KTH Computer Science
and Communication

Prediction of Mild Cognitive Impairment Progression using Longitudinal MRI Scans

THOMAI STATHOPOULOU

DA221X - Master's Thesis in Computer Science
Supervisor: Kevin Smith, Farshad Falahati
Examiner: Johan Håstad
Contact: Thomai Stathopoulou, thomai@kth.se

Master's programme in Machine Learning
School of Computer Science and Communication
KTH, Royal Institute of Technology, Stockholm
April 17, 2017

Abstract

As human life expectancy has significantly increased in the last decades, the number of elderly people has also increased. As a consequence there are more and more incidents of age related diseases and conditions, such as Alzheimer's Disease (AD), which is the most common form of dementia among elderly people. Mild cognitive impairment (MCI) is an intermediate condition between normal ageing and dementia, that involves noticeable decline in cognitive abilities, such as memory and thinking skills. MCI often represents a prodromal form of dementia, conferring a significantly higher risk of converting to AD.

Magnetic resonance imaging (MRI) is a tomographic imaging technique, that produces virtual images from inside of the body. Brain MRIs provide comprehensive data on brain structures. Advance image processing algorithms and machine learning methods provide opportunities to analyse and find inherent patterns in this complex data. By doing so, it is possible to separate groups, determine which factors cause the separation and make predictive models of disease. MRI has been widely studied in clinical trials for early detection of AD and MCI, mostly using cross-sectional MRI data. Longitudinal studies track the same individuals for several years and therefore provide an opportunity to investigate the disease's progression and the changes of patterns overtime, in addition to cross-sectional patterns.

In this project we propose three methods that attempt to analyse and evaluate the structural information included in longitudinal MRI scans. This information is used for the prediction of the progression of individuals diagnosed with MCI while no other diagnostic metric is taken into consideration. These methods are implemented using Hidden Markov Models as well as taking advantage of the strength of Support Vector Machine classifiers.

We conduct a number of experiments in order to evaluate the performance of our system and check the effect that each method has on the performance and the prediction. At the end of the thesis we discuss our results, as well as, certain issues and improvement possibilities.

Referat

Förutsägelse av Lindrig Kognitiv Störning progression med användningen av Långvariga Magnettomografi

Med en ständigt ökande förväntad medellivslängd verkar Alzheimers sjukdom (Alzheimer's Disease - AD), en av de vanligaste formar av demens hos äldre människor, drabba fler och fler individer. Lindrig kognitiv störning (Mild Cognitive Impairment - MCI) är ett tillstånd mellan normalt åldrande och demens, som innebär märkbar nedgång i kognitiva förmågor, såsom minne och rationellt tänkande. MCI utgör ofta en tidig form av demens och resulterar i en betydligt högre risk att insjuka i Alzheimers.

Magnetisk resonanstomografi (Magnetic Resonance Imaging - MRI) är en tomografisk bildteknik, som skapar virtuella bilder från insidan av kroppen. MRI kan användas för att få detaljerade bilder av hjärnstrukturer. Avancerade bildbehandlingsalgoritmer och maskininlärningsmetoder ger möjlighet att analysera och hitta mönster i denna komplexa data. Med hjälp av dessa metoder är det möjligt att separera grupper, bestämma vilka faktorer orsakar olikheter och skapa prediktiva modeller av sjukdomen. MRT har i stor utsträckning studerats i kliniska studier för tidig diagnos av AD och MCI. Långvariga studier följer samma individ under flera år och ger därmed en möjlighet att undersöka sjukdomens progression och förändringar av mönster över tid.

I detta projekt föreslår vi tre metoder som för att analysera och utvärdera den strukturella informationen som finns i MRI data över en längre tid. Dessa modeller används för att förutsäga utvecklingen av personer som diagnostiserats med MCI utan att beakta andra faktorer. Metoderna implementeras med hjälp av Hidden Markov Models samt support vector machines klassificering.

Efter att ha utfört våra experiment kan vi konstatera att vi modellerna visar tillfredsställande resultat. Resultaten diskuteras tillsammans med vissa frågor och problem som uppstår i det sista kapitlet.

Contents

Abbreviations

AD Alzheimer's Disease

ANN Artificial Neural Network

CN Cognitively Normal

CSF Cerebrospinal Fluid

DOR Diagnostic Odds Ratio

EM Expectation Maximization

HMM Hidden Markov Model

MCI Mild Cognitive Impairment

MRI Magnetic Resonance Imaging

NMR Nuclear Magnetic Resonance

OPLS Orthogonal Projection to Latent Structures

RF Radio frequency

RNN Recurrent Neural Network

SVM Support Vector Machine

Chapter 1

Introduction

This chapter is a general introduction to the subject of the thesis, as well as the objective of the research done throughout the thesis work. At the end of the chapter we provide a general description of the report's structure.

1.1 The Ageing Brain

The brain is an extraordinary organ. It is the organ that manages our entire body and automatically coordinates all the necessary body functions, such as breathing, blood circulation and digestion. Additionally, it enables us to do all the functions we consciously do, such as walking, talking, seeing etc. It is also the organ that enables us to think, create memories, retrieve thoughts and in general make conscious decisions.

Anatomically, the brain is also very interesting. It consists of nerve cells (**neurons**) and other types of cells. It is estimated that it contains about 100 billion neurons and 100 trillion **synapses** (gaps between the neurons, used by the neurons to communicate). The brain's network of blood vessels is also very significant for its proper function. Interestingly, even though the brain is only about 2% of the body weight, it receives 20% of the blood's supply. The blood network is comprised by 400 billion blood vessels (called **capillaries**), that carry oxygen, glucose and other nutrients needed for the brain cells' sustenance [2].

Healthy Ageing

This vast number of neurons in the brain, is responsible for it working properly. A very interesting characteristic is that, as opposed to ordinary cells, they have a very long lifespan. They are created while the human is still a fetus and can live for as long as 100 years. If they die they are extremely rarely regenerated. Therefore it is vital that they are properly maintained and repaired in case of damage.

While the human body ages, it changes. The same happens to the brain as well. Changes in the neurons and neurotransmitters can have a negative effect on the communication between neurons. Shrinking of parts of the brain, thinning of the blood vessels or even inflammation of certain areas if an injury or disease has occurred are some of the physiological changes that appear with ageing.

Depending on the extent of these changes and on the time when they start happening they can affect each individual on different levels. Some of the most common indications of ageing are a decreased ability to learn new things, or to retrieve information from memory, or even higher difficulty in performing certain tasks, which would otherwise feel easier. However, these abilities are not absolutely hindered, that is to say that a cognitively normal (**CN**) elder will still be able to perform these tasks though it may take them longer than usual.

Interestingly, it has been observed that, in many cases the brain activates certain parts in order to perform a task that would otherwise require different parts of the brain. Even though this is not fully explained by scientists, it can be perceived as a kind of compensation mechanism for difficulties that other regions may be experiencing.

In general there are not definite factors that lead to normal or abnormal ageing, but a number of factors are believed to play a role. Factors such as overall health, environment, education (and by extension intellectual stimulation of the brain) and genetics are believed to be affecting the brain's course throughout its life.

Alzheimer's Disease

As discussed earlier, the brain goes through morphological and structural changes while it ages. These changes cause a decline in cognitive and memory functions [2]. For a cognitively normal individual, this decline is not as severe so as to interfere with their daily lives. However in an ever increasing number of cases, this decline is quite severe, causing problems in the daily life of an elder person and in many cases leading to early death. This is the case of **dementia**, which is categorised as a neuro-degenerative disorder.

Alzheimer's disease (**AD**) is the most common form of dementia, accounting for $50 - 80\%$ of dementia cases [34]. The average life expectancy of people suffering from AD is $3 - 10$ years and depends greatly on the age when the disease was diagnosed or first appeared. It is believed to be related to structural atrophy of certain regions of the brain, pathological amyloid depositions and metabolic alterations in the brain. However, scientists are still uncertain if these are the causes of the disease, or are by-products of its development [2, 34].

Mild cognitive impairment (**MCI**), which in many cases is also referred to as amnestic mild cognitive impairment (due to its effect on memory), is a condition which is considered to be an early stage of AD. MCI is not characterized as a disease, but resembles in many ways earlier stages of AD. People that exhibit symptoms of MCI, experience problems with memory, as mentioned earlier, language and even judgement. These problems are not so grave, so as to interfere with their lives and that is why the condition cannot be characterized as AD or dementia. They are however severe enough to be noticeable by others and therefore be separated from healthy ageing symptoms. What emphasizes MCI's significance is the fact that people with this condition exhibit a higher risk of developing AD in the future, compared to a cognitively normal individual. This makes MCI a very interesting condition among the medical community.

The lines between the three stages (CN, MCI, AD) are quite blurry and there are no definite criteria for estimating in which stage each individual is, but over years of research certain ways have been developed to assess the brain's health.

1.2 Brain Studies & MRIs

Studying the brain and by extension its abnormalities has been a very important field in science and medicine. However, for a long time, a drawback was the inability to study it non-invasively. Even today, a definite diagnosis for AD can only be made post-mortem, during the autopsy, when a doctor is able to examine the amyloid plaques that are created in the brain and other indications of brain degeneration.

The need for an early detection and diagnosis of AD and MCI is imperative, since it can help patients and their families to be prepared for the disease and begin treatment as early as possible. This can give them the opportunity to participate in clinical trials, where they can receive the latest treatments and in general cope with the condition more smoothly. At the same time, scientists can study the early stages of AD and MCI in an attempt to understand the disease's aetiology, which can lead to better treatment or prevention methods. It is estimated that patients with MCI progress to AD with a rate of $10 - 15\%$, while cognitively normal individuals progress to any form of dementia with a rate of $1 - 2\%$ [34].

Magnetic Resonance Imaging (**MRI** - Appendix B) has given scientists the opportunity to look into the human body in vivo [25]. This means that it is possible to see and examine the brain, at different stages of the disease's progression, or even examine cognitively normal brains, in order to establish a baseline of how it should look like. With the addition of computer science and machine learning it has been possible to study the brain and the occurring changes either due to simply ageing, or due to a certain illness.

1.3 Related Work

In this section we briefly present ongoing research within the fields of Computer Science and Machine Learning regarding the Alzheimer's Disease and Longitudinal Data.

