800 subjects, 200 are cognitively healthy, 400 are diagnosed with MCI and 200 with early AD. They have been followed for about 2-3 years. Specifically, AD subjects have their clinical information and neuro-imaging data updated at 6, 12, 24 months. MCI subjects are medically examined at 0, 6, 12, 18, 24, 36 months because of their high risk of conversion to AD. ADNI collects data and maintains a database to keep track of measurements, tests and assessment of the subjects, in the hope of understanding the progression of MCI to AD.

From ADNI's database, we obtain MRI neuro-imaging information from UCSFFSL.csv, UCSFFSX.csv [14], and UASPMVBM.csv [8]. In order to study the conversion of MCI subjects, only those people at month 6 who are diagnosed as MCI are retained, then their healthy status (MCI/AD) at month 12/24/36 are termed as 0 or 1 as the class label. So we

end up with 3 files which aim to predict subjects' health status on month 12, month 24, and month 36 based on their MRI attributes on month 6. For simplification, we call them mri0612, mri0624, and mri0636. Because not all people go through every clinical/MRI check, the number of instances varies at different time. Specifically, in mri0612 dataset, we have 274 subjects and 239 of them at month12 stay at MCI but 35 of them convert to AD; in mri0624 dataset, we have 230 subjects in total and the number of MCI/AD conversions is 149/81; for mri0636, the total is 173 and MCI/AD conversions is 103/70. The total number of MRI neuro-imaging features is 776 and after the preprocessing step by SVM-RFE, we keep 100 of them as the predictors. These 100 attributes are different for the three datasets.

In our experiments, we use 10-fold and 2-fold cross-validation to estimate how good our model will be generalized to other datasets. 10-fold cross-validation is a widely used way of estimating the prediction errors. We also use 2-fold cross-validation because it is frequently used in the medical research field and accepted as a standard way of estimating prediction errors. So in our experiments we list results of both to see how the models behave under these two different cross-validations and to have a better view of the models' prediction error.

In addition to that, we repeat the same experiments on datasets with both MRI and clinical features. Clinical features are basically extracted from the tables as listed in Table 3.1.

Table 3.1: List of Table names in ADNI database

Number	Table Name
1	Cognitive Behavior
2	ADAS Sub-Scores and Total Scores
3	Diagnosis and Symptoms Checklist
4	Functional Assessment Questionnaire
5	Homocysteine Results
6	Vital Signs
7	Mini Mental State Exam

Clinical datasets are important information for medical doctors to access a subject's health condition (NORMAL/MCI/AD). So here we explain the meaning of some clinical tables. Our explanations are based on ADNI's general procedures manuals [15].

ADAS Sub-Scores and Total Scores: it is a brief cognitive test battery that assesses learning and memory, language ability, orientation, etc. Subjects have word recall test first and have the word recognition task last in order to avoid confusing those words from these two tasks. Once the objective testing finishes, subjective clinical testing of language ability and memory are performed. There's no time pressure in this test because the score doesn't depend upon how quickly it can be finished.

Mini Mental State Exam includes various forms of tests. For example, in Delayed Recall test, the examiner asks the subject if they can tell the objects that they are asked to remember three minutes ago; in Attention test, the subjects are asked to spell a word from end to the beginning ; in Command test, one gives a paper to the subject ask him/her to take the paper in the right hand and fold it in half, then leave it on the floor; in the Reading test, subjects are supposed to read and do something as what they read about; in the Writing test, subjects are asked to write a sentence about the weather or about some other topic.

Functional Assessment Questionnaire: it measures activities of daily living and is administered at baseline and at every subsequent visit.

Vital signs: health index are taken in this test. For example, blood pressure and pulse are measured.

The goal of those testings is to use standardized procedures to objectively and reliably assess a subject's cognitive abilities [15]. The following are some Group Specific Inclusion Criteria:

Normal Controls:

1. No memory Complaints aside rom those common to other normal subjects of that age range.

2. Mini-Mental State Exam score between 24 and 30 inclusive (Exceptions may be made for subjects with less than 8 years of education at the discretion of the project director).
3. Clinical Dementia Rating = 0
4. Cognitively normal, based on an absence of significant impairment in cognitive functions or activities of daily living.