1.3.1 Brain

A lot of research has been focused on identifying and extracting those features of an MRI scan, that function as the best diagnostic predictors in order to be able to proceed with further diagnosis [4, 5, 8, 34, 37]. The most commonly used characteristics are the volume of gray and white matter [25, 32, 37], either throughout the entire brain or within specific regions, such as the frontal, temporal, parietal and hippocampal cortex[3]. Cortical thickness is also quite commonly used [34], as well as density maps of cerebrospinal fluid (CSF) [25, 32].

The procedure of manually selecting and extracting features from MRI scans is quite difficult and tedious. Quite often it can result in faulty data or it can be complicated to re-extract features, if certain feature parameters change. Hence, a number of tools have been developed, such as FreeSurfer [10], FSL [18] and SPM [11], that enable scientists and medical doctors to handle and process MRI scans in different ways and extract accurate features. These tools can perform the selection & extraction tasks with minimal to no supervision.

1.3.2 Alzheimer's Disease

There has also been great progress in the attempt to handle and use those features. Using different classification methods it is possible to distinguish between a cognitively normal and a diseased brain, both from AD and MCI. Such research most commonly uses physiological and structural characteristics of the brain (retrieved by means of MRIs or f-MRIs) in order to identify any changes, normal or not, that occur in the brain. These are volumetric measurements of several regions of the brain, such as the hippocampus, the cingulate cortex, parahippocampal gyrus etc., as well as the cortical thickness and the density of CSF, as mentioned in the previous Section. These measurements help with the observation of abnormalities detected in the brain which can give invaluable information about the presence of any form of dementia.

In a study focused on classifying AD and MCI [37] they use such biomarker features extracted from MRI and PET scans so that an SVM classifier will be able to distinguish between CN and MCI or CN and AD patients. The systems perform with classification accuracies 76.4% for the MCI classification and 93.2% for the AD classification. In a similar study [34] they also use brain features and biomarkers (ROI-based morphological features, such as cortical thickness values and volumes of the cerebral cortical gray matter and the cortical associated white matter). Here they introduce the concept of correlated abnormalities, where they attempt to produce features and detect abnormality patterns by correlating several ROIs with each other. They also train an SVM classifier for separation of CN and MCI, CN and AD and MCI and AD, achieving classification accuracy 83.75%, 92.35% and 79.24% respectively.

Another method developed is the orthogonal projection to latent structures (OPLS) [27]. This method combines the theory behind partial least squares (PLS) [35] regression with orthogonal signal correction (OSC) [29, 36] that was first developed for modelling complex data. Its basis was the assumption that there are latent variables generating the observations. However the method seems to be negatively affected by systematic variation in the independent variables that was not related to the class labels, therefore the OPLS method is developed, which attempts to overcome this problem. The OPLS classifier achieves 86.1% sensitivity and 90.5% specificity (Sec. 2.5) in discriminating AD from CN subjects.

Other less common approaches include artificial neural networks (ANN) [1, 6], decision trees and other methods for classification or regression, though with results inferior to the methods described above.

These methods use cross-sectional MRI scans and mostly focus on detecting the disease from the scan or predicting it using only information from the brain structure within the current scan.

1.3.3 Longitudinal Data

Studying just cross-sectional MRI scans, even though successful and promising, provides information only for a specific instance of a disease or the brain's structure in general. It is not possible to provide information about changes occurring over time, find patterns and even correlate them with different conditions. This is why longitudinal studies have become of great interest within the research community. That means a sequence of data (e.g. brain MRI scans) taken at constant time intervals (e.g. every 6 months or every year).

Some research that focuses on processing longitudinal data has been done on f-MRI scans [19, 23] focusing primarily on the responses that certain regions exhibit. These studies used mostly regression models (linear, generalized least squares etc.), but cannot be perceived as using longitudinal MRI scans, as we intend to in this thesis, since the f-MRI monitors the brain activity over a period of seconds. The longitudinal data that we intend to use study the brain changes occurring during much longer periods.

The use of Hidden Markov Models (HMMs) was introduced in an attempt to detect early dementia (mild Alzheimer's disease) in elderly people in [3]. In this study, features extracted from a sequence of slices of an MRI scan are formed into a time-series which then, using HMMs, are analysed and classified. The proposed method is quite successful in detecting early dementia, and in certain experiments performs with accuracies as high as 97.8%. However it is obvious that in this research detect AD from a snapshot of the brain and they do not attempt to predict it or study its progression over a period of years, while the data is still not longitudinal. Similarly in a different study [31] we encounter HMMs once again for the prediction of age of non-demented subjects, still without longitudinal data. The average error is as low as 2.57 years.

Finally, in a more recent study the use of longitudinal MRI scans is introduced in order to study the changes and correlation over the years between scans of cognitively normal and demented brains [32]. With the use of 9-year longitudinal MRI scans they are able to study the information extracted by the individuals' scans, creating a new path in brain studies, that offer tremendous potential. Though we are using different features and data-set, this study is very important for us, since it is an indication of the rich information that longitudinal MRI scans can provide.

1.4 Research Question

MCI is a cognitive condition where the individual exhibits signs of cognitive decline that are intense enough so that they are noticeable, but not as intense so that they can be characterized as AD or any other form of dementia. It is considered to be a prodromal stage of AD. Even though it is not a definite diagnosis (having MCI doesn't necessarily mean progression to AD; the condition can convert back to healthy ageing or remain stable), it can serve as a red flag for the doctor and the patient, so that they can both be warned, be alert and start treating the condition.

Like any other cognitive condition, many factors are important concerning MCI or AD, such as the time when the condition first appears, the mental status of the individual, environmental or genetic factors etc. This plethora of factors affects the progression of MCI as well, which could either convert to AD or remain stable or even convert to a state where the individual is considered cognitively normal (similar cognitive abilities are indicators of different conditions, depending on the circumstances). It would therefore be very helpful for a doctor to be able to predict or know to a certain extent whether a patient diagnosed with MCI is prone to progress to a more serious condition or not.

The goal of this thesis is to analyse and study the *structural information* extracted from *longitudinal brain MRI scans*. We introduce the use of longitudinal scans in an attempt to study the gradient of the structural and morphological changes occurring in the brain with relation to the progression of MCI towards AD. By using only this information and no other biomarker or clinical and cognitive measurement we attempt to predict a possible AD development. This is something that has not been attempted before, even though, as mentioned in Section 1.3, research regarding AD has used MRI scans.

We use longitudinal series of MRI scans of different individuals, that are diagnosed with different conditions (CN, MCI, AD) and that progress to different conditions (CN, MCI, AD). By using only the information extracted by the structural (volumetric) changes of the brain, we hope to eliminate the human input from *complicated* and *time-consuming* processes and to focus on the information that an MRI can provide. This way we can eliminate any errors of judgement that can occur and enable an earlier prediction of the disease, since a patient usually consults a doctor after symptoms start occurring and the diagnostic procedure needs a certain amount of time to formulate a diagnosis. Additionally, this thesis constitutes an effort to evaluate and study the strength of the longitudinal MRI within this specific field with regard to its ability to predict MCI to AD conversion.

We use our longitudinal MRI scans as a sequence of observations, which we attempt to model with HMMs (Section 2.2). We then proceed with the prediction either by using only the HMMs or by using the HMM modelling of our data to train an SVM classifier (Section 2.3), which in turn implements the prediction. The different methods and approaches are discussed in detail in Section 2.4.

It should be noted at this point that this thesis does not implement, extend or attempt to improve a specific defined method. The novelty of our approach is the use of longitudinal MRI scans one-year apart combined with the use of merely the structural information extracted from the scans. This means that it is not possible to compare closely the performance and results of our experiments with the state-of-the-art results. What we do instead is compare our results with a random classifier/method, which we use as a baseline of a lower limit (3.2.1).

1.5 Report Structure

The rest of the report is divided into three separate chapters.

In Chapter 2 we initially present the data-set that we use and we give the distribution of the cases, as well as some basic characteristics of the participants. We move on to introducing the theory behind Hidden Markov Models (HMMs) and Support Vector Machines (SVMs), which are the basic features of our methods. These methods are described right after the theoretical background is complete. Finally we provide the formulas and a basic introduction to the metrics, which we use in order to evaluate the implemented systems.

Chapter 3 is dedicated in the experiments that are conducted in order to test the performance of the methods upon the provided dataset. Initially we give practical information and details about how the experiments are designed and executed. We then establish a baseline of the performance that we set as our minimum accepted threshold. The results of the three methods are then presented and commented along with an overview of the entire experiment process.

In the final Chapter we present our conclusions along with the issues we confronted and some suggestions about how to overcome them or even improve our methods in the future.

Chapter 2

Methods

This chapter presents the background theory and the techniques and materials used for the design and completion of the thesis. The proposed system is described, along with all the information needed from the reader in order to understand how the system works and experiments are carried out.

2.1 Participants

The MRI data used in this thesis was obtained by the Alzheimer's disease Neuroimaging Initiative (**ADNI**). ADNI was launched in 2003 and is funded by the National Institute on Ageing (**NIA**), the National Institute of Biomedical Imaging and Bioengineering (**NIBIB**), the Food and Drug Administration (**FDA**), several private pharmaceutical companies and non-profit organizations (Alzheimer's Association, Institute for Study of Ageing) and in collaboration with the NIH Foundation [21].

Its primary goal is to develop and implement methods for the acquisition of longitudinal data on patients with AD and MCI, as well as CN individuals. ADNI uses this data to test whether serial MRI, PET, other biological markers and clinical and neuropsychological assessments can be combined to monitor and measure the progression of MCI and early AD and also provide a generally accessible imaging and clinical data repository, which describes longitudinal changes in brain structure and metabolism, cognitive function, and biomarkers in CN, MCI, and AD patients. ADNI subjects were recruited from over 50 sites across the U.S.A. and Canada [7].

For this thesis we have been provided with longitudinal MRI data of 631 individuals. Of these individuals 192 are diagnosed as CN at the time of their first scan, 309 as MCI and 130 as AD. While at the time of their last scan 189 are diagnosed as CN, 202 as MCI and 240 as AD. Each individual has a different number of follow-up scans, ranging from 1 up-to 3 follow-ups, all with a one-year time window between scans. Figure 2.1 and Tables 2.1 & 2.2 provide a summary of the basic characteristics of the data-set. In total 1913 MRI scans were provided (*1.5T sagittal 3D T1-weighted MPRAGE MRI scans*). Specifically in Figure 2.1 we can see the different "direction" that the individuals take throughout their MRI sequences. On the left side lie the diagnoses at the time of the first scan, while the arrows point to all the possible final diagnoses (the diagnosis at the time of the last scan) that can occur after each first diagnosis. We can therefore divide the subjects in two different types of groups. Groups of the first and last diagnosis. Both groups contain three "labels", i.e. CN, MCI and AD.

We can see that the individuals that are diagnosed with AD from the first scan have no alteration to their condition, there are no conversions to either MCI or CN states. Also, these individuals have upto two follow-up scans (three MRI scans in total per sequence). Within the other two labels, the individuals have from one upto three follow-up scans and

Table 2.1: Subjects' Characteristics

		Baseline Diagnosis		
		CN	MCI	AD
Gender	Male	100	200	67
	Female	92	109	63
Age (μ, σ)	Male	$(75.1, 6.94)$	$(75.7, 6.55)$	$(76.1, 7.34)$
	Female	$(76.1, 6.21)$	$(75.1, 6.48)$	$(76.1, 6.41)$

Table 2.2: Summary of the Data-set

		Baseline Diagnosis		
		CN	MCI	AD
Total		192	309	130
Diagnosis at Last Scan	CN	180	9	0
	MCI	10	192	0
	AD	2	108	130
No. of Follow-Up Scans	1	61	160	33
	2	2	4	97
	3	129	145	0

their diagnoses progress to all three possible conditions.

These MRI scans are preprocessed using the Freesurfer [9] pipeline, which is an open source suite that provides tools for the extensive and automated analysis of key features of the human brain. This includes volumetric segmentation of most macroscopically visible brain structures, segmentation of hippocampal sub-fields, inter-subject alignment based on cortical folding patterns, estimation of architectonic boundaries from in vivo data, mapping of the thickness of cortical gray matter and other functions, which would be very difficult and time-consuming to do manually and for large number of data. As a result, a total of 55 MRI-derived regional measures, including 34 cortical thickness and 21 subcortical volumes, are extracted from each MRI scan. In Appendix C we have a list of the regional features that were used.

2.2 Hidden Markov Models

A **Hidden Markov Model** (**HMM**) is a tool that can model a series of observations that are produced by a real-world process assumed to be a Markov chain, that produces a number of hidden states (latent variables).

Markov Chain Assuming that there is a number of N possible *states*, $S = s_1, s_2, \ldots, s_N$, the process starts from one of these states and can move "forward" from one state to the other, producing a sequence of different states. At every step, the process can move to a new state based on a probability distribution, that involves the current state i and all other states (including itself)[15]. This is called the *transition probability* and is usually

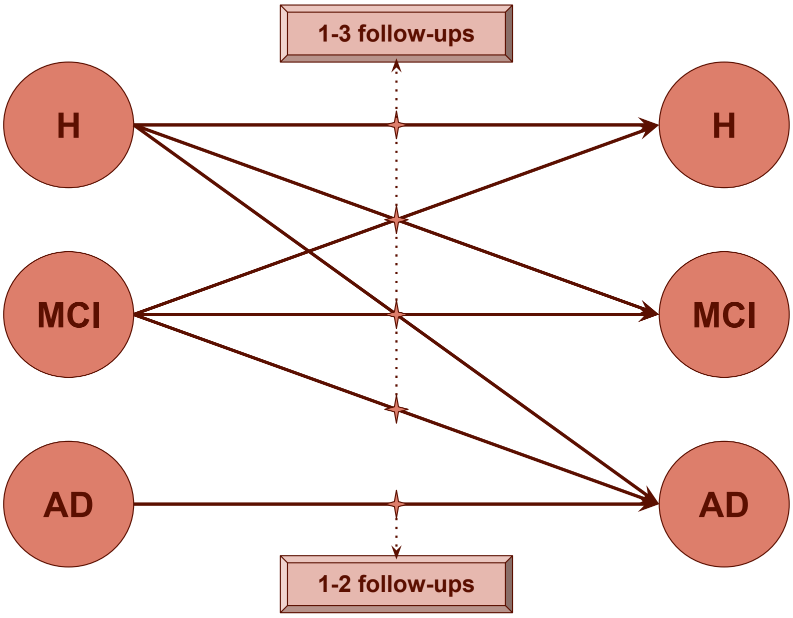


Figure 2.1: The possible diagnoses of the subjects' cross-sectional and final MRI scans

represented by an $N \times N$ matrix A, called a *transition matrix*:

$$A = \begin{bmatrix} a_{11} & a_{12} & \dots & a_{1N} \\ a_{21} & a_{22} & \dots & a_{2N} \\ \vdots & \vdots & \ddots & \vdots \\ a_{N1} & a_{i2} & \dots & a_{NN} \end{bmatrix}$$

Each element a_{ij} of this matrix gives the probability of moving towards state s_j when the current state is s_i. Matrix A is row-stochastic, meaning that each row is a unique probability distribution which needs to add up to 1. It is possible for elements to be zero, which would indicate that a certain transition is not possible (e.g. if $a_{35} = 0$, it is not possible to move to state s_5 from state s_3).

In addition to matrix A, we have a vector, $\pi = [\pi_1, \pi_2, \dots, \pi_N]$. It is a probability distribution for the starting state of the sequence. Using A and π it is possible to produce a sequence of states (potentially infinite), which is called a **Markov chain**. The basic concept behind the Markov chain is that at every step of the chain, the next step is *dependent only on the current state.*

Hidden Markov Models Building upon the Markov chain described in the previous section, it is now assumed that the state sequence that is produced by the process is not observable (it is hidden), but that it is possible to acquire a sequence of observations produced by the states at every step of the process [13]. We therefore assume that we have a number of N possible states, $S = s_1, s_2, \dots, s_N$, a transition matrix A, an initial vector of probabilities π and a number of M possible observations $O = o_1, o_2, \dots, o_M$. When the process, at step t, is at state s_i, it emits an observation based on the $N \times M$

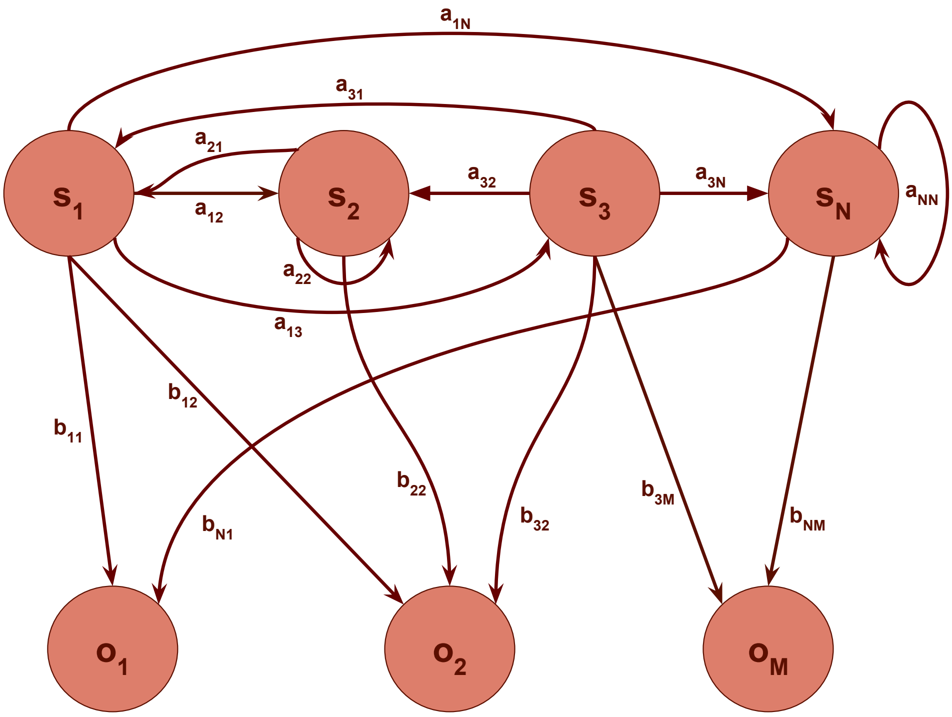


Figure 2.2: Probabilistic Relations among the states and observations of a HMM

emission matrix B:

$$B = \begin{bmatrix} b_{11} & b_{12} & \dots & b_{1M} \\ b_{21} & b_{22} & \dots & b_{2M} \\ \vdots & \vdots & \ddots & \vdots \\ b_{N1} & b_{i2} & \dots & b_{NM} \end{bmatrix}$$

Similarly to the transition matrix, the emission matrix is also row-stochastic. Each row is the probability distribution of the corresponding state emitting a possible observation (e.g. element b_{ij} is probability of state s_i emitting observation o_j).

Figures 2.2 and 2.3 show graphical representations of a HMM. Figure 2.2 shows the connections between the different states and between states and observations with regard to the transitions and emission probabilities. It is a partial representation of the transition and emission matrices in one graphical image. In Figure 2.3 we see a snapshot of a HMM while it progresses through time. We can observe that the occurrence of a state causes the occurrence of the next and at the same time causes the emission of an observation. It should be noted here that the indices $[..., t-1, t, t+1, ...]$ refer only to the time-point of the observed emission and the hidden state and not to their index within their respective groups ($[1, N]$ for states and $[1, M]$ for observations).

Having defined an HMM λ, there are three problems of interest that we need to solve in order for it to be applicable to real-life applications [24, 33]:

1. **Evaluation Problem:** Given an HMM λ and a sequence of observations $O = o_1, o_2, \dots, o_T$, what is the probability $P(O|\lambda)$ that the sequence is generated by the model λ?