Mild Cognitive Impairment (MCI):

1. Memory complaint by subject or study partner that is verified by a study partner.
2. Mini-Mental State Exam score between 24 and 30 inclusive (Exceptions may be made for subjects with less than 8 years of education at the discretion of the project director).
3. Clinical Dementia Rating = 0
4. General cognition and functional performance sufficiently preserved such that a diagnosis of Alzheimer's disease cannot e made by the site physician at the time of the screening visit.

Alzheimer's Disease (AD):

1. Memory complaint by subject or study partner that is verified by a study partner.
2. MMSE between 20 and 26 inclusive (Exceptions may be made for subjects with less than 8 years of education at the discretion of the protocol PI).
3. Clinical Dementia Rating = 0.5, 1.0

3.2 Results

We use LIBSVM, LIBLINEAR and logistic regression to classify our dataset by 10-fold and 2-fold cross-validation. As an application of machine learning techniques in a medical problem, we learn more towards sensitivity instead of specificity. They are both measures of

performance of a binary classification test. Sensitivity (recall rate) measures the proportion of actual positives which are correctly identified as such (e.g. the percentage of AD's who are correctly identified). Specificity measures the proportion of negatives correctly identified (e.g. the percentage of MCI people who are correctly identified). Therefore, also due to the minority of AD's in our datasets, we increase AD's class weight accordingly.

We test different kernels for LIBSVM and as expected, for such a large dimensional space (100-dimensional), linear kernel gives the best fitting. So we only list the results for linear kernel. In LIBLINEAR, we test different combination of losses and regularizations as listed in Eqn. 2.6-2.8., and, the L1-loss (SVM hinge Loss) SVM always performs the best. To increase the sensitivity, we assign a larger weight the AD class. So the results shown in the following tables are the ones with an emphasis on achieving good sensitivity. If necessary, the specificity is allowed to be sacrificed a little in order to enlarge the sensitivity.

Table 3.2: 10-fold cross-validation for LIBSVM

DATA	accuracy	sensitivity	specificity
mri0612	0.996	0.971	1
mri0624	0.983	0.988	0.98
mri0636	1	1	1

Table 3.3: 2-fold cross-validation for LIBSVM

DATA	accuracy	sensitivity	specificity
mri0612	0.682	0.771	0.669
mri0624	0.791	0.852	0.758
mri0636	0.913	0.943	0.893

Table 3.4: 10-fold cross-validation for LIBLINEAR

DATA	accuracy	sensitivity	specificity
mri0612	0.996	0.971	1
mri0624	0.986	0.988	0.987
mri0636	0.913	0.957	0.883

Table 3.5: 2-fold cross-validation for LIBLINEAR

DATA	accuracy	sensitivity	specificity
mri0612	0.668	0.829	0.644
mri0624	0.822	0.852	0.805
mri0636	0.815	0.914	0.748

Table 3.6: 10-fold cross-validation for logistic model

DATA	accuracy	sensitivity	specificity
mri0612	0.942	0.829	0.958
mri0624	0.887	0.852	0.906
mri0636	0.896	0.9	0.893

Table 3.7: 2-fold cross-validation for logistic model

DATA	accuracy	sensitivity	specificity
mri0612	0.923	0.743	0.95
mri0624	0.813	0.802	0.819
mri0636	0.786	0.757	0.806

From Table 3.2 through 3.7, it is not surprising to see Logistic regression and SVM both perform well in prediction. As a separating hyper-plane seeking method, SVM shares some qualitative features with logistic regression. From our experiments, we can tell there must exist separating hyper-planes in the feature space of mri0636 for the perfect prediction results in 10-fold cross-validation .

The prediction on month 12 is no better than month 24 and month36. It is a bit counter-intuitive because it should be easier to predict something that happens in the near future. We attribute it to the insufficient number of AD's (35) for mri0612. As a comparison, we have 81 and 70 AD's for mri0624 and mri0636. So the number may not be sufficient for training a stable and effective classifier in our study of MCI's conversions on month 12.

As a comparison between different methods, we include in Tables 3.8 through 3.13 the results from fitting some other models. Both 10-fold and 2-fold cross-validation results are shown. Among these methods, J48, J48graft, BFTree, ADTree and Simple Cart are decision tree methods; PART, OneR, Ridor, DecisionTable and ConjunctiveRule are Rule methods; IBk, KStar are lazy methods; MultilayerPerceptron and RBFNetwork are artificial neural network models; NaiveBayes and ComplementNaiveBayes are Bayesian methods; LMT is a combination of logistic model and tree model; SMO is a support vector machine classifier with a minimization algorithm different from SVM models in LIBSVM and LIBLINEAR; SPegasos can also be categorized as an SVM model; RandomForest is an ensemble method with voting from tree classifiers. In these experiments, we use WEKA [22] to test different models. WEKA is an open source data mining software written in Java, with implementations of various popular machine learning algorithms.

It is worth noting that SMO [23] performs excellent in mri0636 dataset under 2-fold cross-validation. It is better than any other methods we have so far tested. The only drawback is that it doesn't perform as well for mri0612 and mri0624 datasets. SMO is an SVM model with a different minimization algorithm than the ones in LINLINEAR and LINSVM; so it is not surprising to see that it also gives good prediction for our datasets.

Table 3.8: 10-fold cross-validation for mri0612

METHODS	accuracy	sensitivity	specificity
J48	0.77	0.114	0.866
J48graft	0.792	0.086	0.895
BFTree	0.861	0.029	0.983
ADTree	0.796	0.029	0.908
SimpleCart	0.865	0	0.992
PART	0.796	0.229	0.879
OneR	0.836	0	0.958
Ridor	0.869	0	0.996
DecisionTable	0.869	0	0.996
ConjunctiveRule	0.872	0	1
IBk	0.847	0.171	0.946
KStar	0.869	0.114	0.979
MultilayerPerceptron	0.989	0.943	0.996
NaiveBayes	0.803	0.343	0.87
ComplementNaiveBayes	0.774	0.686	0.787
LMT	0.876	0.2	0.975
SMO	0.993	0.943	1
RBFNetwork	0.883	0.457	0.946
SPegasos	0.971	0.8	0.996
RandomForest	0.872	0	1

Table 3.9: 10-fold cross-validation for mri0624

METHODS	accuracy	sensitivity	specificity
J48	0.652	0.506	0.732
J48graft	0.678	0.506	0.772
BFTree	0.643	0.395	0.779
ADTree	0.604	0.42	0.705
SimpleCart	0.643	0.259	0.852
PART	0.635	0.519	0.698
OneR	0.565	0.346	0.685
Ridor	0.678	0.457	0.799
DecisionTable	0.665	0.494	0.758
ConjunctiveRule	0.652	0.074	0.966
IBk	0.6	0.444	0.685
KStar	0.643	0.407	0.772
MultilayerPerceptron	0.983	1	0.973
NaiveBayes	0.678	0.531	0.758
ComplementNaiveBayes	0.691	0.63	0.725
LMT	0.813	0.716	0.866
SMO	0.986	0.988	0.987
RBFNetwork	0.674	0.568	0.732
SPegasos	0.943	0.938	0.946
RandomForest	0.7	0.296	0.919

Table 3.10: 10-fold cross-validation for mri0636

METHODS	accuracy	sensitivity	specificity
J48	0.624	0.629	0.621
J48graft	0.618	0.614	0.621
BFTree	0.601	0.429	0.718
ADTree	0.613	0.571	0.641
SimpleCart	0.624	0.457	0.738
PART	0.653	0.6	0.689
OneR	0.543	0.386	0.65
Ridor	0.647	0.543	0.718
DecisionTable	0.636	0.471	0.748
ConjunctiveRule	0.543	0.357	0.67
IBk	0.653	0.614	0.68
KStar	0.584	0.629	0.553
MultilayerPerceptron	0.983	0.971	0.99
NaiveBayes	0.676	0.6	0.728
ComplementNaiveBayes	0.653	0.614	0.68
LMT	0.809	0.729	0.864
SMO	1	1	1
RBFNetwork	0.711	0.657	0.748
SPegasos	0.884	0.886	0.883
RandomForest	0.688	0.386	0.893

Table 3.11: 2-fold cross-validation for mri0612

METHODS	accuracy	sensitivity	specificity
J48	0.828	0.171	0.925
J48graft	0.843	0.029	0.962
BFTree	0.854	0.029	0.975
ADTree	0.796	0.171	0.796
SimpleCart	0.872	0	1
PART	0.81	0.171	0.904
OneR	0.825	0.143	0.925
Ridor	0.818	0.086	0.925
DecisionTable	0.825	0	0.992
ConjunctiveRule	0.814	0.057	0.925
IBk	0.825	0.114	0.929
KStar	0.858	0.086	0.971
MultilayerPerceptron	0.88	0.4	0.95
NaiveBayes	0.821	0.257	0.904
ComplementNaiveBayes	0.752	0.514	0.787
LMT	0.821	0.086	0.929
SMO	0.898	0.486	0.958
RBFNetwork	0.818	0.086	0.925
SPegasos	0.843	0.2	0.937
RandomForest	0.869	0	0.996