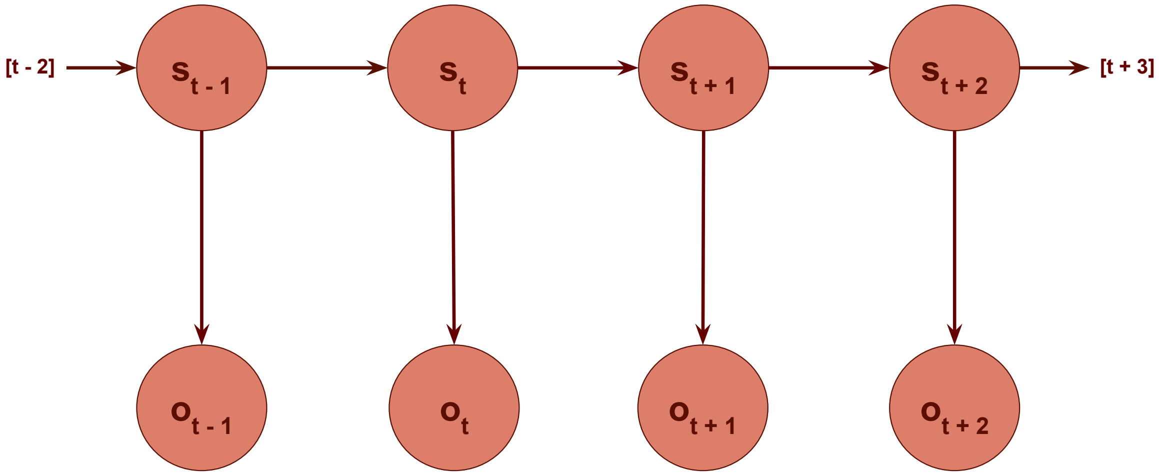


Figure 2.3: A Snapshot of a HMM while it progresses over time

2. **Decoding Problem:** Given an HMM λ and a sequence of observations $O = o_1, o_2, \ldots, o_T$, what is the most likely state sequence that could have produced the observation sequence?

3. **Learning Problem:** Given an HMM λ and a sequence of observations $O = o_1, o_2, \ldots, o_T$, how can the model parameters be adjusted so that the probability $P(O|\lambda)$ is maximised?

Fortunately, these problems have been solved and there are algorithms that provide the solutions:

Evaluation Problem | Forward Algorithm This algorithm introduces an auxiliary variable called forward variable, which is defined as $\alpha_t(i) = P(o_1, o_2, \ldots, o_t, s_t = i|\lambda)$. It represents the following probability:
Having observed the observation sequence until the t-th observation, what is the probability that the t-th state is $s_t = i$?

We can easily calculate $\alpha_1(i)$ for $1 \leq i \leq N$ as follows:

$$\alpha_1(i) = \pi_i b_{io_1},$$

which, for all possible i's will give us the probabilities of having observed the first observation originating from all possible states. The algorithm then uses the following recursive formula:

$$\alpha_{t+1}(j) = b_{jo_{t+1}} \sum_{i=1}^{N} \alpha_t(i) a_{ij}, 1 \leq j \leq N, 1 \leq t \leq T-1$$

Using this formula it is possible to calculate all possible $\alpha_T(i), 1 \leq i \leq N$. Finally, the probability that we are looking for can be calculated:

$$P(O|\lambda) = \sum_{i=1}^{N} \alpha_T(i)$$

This method has complexity $O(N^2T)$. Similarly we can define another auxiliary variable $\beta_t(i)$, called the backward variable:

$$\beta_t(i) = P(o_{t+1}, o_{t+2}, \ldots, o_T | s_t = i, \lambda)$$

This new variable gives the probability that, given a HMM λ and assuming that the tth state is $s_t = i$, the observation sequence from step $t+1$ to the end is $O = o_{t+1}, o_{t+2}, \ldots, o_T$.

For β_t the last time-step $\beta_T(i)$ is set first, for $1 \leq i \leq N$:

$$\beta_T(i) = 1$$

This is an arbitrary definition for β_T being 1 for all i. Then, $\beta_t(i)$ is recursively calculated:

$$\beta_t(i) = \sum_{j=1}^{N} \beta_{t+1}(j) a_{ij} b_{jo_{t+1}}, 1 \leq i \leq N, 1 \leq t \leq T$$

After calculating both the forward and backward variables we can alternatively calculate $P(O|\lambda)$:

$$P(O|\lambda) = \sum_{i=1}^{N} P(O, s_t = i|\lambda) = \sum_{i=1}^{N} \alpha_t(i)\beta_t(i)$$

Decoding Problem | Viterbi Algorithm For the decoding problem there are two alternative approaches to what the solution could be. One possibility is to try to find the states s_t that are individually most likely. This solution produces a sequence with the *highest number of correct states.* It requires a new variable,

$$\gamma_t(i) = P(s_t = i|O, \lambda) = \frac{\alpha_t(i)\beta_t(i)}{P(O|\lambda)} = \frac{\alpha_t(i)\beta_t(i)}{\sum_{j=1}^{N} \alpha_t(j)\beta_t(j)},$$

which gives the probability that the t-th state is $s_t = i$, given a HMM λ and an observation sequence $O = o_1, o_2, \ldots, o_T$. Using $\gamma_t(i)$ we can get all optimal states s_t by simply finding:

$$s_t = \arg \max_{1 \leq i \leq N} (\gamma_t(i)), 1 \leq t \leq T$$

This method, even though it produces a sequence with the highest number of the most likely states, has an important drawback. That is the fact that it completely disregards the **possible** transitions. This could result in a state sequence that includes transitions that are not even valid (it could be that $a_{ij} = 0$, but it is possible for the sequence to contain the transition $s_t = i, s_{t+1} = j$).

Therefore the most common approach to this problem is the Viterbi algorithm, which produces the most likely state sequence as a whole. We introduce yet another variable:

$$\delta_t(i) = \max_{s_1 s_2 \ldots s_{t-1}} P(s_1, s_2, \ldots, s_{t-1}, s_t = i, o_1, o_2, \ldots, o_{t-1}|\lambda)$$

This variable gives the maximum probability that partial observation and state sequences up to step t can have, given that we currently are at state $s_t = i$. Again, to initialize, for $1 \leq i \leq N$:

$$\delta_1(i) = \pi_i b_{io_1}$$

And the recursive formula is:

$$\delta_{t+1}(j) = [\max_{1\leq i\leq N} \delta_t(i)a_{ij}]b_{jo_{t+1}}, 1 \leq i \leq N, 1 \leq t \leq T-1$$

So, after calculating $\delta_T(j), 1 \leq j \leq N$, we can find state j_T^* which maximizes $\delta_T(j_T^*)$:

$$j_T^* = \arg\max_{1\leq j\leq N} \delta_T(j)$$

Starting from this state, it is possible to back-track the most likely state sequence using each j_t^* as a pointer for the optimal state of the previous step.

Learning Problem | Baum-Welch Algorithm The learning problem is in a way a problem of "training" a HMM, given a set of observation sequences, which we need to model, having little or no prior knowledge about how the model works (matrices A and B and vector π).

The Baum-Welch algorithm uses the Expectation-Maximization (EM) method, in order to estimate a set of parameters for a HMM $\lambda = (A, B, \pi)$, which maximize $P(O|\lambda)$ at least locally. The basic concept of every EM algorithm is that it calculates a set of parameters, which are then re-estimated in order to maximize a certain quantity (in our case $P(O|\lambda)$), that is expressed via the auxiliary function:

$$Q(\lambda, \bar{\lambda}) = \sum_s P(s|O, \lambda)\log(P(O, s, \bar{\lambda}))$$

This algorithm uses the variables $\alpha_t(i), \beta_t(i)$ and $\gamma_t(i)$ introduced in the two previous problems. It also introduces another variable, that similarly to $\gamma_t(i)$ can be expressed using $\alpha_t(i)$ and $\beta_t(i)$:

$$\xi_t(i,j) = P(s_t = i, s_{t+1} = j|O, \lambda) = \frac{P(s_t = i, s_{t+1} = j, O|\lambda)}{P(O|\lambda)} = \frac{\alpha_t(i)a_{ij}\beta_{t+1}(j)b_{jo_{t+1}}}{\sum_{i=1}^N \sum_{j=1}^N \alpha_t(i)a_{ij}\beta_{t+1}(j)b_{jo_{t+1}}}$$

This last variable gives the probability of being at state $s_t = i$ at the tth step and moving to state $s_{t+1} = j$ at the $(t+1)$th step. Having defined $\xi_t(i,j)$, we can now express variable $\gamma_t(i)$ defined in the decoding problem as:

$$\gamma_t(i) = \sum_{j=1}^N \xi_t(i,j), 1 \leq i \leq N, 1 \leq t \leq T$$

At this point it is simple to describe the learning process. We initialize the model's parameters, either randomly, or based on any prior knowledge we may have about the model $\lambda = (A, B, \pi)$. We then calculate the α's, β's, γ's and xi's based on the formulas given earlier. We re-calculate the model's parameters in order to maximize the quantity $P(O|\lambda)$, using the following formulas:

$$\pi_i' = \gamma_1(i), 1 \leq i \leq N$$

which is the expected number of times the model has started from state $s_1 = i$ at the first step,

$$a_{ij}' = \frac{\sum_{t=1}^{T-1} \xi_t(i,j)}{\sum_{t=1}^{T-1} \gamma_t(i)}, 1 \leq i \leq N, 1 \leq j \leq N$$

which is the expected number of transitions from state $s = i$ to state $s = j$ relative to the transitions from state $s = i$ and

$$b'_{jo_k} = \frac{\sum_{t=1}^{T} 1_{o_t=o_k}\gamma_i(t)}{\sum_{t=1}^{T}\gamma_i(t)}, \text{ where } 1_{o_t=o_k} = \begin{cases} 1, & \text{if } o_t = o_k \\ 0, & \text{otherwise} \end{cases}$$

which is the expected number of observations being o_k while being in state $s = i$ relative to the expected number of times being in state $s = i$.

While the Baum-Welch algorithm cannot guarantee finding a global maximum, it guarantees that it will definitely converge to at least a local maximum.

Different Types of HMMs

Observations As described earlier, matrix B gives the probability that a certain observation is emitted by a certain state. Every row of the matrix represents the probability distribution of the observations for the corresponding state. Naturally since an HMM models real-life processes, it is possible that the observations can be either discrete or continuous. In either case the probability distributions have the fitting type [20].

Infinite Duration HMM In theory all HMMs are infinite duration, unless specifically designed otherwise, meaning that the process can move from state to state forever, and practically the length of the sequences (either state or observation) is only restricted by the real-life constraints of the process (it usually stops at some point) [20].

In the case our data, due to its nature, we will be using an HMM with continuous probability distributions and of infinite duration (no designated *exit* state will be defined). The setup of our experiments is discussed in Section 3.1.