Table 3.12: 2-fold cross-validation for mri0624

METHODS	accuracy	sensitivity	specificity
J48	0.596	0.395	0.705
J48graft	0.591	0.42	0.685
BFTree	0.622	0.198	0.852
ADTree	0.6	0.37	0.725
SimpleCart	0.617	0.21	0.839
PART	0.652	0.506	0.732
OneR	0.643	0.432	0.758
Ridor	0.661	0.385	0.826
DecisionTable	0.657	0.296	0.852
ConjunctiveRule	0.648	0	1
IBk	0.639	0.42	0.758
KStar	0.617	0.309	0.785
MultilayerPerceptron	0.843	0.691	0.926
NaiveBayes	0.661	0.497	0.752
ComplementNaiveBayes	0.674	0.63	0.698
LMT	0.752	0.556	0.859
SMO	0.852	0.741	0.913
RBFNetwork	0.665	0.457	0.779
SPegasos	0.804	0.617	0.906
RandomForest	0.7	0.321	0.906

Table 3.13: 2-fold cross-validation for mri0636

METHODS	accuracy	sensitivity	specificity
J48	0.555	0.5	0.592
J48graft	0.572	0.457	0.65
BFTree	0.595	0.414	0.718
ADTree	0.572	0.471	0.641
SimpleCart	0.618	0.214	0.893
PART	0.59	0.471	0.67
OneR	0.549	0.329	0.699
Ridor	0.618	0.614	0.621
DecisionTable	0.613	0.414	0.748
ConjunctiveRule	0.572	0.643	0.524
IBk	0.595	0.6	0.592
KStar	0.584	0.557	0.602
MultilayerPerceptron	0.902	0.886	0.913
NaiveBayes	0.665	0.6	0.709
ComplementNaiveBayes	0.671	0.671	0.67
LMT	0.717	0.671	0.748
SMO	0.919	0.929	0.913
RBFNetwork	0.63	0.686	0.592
SPegasos	0.85	0.871	0.835
RandomForest	0.796	0.514	0.796

3.3 Feature Selection and Results

We have 100 features in our dataset which is still too large considering our limited instances. So we would like to retain a most predictive subset of features for a better prediction accuracy and for the subset of features to also be useful in medical diagnoses of AD's. On the other hand, from our previous experiments, we know that hyper-plane seeking methods work well for our datasets. So, a method seeking a separating hyper-plane and also shrinks weights of redundant features down to zero — Lasso, is appropriate for our study.

Figure 3.1-3.3 plot accuracy, sensitivity and specificity vs. number of attributes left for three datasets in 10-fold cross-validation by Lasso:

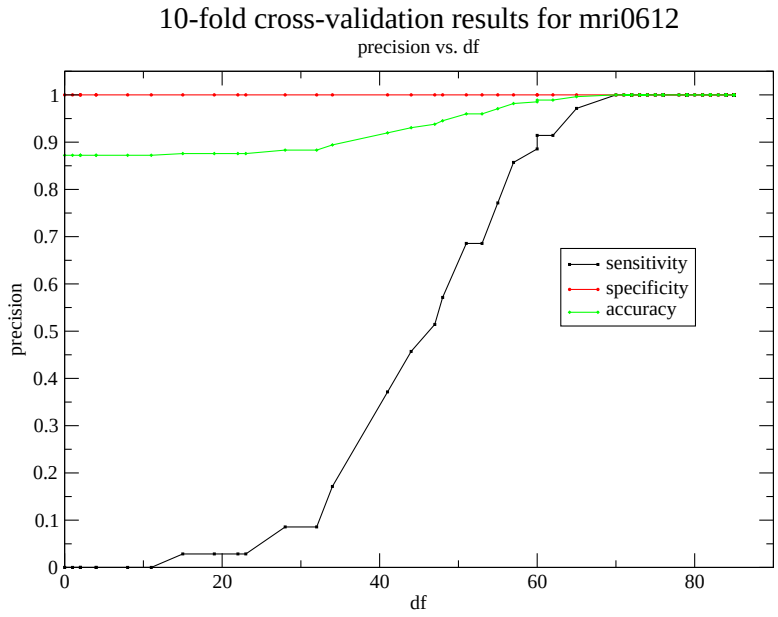

Figure 3.1: We use different values of penalty parameter to adjust the weights of L1 regularization term. So we end up with different degrees of freedom. Sensitivity/Specificity/Accuracy vs. df curves are plotted. Sensitivity of above 0.8 is achieved down to degree of freedom (df) of 57.

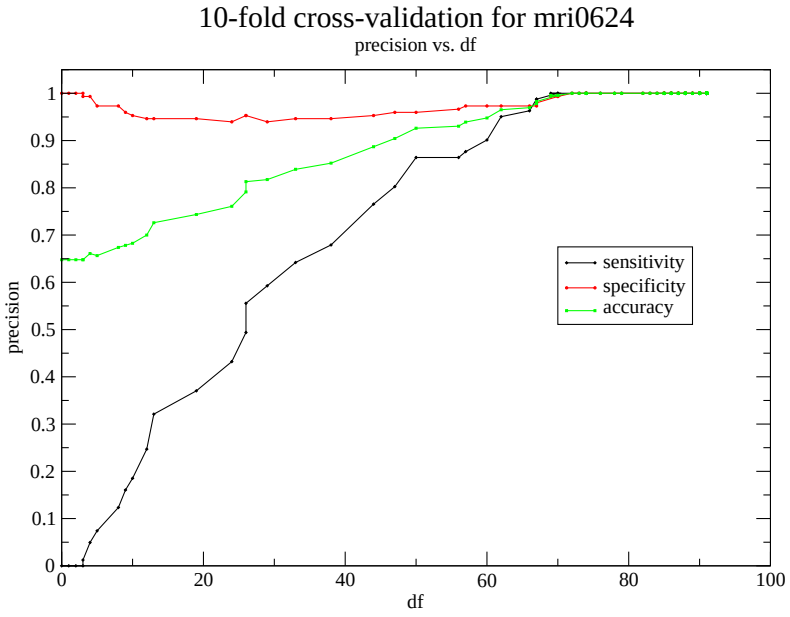

Figure 3.2: Sensitivity of above 0.8 is achieved down to df of 50

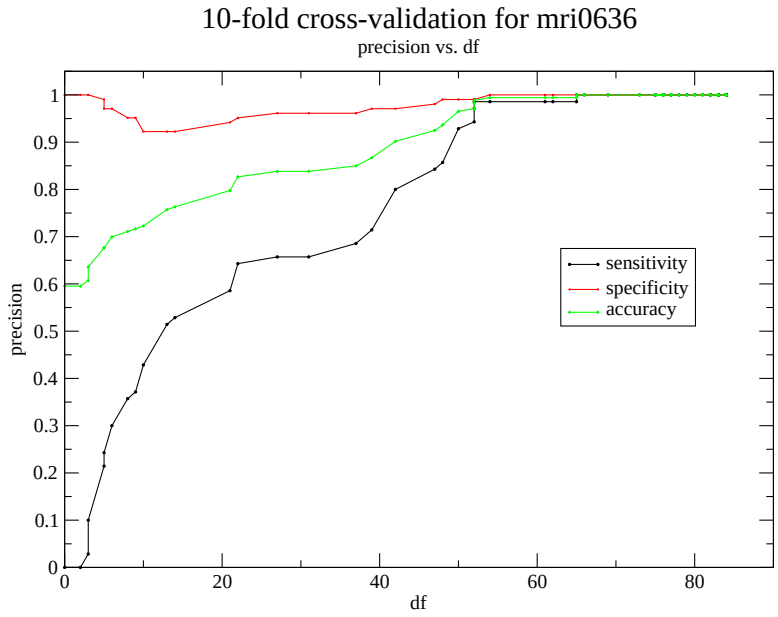

Figure 3.3: Sensitivity of above 0.8 is achieved down to df of 50

We adjust the penalty parameter λ in Eqn. 2.10, to control the weights of the L1-regularization term. By doing so we have different numbers of features left in the model. So in the graph we plot sensitivity/specificity/overall accuracy vs. df (the number of features in the model) to have a better view.