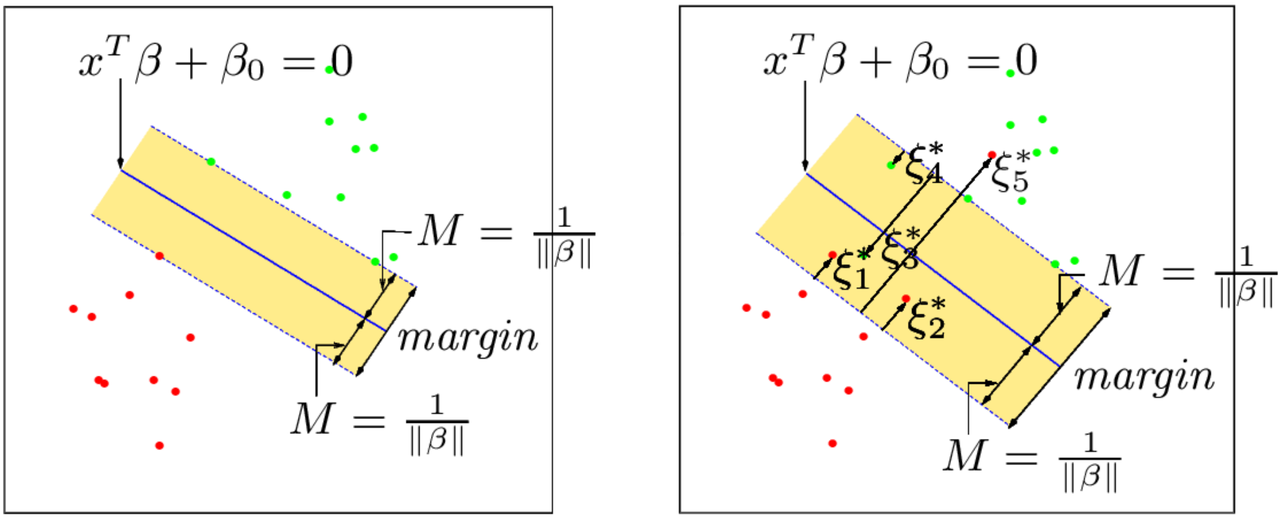
2.3 Support Vector Machines

A **Support Vector Machine** (**SVM**) is a *supervised learning* algorithm, which is used for binary data classification and also, less frequently and with certain extensions, for regression tasks and also multi-labelled data classification. The SVM has a very robust performance when it comes to data noise and sparsity, which makes it very effective and very popular in a great variety of applications. When it comes to binary classification the main goal of an SVM is, given a labelled dataset, to find the optimal separating hyperplane such that it correctly classifies the training data and generalizes in the best way possible with unseen data.

Let D be a training dataset that contains N pairs $(x_1, y_1), (x_2, y_2), \dots, (x_N, y_N)$, where $x_i \in \mathrm{IR}^{\mathrm{P}}$ is a P-dimensional vector representing a single data-point and $y_i \in \{-1, 1\}$ is the label of a single data-point.[1] The data-set is assumed to be perfectly separable. Given this training set, the SVM will define a hyperplane $x : f(x) = x^T\beta + \beta_0$, where β is a unit vector. $f(x)$ gives the signed distance of a point x to the hyperplane. If vector β is adjusted correctly, then the hyperplane can be separating the data-points that belong to the two classes, with $y_i f(x_i) > 0 \forall i$, while also optimizing the hyperplane so that the distance of the data-points that lie closest to it (*margin*) is maximum. Figure 2.4a shows such a separating hyperplane, for the case where the training set consists of 2D points. It can be shown that the width of the margin is equal to $2M = \frac{2}{\|\beta\|}$. This leads to the following inequality:

$$y_i(x_i^T\beta + \beta_0) \geq M, i = 1, \dots, N, \tag{2.1}$$

[1] Notation in this section follows *The Elements of Statistical Learning*[16]



(a) Perfectly separable data-set

(b) Non-linearly separable data-set

Figure 2.4: Support Vector Classifiers (Figure taken from *The Elements of Statistical Learning* [16])

meaning that all points are lying at a minimum distance M from the hyperplane, depending on their class. The data-points, that lie on the boundaries of the classes (distance $\frac{1}{\|\beta\|}$ on either side of the hyperplane) and, therefore, are the ones that define the margin and determine the position of the hyperplane, are called **support vectors**.

It is very possible to run into cases, where the two classes are not perfectly separable by a hyperplane. In order to overcome this issue, an approach has been developed, that still makes it possible to calculate the optimal hyperplane with the maximum margin. That is, to introduce the so-called *slack variables* $\xi = (\xi_1, \xi_2, \dots, \xi_N)$ and modify the inequality previously introduced (Eq. 2.1) as follows:

$$y_i(x_i^T\beta + \beta_0) \geq M(1 - \xi_i), i = 1, \dots, N, \xi_i \geq 0, \sum_{i=1}^{N} \xi_i \leq C.$$

These slack variables, indicate the extent to which each data-point is “allowed” to violate the class’ borders. This could mean that certain data-points may lie on the right side of the hyperplane, but within the margin (correctly classified), or they may lie on the wrong side of the hyperplane (misclassified). Each variable ξ_i is the proportion to which, the distance $f(x)$ is on the wrong side of the hyperplane. ξ_i’s values can be divided into the following areas:

$$\xi_i \begin{cases} = 0, & \text{if } x_i \text{ lies on the correct side of the hyperplane} \\ \in (0,1), & \text{if } x_i \text{ lies within the margin (correct classification)} \\ = 1, & \text{if } x_i \text{ lies on the hyperplane} \\ > 1, & \text{if } x_i \text{ is missclassified} \end{cases}$$

With the use of these variables, the classifier is still able to define a hyperplane with maximum margin, at the expense of having a number of points either misclassified, or lying within the margin. By applying the constraint $\sum \xi_i \leq C$ we define the “strictness” of the training procedure, restricting the number of misclassifications. This whole alteration and use of the slack variables is displayed in Figure 2.4b.

In the case that the given dataset is not linearly separable, even with the use of the slack variables, it is sometimes possible to define a linear separating hyperplane, by

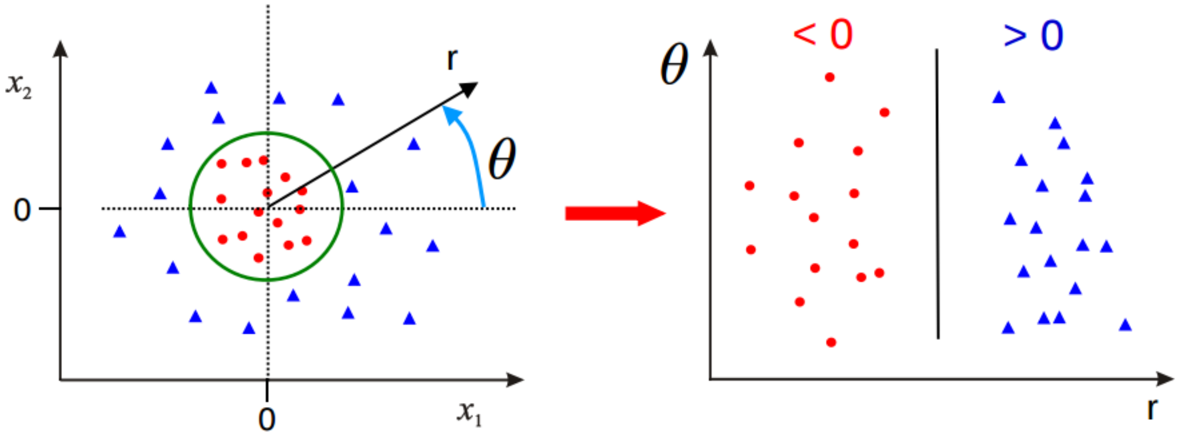


Figure 2.5: Using Polar Coordinates to achieve linear separation(Figure taken from the Lecture notes for the Machine Learning Course at the University of Oxford [38])

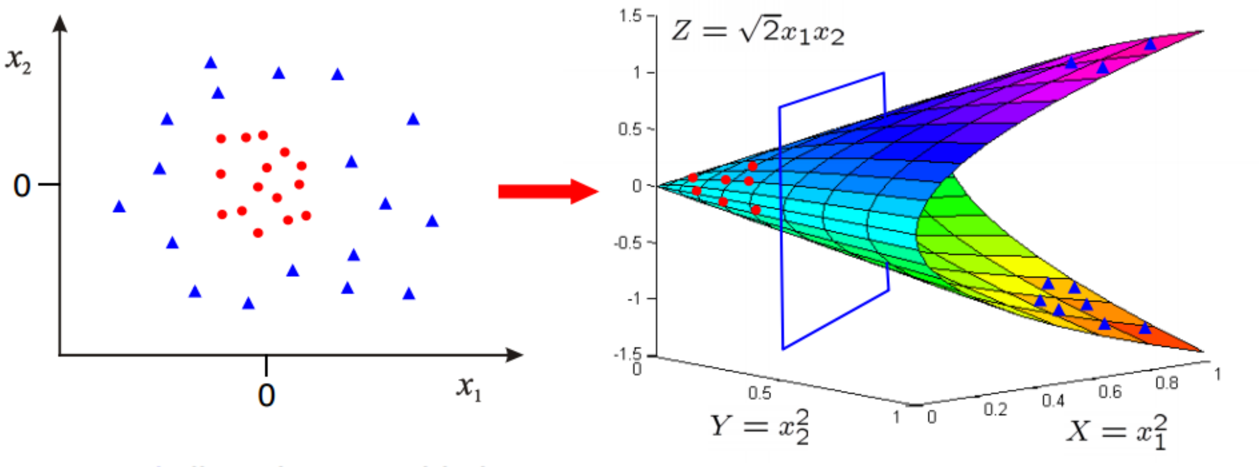


Figure 2.6: Mapping data from 2D to 3D to achieve linear separation [38]

mapping the data to a higher dimension, or by expressing the data using a different co-ordinate system. In general, a linear solution can be found by defining a different feature space, in order to map the data and render it linearly separable.

Figure 2.5 shows the case where the dataset is not linearly separable. When the data is expressed using the polar co-ordinate system, it becomes perfectly separable, a very simple and "easy" case for an SVM classifier. Figure 2.6, on the other hand, shows the same data-set, where the approach for the classification is different. In this case, the data has been mapped to a higher dimension ($2D \rightarrow 3D$), where again it is perfectly separable by a hyperplane of a higher dimension.