For 2-fold cross-validation, the results show that to eliminate features downgrade the performance of the prediction. So these 100 attributes are retained. The results are as follows:

Table 3.14: 2-fold cross-validation for Lasso

DATA	accuracy	sensitivity	specificity
mri0612	0.865	0.714	0.887
mri0624	0.874	0.889	0.865
mri0636	0.757	0.828	0.747

Although 10-fold cross-validation gives us satisfactory results with about half of the original attributes left, we cannot reach the conclusion that these results will help much for our prediction purposes because the 2-fold cross-validation results we have indicate that throwing

away features severely harms the prediction accuracy. More investigation should be done on this subject to see if the number of features can be safely reduced without harming the predictive ability.

3.4 More on Logistic Model

From the previous experiments and results, we see that 2-fold cross-validation is much more challenging than 10-fold cross-validation. Various models perform very well under 10-fold cross-validation. However, we would like to focus more on 2-fold cross-validation in the following numerical experiments.

The overall results of Lasso with 2-fold cross-validation do not show an obvious improvement compared with other methods. Now we want to see if a change of Loss function brings about any difference. As indicated in Section 2.1.4, squared-error Loss function is not as robust as many other Loss functions. Because it is not a monotone function of $y \cdot f$, where $y = \pm 1$ is the label of a class, and f is the prediction, with class prediction $sign(f)$. When $y \cdot f \geqslant 1$, squared-error Loss function penalizes large $y \cdot f$, which will have negative influence on the classification correctness.

If we use binomial deviance Loss function instead of squared error Loss function, we may be able to obtain a better prediction accuracy. Hence, here we use logistic model again and see how it works with L1/L2 norms for the 2-fold cross-validation.

1. L1-regularized Logistic Regression:

$$\min_w \|w\|_1 + C \sum_{i=1}^{N} (\log(1 + e^{-y_i w^T x_i})) \tag{3.1}$$

2. L2-regularized Logistic Regression:

$$\min_w \|w\|_2 + C \sum_{i=1}^{N} \log(1 + e^{-y_i w^T x_i}) \tag{3.2}$$

We adjust the weight of the penalty term to control the complexity of the model and also the weight of the class 'AD' for good sensitivity. The results we show above are those with both good sensitivity and specificity, with a little bias towards good sensitivity. There are many other good results by using different combinations of weights for the regularization term and the class, and here we only include the representative ones in the Tables 3.15 and 3.16.

We can tell from the results with L1-norm penalty that the original attributes can be reduced down to 80 without harming accuracy much. In Lasso, we cannot reduce the number of attributes due to severe harm to prediction accuracy. More interesting results are from the L2-norm penalty, with this regularization term to control the model's complexity, a boost of both sensitivity and overall accuracy is achieved. The logistic model with L1/L2-norm penalty does give us better prediction performance. We would attribute this to the regularization term, which successfully controls the complexity of the model. On the other

Table 3.15: 2-fold cross-validation for Logistic model with L1 norm penalty

DATA	accuracy	sensitivity	specificity	attributes
mri0612	0.836	0.857	0.711	80
mri0624	0.826	0.815	0.832	80
mri0636	0.803	0.80	0.801	97

Table 3.16: 2-fold cross-validation for Logistic model with L2 norm penalty

DATA	accuracy	sensitivity	specificity
mri0612	0.923	0.828	0.937
mri0624	0.852	0.913	0.819
mri0636	0.826	0.928	0.757

hand, it outperforms Lasso, which also has the L1-norm penalty term, due to the robustness of binomial deviance Loss function used in the fitting.

3.5 Results of MRI combined with clinical features

As a thorough investigation, we also include attributes from clinical dataset and then repeat the same procedure as we do for pure MRI features datasets. Here we give results from both 10-fold and 2-fold cross validations for mri0612 and mri0624 with clinical features (For mri0636, there are not so many corresponding clinical features from the database, so we simply skip it here). From the results in Tables 3.17-3.23, we can tell that we get comparable performance in different models with and without clinical features.

Table 3.17: 10-fold cross-validation for LIBSVM model with clinical attributes

DATA	accuracy	sensitivity	specificity
mri0612	1	1	1
mri0624	0.975	0.928	0.995

Table 3.18: 2-fold cross-validation for LIBSVM model with clinical attributes

DATA	accuracy	sensitivity	specificity
mri0612	0.958	0.758	0.987
mri0624	0.915	0.77	0.975

Table 3.19: 10-fold cross-validation for LIBLINEAR model with clinical attributes

DATA	accuracy	sensitivity	specificity
mri0612	0.996	0.97	1
mri0624	0.979	0.94	0.995

Table 3.20: 2-fold cross-validation for LIBLINEAR model with clinical attributes

DATA	accuracy	sensitivity	specificity
mri0612	0.965	0.788	0.991
mri0624	0.911	0.759	0.975

Table 3.21: 10-fold cross-validation for Logistic model with clinical attributes

DATA	accuracy	sensitivity	specificity
mri0612	0.95	0.879	0.96
mri0624	0.89	0.867	0.899

Table 3.22: 2-fold cross-validation for Logistic model with clinical attributes

DATA	accuracy	sensitivity	specificity
mri0612	0.903	0.606	0.947
mri0624	0.904	0.759	0.965

Table 3.23: 2-fold cross-validation for L2-norm Logistic model with clinical attributes

DATA	accuracy	sensitivity	specificity
mri0612	0.969	0.788	0.996
mri0624	0.911	0.783	0.965

3.6 Experiments with Clustering techniques

As unsupervised learning, clustering has been widely applied to machine learning, data mining, image analysis, information retrieval, and bioinformatics. In this study we use different clustering techniques to assign MCI subjects to different clusters and by belonging to different clusters it may indicate what conversion type (MCI/AD) one will have. We use several techniques including K-means clustering [19], fuzzy C-means clustering [20], and spectral clustering [21], to cluster our MCI instances. Unfortunately no clustering method gave us informative results which indicate MCI's conversion type successfully. We guess the curse of dimensionality can be one of the reasons. So far no regularization methods can be used in clustering to control a model's complexity. Thus the challenge is to find a best subset of variables and then cluster instances based on these features. Clustering MCI and then predicting their conversion types remain a challenging problem in this study.

Chapter 4

Conclusion

We use different machine learning methods to predict MCI's potential future conversion to AD, based on the current MRI features. To estimate different models' prediction error, we employ 10-fold and 2-fold cross-validation on the dataset. Linear-kernel SVM, Logistic regression and Lasso give us very good results for 10-fold cross-validation. We think there exist separating hyper-planes in the feature space of dataset mci0636, and that is why these separating hyper-plane seeking methods all perform well for prediction purposes. For 2-fold cross-validation, the sensitivity and specificity are not as good as in 10-fold cross-validation. When we use L1/L2-norm regularization to control the complexity of Logistic model, a better result for 2-fold cross-validation is obtained.

By using Lasso we try to eliminate irrelevant MRI features, and it works well for 10-fold cross-validation. The number of features can be reduced down to around 50 without increasing prediction error significantly. However, for 2-fold cross-validation, any reduction of the number of MRI features leads to an obvious drop of prediction accuracy. So we would not confidently conclude that the 100 features picked up from original 776 features by SVM-RFE, can be further decreased to 50 features for prediction purposes.

A review of Loss functions indicates that the squared-error Loss function used in Lasso, for classification, is actually not as good as many other Loss functions such as the binomial deviance Loss function. Therefore we come back to Logistic model again for the binomial deviance Loss function it uses. We use L1-norm penalization term for Logistic model in the hope of keeping the most informative MRI features. The use of L2-norm regularization also controls the complexity of the model to avoid over-fitting. We only focus on the results

from 2-fold cross-validation because it is obviously much more difficult than 10-fold cross-validation from our previous experiments. From the results, we can tell that the features can be safely reduced to 80 for mri0612 and mri0624. Also, we gain better performance by using L2-norm for Logistic model than previous models.

So with these results, we can say that the machine learning methods we use accurately and robustly (with 10-fold and 2-fold cross-validation) predict MCI's conversion types. Also for a short term prediction, say, for month 12 and month 24, the feature numbers can be reduced to 80 without increasing the prediction error by much. In addition to that, we notice that L1/L2-norm regularization plays an important role in controlling the complexity of models to avoid over-fitting, and also helps us achieve lower prediction error. The squared-error Loss function in classification is criticized and an advantage of binomial deviance Loss function over the former can be shown from our experiments.