In both cases the data-points have been replaced by their representation in a new feature space, using the *feature map* $\Phi(x)$. This way the initial hyperplane function is expressed:

$$f(x) = \Phi(x)^T\beta + \beta_0.$$

Using this feature map, during training and during the final classification of the data, in calculations, only the inner product $\Phi(x_j)^T\Phi(x_i)$ occurs. The need for knowledge of the mapping $\Phi(x)$ is eliminated and the need for knowledge of the inner product rises. Hence, the algorithm uses a **kernel function**, which is defined as follows:

$$k(x_j, x_i) = \Phi(x_j)^T \Phi(x_i).$$

The use of the kernel method simplifies the calculations occurring during training and lightens the computational load. More importantly it converts the computational complexity of the algorithm to a function of the size of the data-set (N) from a function of the dimensionality of the data-set. This means that the algorithm can handle large feature spaces that potentially carry more information.

Some of the most popular kernels are:

- **Linear**: $k(x, x') = x^T x'$
- **Polynomial**: $k(x, x') = (1 + x^T x')^d, d > 0$
- **Radial Basis Function**: $k(x, x') = \exp(-\gamma \|x - x'\|^2))$

2.4 Proposed Methods

As discussed in Section 1.4 the aim of the thesis is to study the course of the MCI condition throughout time by examining brain structure (volumetric features extracted from brain MRI scans using Freesurfer (Sec. 2.1)).

The basis of the methods that we developed are HMMs. We chose HMMs because they inherently study sequential data. Their structure is such that they fully represent Markov Chains which are the hidden states and their emissions (observations from our perspective), which is what can be observed. Even though, by default, they are not specialised in temporal data, but in Markov Chains, they are most commonly used for time-sequential data (e.g. speech processing) and they can very nicely monitor and construct all sequential relations existing in the data. In our case this is very useful since we can feed the longitudinal MRI scans as observations and let the HMM locate the relations between them and most importantly the hidden structure that is the Markov Chain.

We have developed three different methods, each one built upon the previous. These methods will be discussed and explained in the following sections.

Notations and Data Separation At this point, it is fitting to present certain basic notations that will be used in the following sections, as well as the way that the data will be divided and used. As described in Section 2.1, the dataset consists of a number of MRI scans that belong to different subjects. Each subject has a series of scans that consists of the **cross-sectional** scan, which is the initial scan, and a number of **follow-ups** (one, two or three).

There are two ways to divide the data-set. One way is based on the diagnosis that is formed based on the cross-sectional scan, which is referred to as **subject-initial-group** and can be **cognitively normal (CN)**, **mild cognitive impairment (MCI)** or **Alzheimer's disease (AD)**. This separation group considers *only* the baseline diagnosis, completely disregarding any of the follow-ups' diagnoses. The other way is to group the subjects based on their final follow-up. This grouping is referred to as **subject-end-group** and produces the same diagnoses/categories as the subject-initial-group (**CN**, **MCI** & **AD**). Similarly to the subject-initial-group, this separation considers only the diagnosis derived from the final follow-up scan.

Even though this type of separation of the data is derived naturally from the clinical conditions of the subject, within the scope of this thesis it is more practical to divide it differently. The subject-initial-grouping and subject-end-grouping still remain, however the labels that they get assigned change. For the subject-initial-group, the CN and AD

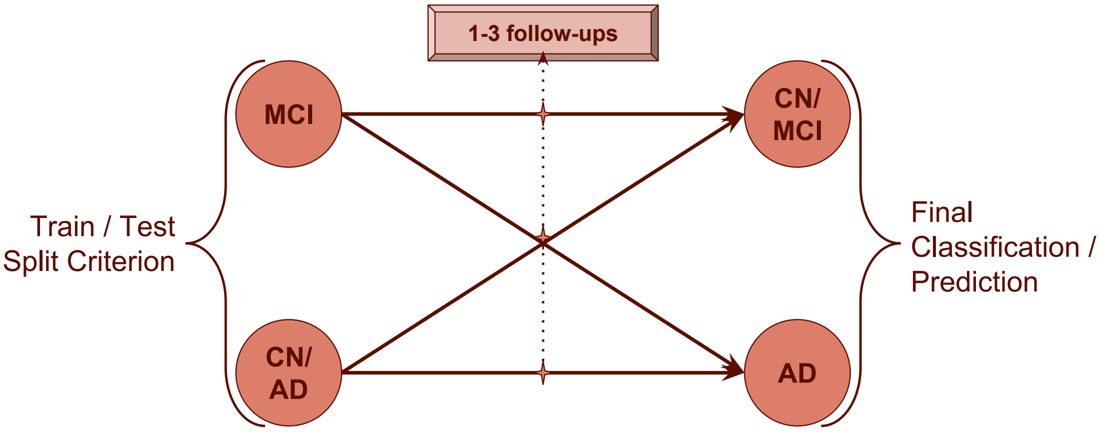

1-3 follow-ups
MCI
CN/ MCI
Train / Test Split Criterion
Final Classification / Prediction
CN/ AD
AD

Figure 2.7: Separation of the dataset based on the diagnoses of each subject's cross-sectional and final MRI scans

groups are merged, so the subject-initial-group can now be either **non-MCI** or **MCI**. This is more fitting, because the aim is to study the progression of the MCI subject-initial-group, since it constitutes a high-risk group. Following the same concept, we wish to detect whether the MCI will deteriorate to AD or not. Therefore the subject-end-group can be either **AD** or **non-AD** (Figure 2.7).

The subject-initial-group defines the training and testing sets for our models as well, with the non-MCI being the training and the MCI being the testing sets. This happens mainly because we are interested in MCI patients and how they can progress over the years. As discussed in Section 1.4, MCI is a group of interest within the medical research field. It covers a wide range of cognitive impairment among individuals, which can progress to different conditions and most importantly constitutes a higher risk for developing AD. It is therefore our intention to study how well our systems will be able to predict the progression of this particular group. The use of non-MCI subjects as a training set is based on the attempt to see how well a HMM can derive basic and generic structural changes of the progression towards AD or non-AD (convert to CN or remain stable). We then want to see how well these derived features can match or can be applied to the MCI group (*during experimentation, we actually use part of the MCI group for the training process, to test the effect on the overall performance of all the methods*).

2.4.1 Method I: HMM Classification

The first approach uses only HMMs in an attempt to explore how well they can extract and model information about the temporal structural changes of a brain when it is on the path towards AD or when it ages in a healthy manner. Then, using this information, we aim to examine how similar these changes are to the changes of a brain diagnosed with MCI.

As described in Section 2.2 (**Learning Problem**) an HMM can be trained in such a way, so that the probability $P(O|\lambda)$ is maximized, where λ is the HMM model and $O = [o_1, o_2, \ldots, o_T]$ is a sequence of observations. In the case of our data-set (Section 2.1), one observation sequence O is one longitudinal MRI scan sequence of a subject, where

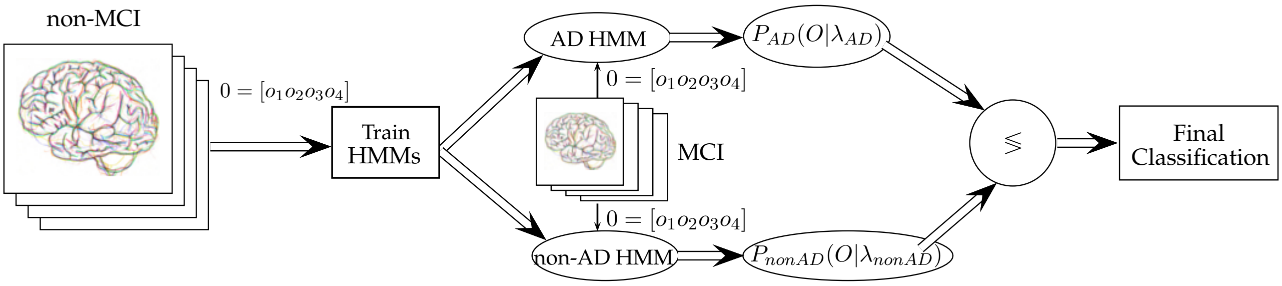


Figure 2.8: Work-flow Diagram for the HMM classification Method

each one of the observations o_t are vectors of size $[1 \times 55]$ representing the volumetric data extracted from each scan, with $T \in [2, 4]$.

By stacking the $[1 \times 55]$ sized vectors o_t, we initially produce the observation sequences of all the subjects ($[n \times 55], n \in [2, 4]$). Then the observations of the non-MCI subject-initial-group are used in order to train two HMMs, λ_{AD} and λ_{non-AD}, using for each HMM only the observations with subject-end-groups AD and non-AD correspondingly. After the two HMMs have been trained, we then use the MCI subject-initial-group to test it. With the use of the forward algorithm (Section 2.2, **Evaluation Problem**), two probabilities are produced for each observation sequence, $P_{\text{AD}}(O_i|\lambda_{\text{AD}})$ and $P_{\text{non-AD}}(O_i|\lambda_{\text{non-AD}})$, indicating the probability that each sequence could be produced by the corresponding HMM. Based on these probabilities, the prediction for the sequence is decided:

$$y_i = \begin{cases} \text{AD}, & \text{if } P_{\text{AD}}(O_i|\lambda_{\text{AD}}) \geq P_{\text{non-AD}}(O_i|\lambda_{\text{non-AD}}) \\ \text{non-AD}, & \text{if } P_{\text{AD}}(O_i|\lambda_{\text{AD}}) < P_{\text{non-AD}}(O_i|\lambda_{\text{non-AD}}) \end{cases}$$

It should be noted at this point that we do not assign specific meaning to the states of the HMMs. When using HMMs, it is a common practice to choose the number of states in such a way, that each state, or a group of states represent a certain “logical state” of the actual process. In our case, however due to the sparsity of the scans it is not possible to model the sequences in such a way. That is why, from now on, we consider the states to be an independent variable of the system, one that we can change at will and study its effect on the overall performance of the system.

A descriptive diagram of this method is shown in Figure 2.8.

2.4.2 Method II: HMM Modelling - SVM Classification

While the first method produces satisfactory results (Section 3.2.2), it is presumed that there is room for improvement. We would additionally like to study the ability of the HMM to extract and model the information included in the data in a way that can be usable by other models and methods. Even though we earlier discussed the fact that we do not assign explicit meaning to the states of the HMM, we work based on the premise that after it has been trained the HMM has defined some explicit meaning to its states, which is modelled by the transitions matrix.