There are many things we can do to further this research project. First of all, we may change the way of selecting features from the 776 features from the very beginning. We use SVM-RFE to rank the features and keep the top 100 of them. However, an unsupervised feature selection or feature extraction method may be better methodologically. It will be more challenging to select attributes unsupervised. Secondly, boosting methods are very popular over the recent 20 years [9]. We should not restrict our methods only to conventional machine learning methods. The ensemble learning is a very promising direction to work with. Generally ensemble learning consists of two steps: the first step is to develop a population of base learners, and then form a committee by combining them and making composite decisions. An application of boosting methods to MCI's conversion type prediction will be very interesting. Besides, L1/L2 norm penalization can also be embedded into these models to avoid over-fitting. Thirdly, generalized additive models can also be used in our research. They can be used to identify and characterize nonlinear regression effects. For example, the additive logistic regression model replaces each linear term by a more general functional form. The following is for a binary classification instance:

$$\log \frac{Pr(G=1|X=x)}{Pr(G=2|X=x)} = \alpha + f_1(x_1) + \cdots + f_p(x_p) \tag{4.1}$$

f_j's are estimated in a flexible manner, using an algorithm whose basic building block is a scatterplot smoother such natural cubic splines. Of course many other forms of basis functions are allowed here. For example, MARS [12] is a generalization of stepwise linear regression which is well suited for high-dimensional problems.

In short, many aspects of the research methodologies and techniques can be reviewed and reconstructed. It deserves further attention and effort to dig more into the problem.

Bibliography

[1] http://www.alz.org, accessed on Dec. 2, 2011.

[2] http://ani.loni.ucla.edu/about/about-the-study/, accessed on Dec. 2, 2011.

[3] C. Cortes and V. Vapnik: Support-Vector Networks, Machine Learning, 20, (1995): 273-297.

[4] C. Chang and C. Lin, LIBSVM : a library for support vector machines. ACM Transactions on Intelligent Systems and Technology, 2, (2011): 27:1–27:27

[5] R. Fan, K. Chang, C. Hsieh, X. Wang, and C. Lin: LIBLINEAR: A Library for Large Linear Classification, Journal of Machine Learning Research, 9, (2008), 1871-1874.

[6] R. Tibshirani, J. Statist. Soc. B (2011)

[7] L. Squire and R. Novelline: Squire's fundamentals of radiology (5th ed.). Harvard University Press. (1997) ISBN 0-674-83339-2

[8] http://www.fil.ion.ucl.ac.uk/spm/, accessed on Dec. 2, 2011.

[9] T. Hastie, R. Tibshirani and J. Friedman: The Elements of Statistical Learning: Data Mining, Inference, and Prediction. Springer Series in Statistics (2009)

[10] I. Guyon, J. Weston, S. Barnhill and V. Vapnik: Gene Selection for Cancer Classification using Support Vector Machines (2002)

[11] S. Geisser: Predictive Inference (1993), New York: Chapman and Hall. ISBN 0412034719

[12] J. Friedman: Multivariate adaptive regression splines (with discussion), Annals of Statistics, 19, (1991): 1-141

[13] R. Tibshirani: Regression shrinkage and selection via the lasso. J. Royal. Statist. Soc B., Vol. 58, No. 1, (1996): 267-288

[14] http://surfer.nmr.mgh.harvard.edu/, accessed on Dec. 2, 2011.

[15] http://www.adni-info.org/Scientists/ProceduresManuals.aspx, accessed on Dec. 2, 2011.

[16] N. Berchtold, C. Cotman: Evolution in the conceptualization of dementia and Alzheimer's disease: Greco-Roman period to the 1960s, Neurobiol Aging 3 (1998): 173-189.

[17] Alzheimer's disease Facts and Figures, Alzheimer's & Dementia: The Journal of the Alzheimer's Association 6 (2010):158-194.

[18] A. Rodgers: Alzheimer's disease unraveling the mystery, National Institutes of Health NIH (2003)

[19] J. MacQueen: Some Methods for classification and Analysis of Multivariate Observations, Proceedings of 5th Berkeley Symposium on Mathematical Statistics and Probability. University of California Press. (1967) pp. 281-297

[20] R. Nock and F. Nielsen: On Weighting Clustering, IEEE Trans, on Pattern Analysis and Machine Intelligence, 28, (2006): 1-13

[21] S. Yu and J. Shi: Grouping with bias. NIPS 14 (2002)

[22] M. Hall, E. Frank, G. Holmes, B. Pfahringer, P. Reutemann, and I. Witten: The WEKA Data Mining Software: An Update; SIGKDD Explorations, 11 (2009)

[23] J. Platt: Fast Training of Support Vector Machines using Sequential Minimal Optimization. Advances in Kernel Methods, 208, (1998): 1-21