HMMs by nature consider their states to be Markov Chains (Section 2.2). Therefore, when an HMM is trained, it looks into the structure of the data regarding the observations’ variable (time in our case). It detects any recurring or not patterns present in the observation sequences and structures the probabilities (initial, transition and emission) accordingly.

The following example will motivate and explain the choices made for this method:

Let’s try to model the behaviour/everyday life of an adult, depending on certain observations we’ve made on their daily life. Our data contains observations of adults who can be categorized as *working* and *unemployed.* Our observations are "screen-shots" of what the adults are doing taken throughout the day at fixed intervals. Some observations can be: *sitting in the train/bus*, *sitting in front of a computer*, *lying in bed*, *eating* etc. As the two types of adults lead different lives, their observation sequences will also differ. So when we train an HMM it will adjust its probabilities, so that the state Markov chains will have a structure fitting to each adult and will produce corresponding observation sequences.

If it were possible to pre-model the state transitions, we could initialize the transition matrix in such a way, so as to guide the HMM to assign a certain meaning to the states. In the above example we could assign meanings to the states like: *commuting to/from work*, *working*, *sleeping*, *resting* etc. In this case we could initialize higher probabilities for state transitions such as: *commuting to/from work* to *working*, *working* to *commuting to/from work*, *commuting to/from work* to *eating*, rather than state transitions like: *working* to *sleeping* and *sleeping* to *working.* Then the HMM would structure the state Markov chains, so that they would behave in a way easier for us to understand.

However, even though this state-to-actual-action matching is easier for us to understand and explain, it does not necessarily help the HMM itself. If we don't match a state to an action, the HMM will still be able to structure the Markov chains in a way fitting to the behaviour of the data, even though it may not be as obvious to us. The HMM may even assign the same action to a number of different states. The fact remains, though, that the overall structure of the states and their behaviour will be fitting to the behaviour of the data used during the training. ■

This part of the HMM modelling is exactly what we try to exploit in this method. We theorize that the state sequences corresponding to our observation sequences, carry important information and can be used as features for a different model/classifier.

Based on this logic and building upon the method of the previous section, we intend to produce state sequences/features, which we will then use in order to train an SVM classifier.

The method prepares the data as before, producing observation sequences $O_i = [o_1, o_2, \ldots, o_T]$ for all subjects. An HMM is then trained using the non-MCI subject-initial-group. The difference from the previous method is that now, only *one HMM is trained*, regardless of the subject-end-group of each observation. This is done that way, because we aim at a more generic feature extraction from the observations and want to explore the HMM's ability to model and define the discriminative information on its own.

After the HMM λ is trained, it is used to produce state sequences for all the observation sequences (subject-initial-groups non-MCI and MCI) (Section 2.2, **Decoding Problem, Viterbi Algorithm**). These state sequences are used as features for the next phase of the method. The feature sequences of the non-MCI subject-initial-group are used in order to train an SVM classifier. The SVM is trained to binary classify the data into AD and non-AD, according to each sequence's subject-end-group. After the training, the SVM is tested on how well it is able to classify the MCI subject-initial-group sequences into the two classes (AD, non-AD).

A descriptive diagram of this method is shown in Figure 2.9.

2.4.3 Method III: HMM Modelling - SVM Classification II

Due to the nature of the data (more will be discussed in later sections), the state sequences produced by the HMM are quite unstable and display high variance, especially while the number of states increases. Additionally, the features that are produced (the unaltered state sequences), inevitably have varying lengths, according to the length of the observation sequence that produced it. This causes instability to the classification and interferes with the final prediction.

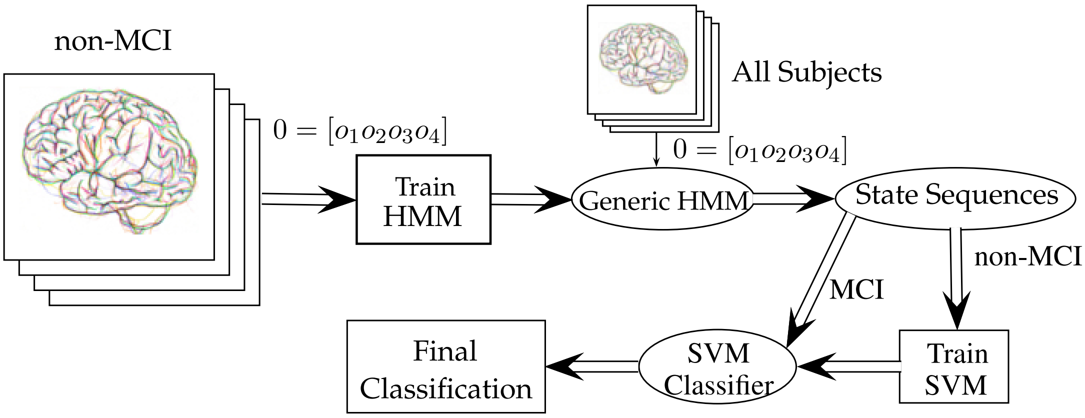


Figure 2.9: Work-flow Diagram for the HMM modelling with SVM classification Method

After producing state sequences for different numbers of states, it has been noticed that the state sequences of the non-AD subject-end-group follow a steadier path than those of the AD subject-end-group, both for the non-MCI and the MCI data. There are fewer inter-state transitions occurring for the non-AD subjects. This difference between the erratic and steadier behaviour of the AD and non-AD subject-end-groups can be a result of a similar behaviour of the actual structure of the brain when it comes to normal and abnormal ageing. During normal ageing the different areas of the brain that we are following are changing in a quite steady and usually low pace and they keep the same monotony (area x will keep decreasing throughout the years). However, during abnormal ageing these changes become more abrupt and harder to following precisely because of their erratic behaviour. Though it is still not clear if the erratic changing is the cause or a result (or even something in-between) of abnormal ageing, it is certainly highly connected and correlated.

When counting the transitions occurring we are interested in those that lead the sequence to the same state, or to a different one, i.e:

$$s_t \to s_{t+1} : \begin{cases} \text{same-state transition,} & \text{if } s_t \equiv s_{t+1} \\ \text{inter-state transition,} & \text{if } s_t \not\equiv s_{t+1} \end{cases}$$

We therefore keep track of the same- and inter-state transitions that occur in each group and end-group, as well as the total number of transitions. As a showcase, we consider the following example:

We have the following state sequences of non-MCI subjects with AD end-group, produced by a 10-state HMM:

$$S_1 = \{2, 2, 3, 1\}$$
$$S_2 = \{1, 2, 3, 4\}$$
$$S_3 = \{1, 1, 7\}$$
$$S_4 = \{9, 8, 8, 2\}$$
$$S_5 = \{5, 6\}$$

Table 2.3

	MCI Subject Group		**non-MCI Subject Group**	
Transition Type\End-Group	non-AD Subjects	AD Subjects	non-AD Subjects	AD Subjects
Same-State Transitions (μ, σ)	77.1%, 0.028	62.9%, 0.043	78.7%, 0.023	64%, 0.044
Inter-State Transitions (μ, σ)	22.9%, 0.028	37.1%, 0.043	21.3%, 0.023	36%, 0.044

For these sequences we can count the following transitions:

S_1 :	1 same-state	$2 \to 2$	2 inter-state	$2 \to 3, 3 \to 1$	3 total
S_2 :	0 same-state		3 inter-state	$1 \to 2, 2 \to 3, 3 \to 4$	3 total
S_3 :	1 same-state	$1 \to 1$	1 inter-state	$1 \to 7$	2 total
S_4 :	1 same-state	$8 \to 8$	2 inter-state	$9 \to 8, 8 \to 2$	3 total
S_5 :	0 same-state		1 inter-state	$5 \to 6$	1 total

So, assuming that these 5 subjects are the only non-MCI subjects with AD end-group, we would then calculate:

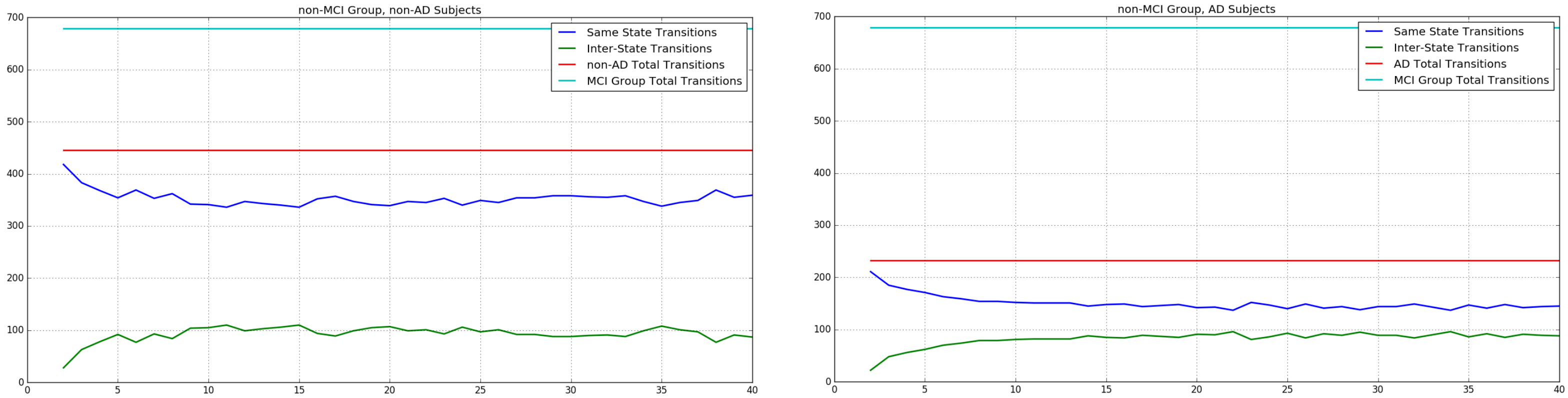
non-MCI group, AD end-group: 3 same-state 9 inter-state 12 total

which would indicate that 25% of the transitions occurring within the AD end-group of the non-MCI subjects are same-state and 75% are inter-state. ■

By calculating these percentages for all groups and end-groups, we can examine interesting and potentially useful differences that the HMMs have highlighted. Figures 2.10 & 2.11b show the same- and inter-state transition count for the different groups for HMMs with increasing number of states, along with the total transition counts of the corresponding groups as a frame of reference. The total transitions are of course constant throughout the graphs, because they are merely related to the lengths of the sequences (number of follow-up scans of the subjects) and not to the structural analysis of the HMMs. We can observe in the figures that the number of same-state transitions is significantly higher than the number of inter-state transitions for the non-AD subjects ($\sim 3-4$ times higher), both for non-MCI and MCI, rather than the AD subjects ($\sim 1.5 - 2$ times higher).

This observation can be verified by Table 2.3. For Table 2.3, we have calculated the same- and inter-state transitions' percentage of the total number of transitions. We then calculated the mean and variance of the percentages over an increasing number of HMM states. From the table we can conclude that $\sim 78\%$ of the non-AD subjects' transitions are same-state as opposed to $\sim 22\%$ that are inter-state. On the other hand the AD subjects produce sequences, where $\sim 63 - 64\%$ of the transitions are same-state, hence our initial statement that the AD subjects produce sequences that follow a steadier path than the non-AD.

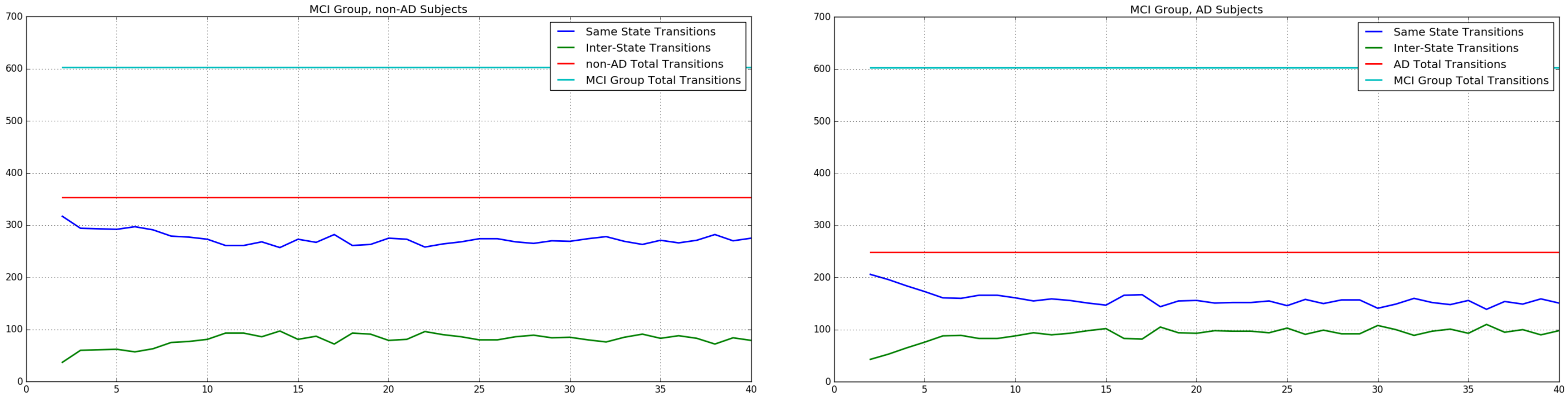
At this point, we attempt to exploit this characteristic and also attempt to remove the length variable from the produced features. We, therefore, produce the **transition frequency maps**, shown in Figures 2.12a and 2.12b. These maps are matrices of size $[N \times N]$, where N is the number of the HMM states and each element a_{ij} is a counter of the transitions from state i to state j. Naturally the elements of the matrix's diagonal represent the same-state transitions and all other elements represent the inter-state transitions. It should, once again, be noted that we regard the number of the HMM states as a variable, which we can tune and examine how it affects the performance of our system. Hence, the 13 states of Figures 2.12 have no logical significance but are rather used as an example.



(a) non-AD end-group

(b) AD end-group

Figure 2.10: Count of state transitions occurring for the non-MCI group for HMMs trained with increasing number of states. x-axis indicates the number of the HMM's states and y-axis indicates the transition count.



(a) non-AD end-group

(b) AD end-group

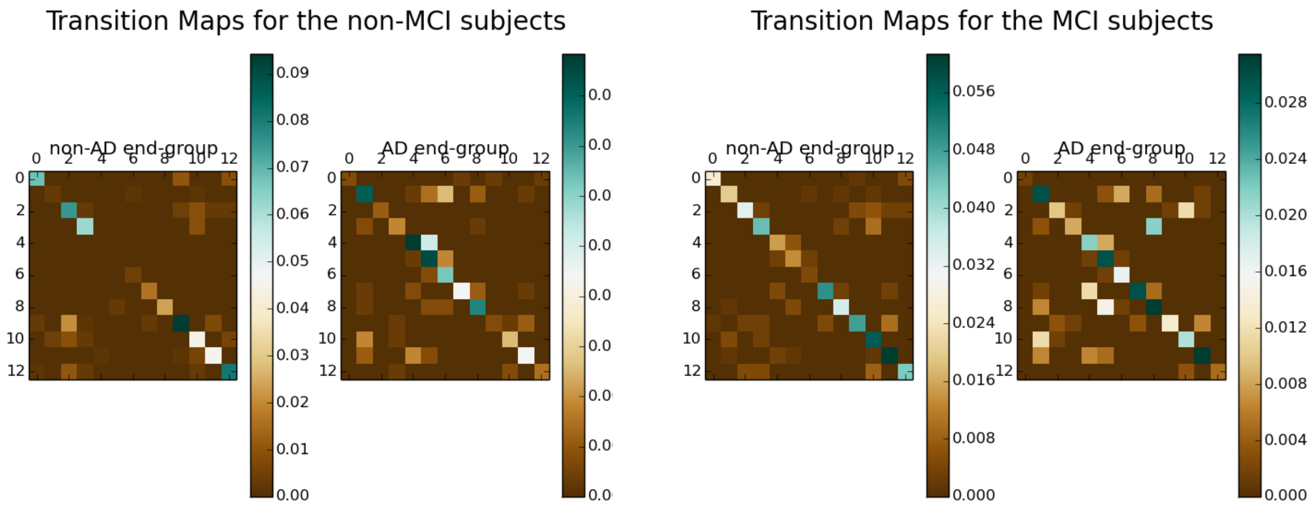
Figure 2.11: Count of state transitions occurring for the MCI group for HMMs trained with increasing number of states. x-axis indicates the number of the HMM's states and y-axis indicates the transition count.

These matrices are serialized row-wise and then used as the feature vectors of the subjects. This way, our features are now freed from the temporal element, which is included within their structure, but does not affect their length, which would be a practical inconvenience.

It can be deduced by the nature of the initial matrices, that the feature vectors are very sparse, including mostly zeros except for a few non-zero elements that can take values $a_{ij} \in [1, 2, 3]$. Hence, the positions of the non-zero elements are of greater significance than their actual values. For an SVM classifier this constitutes a much simpler task, compared to the separation of the state sequences used in the previous method. This task is simpler, because for an SVM it can be perceived as a 2D data spacial separation.

When it comes to the steps followed by this method, they are very similar to the previous method's steps. The data is first prepared and the observation sequence $O_i = [o_1, o_2, \ldots, o_T]$ are produced for all the subjects. One HMM is again trained using the non-MCI subject-initial-group.

We then produce state sequences for all the observation sequences, which we use to produce the transition maps and finally the feature vectors, as described previously. These feature vectors are now used to train the SVM classifier (non-MCI training set) so that



(a) Transition Frequency Maps of non-MCI subject-initial-group

(b) Transition Frequency Maps of MCI subject-initial-group

Figure 2.12: Transition Frequency Maps Produced by a 13-State HMM. Rows and Columns represent the 13 States.

it can once again be able to classify the data into AD and non-AD, according to the subjects' end-groups. In the end the classifier is tested on the MCI testing set.

Comparing Methods II and III we can see that they only differ the nature of the features fed into the SVM classifier.

A descriptive diagram of this method is shown in Figure 2.13.

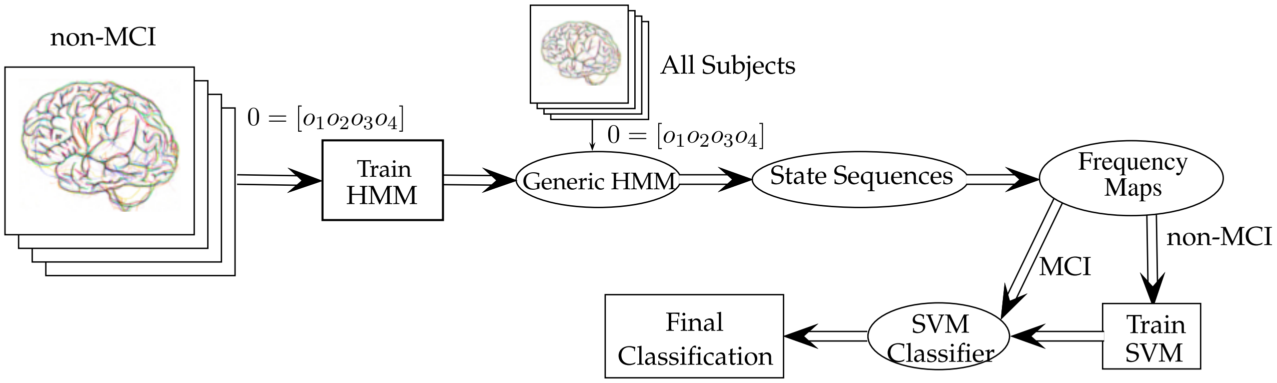


Figure 2.13: Work-flow Diagram for the HMM modelling with SVM classification Using Transition Maps Method

2.4.4 Practical Information

All three methods that have been described in the previous Sections have focused on excluding the MCI subject-initial-group from the entire training phase and only use it to test the system that has been designed. The rationale is that we want to explore the informative "strength" of the longitudinal MRIs in providing patterns of the structure of the brain while it ages towards AD or normally and how well this information applies to the MCI data.

However, on a more practical approach, when building a prediction system it's best to use as much available data as possible so that the system can include that information when making a prediction decision. Therefore, during the experiments we have also included altered versions of these methods, where we include part of the MCI data in the training process. This will be described in more detail in Section 3.1.

2.5 Evaluation Metrics

For the evaluation of the developed methods a certain group of metrics has been used. These are described in this Section.

True Positives (TP) The number of subjects that have correctly been classified as AD.

True Negatives (TN) The number of subjects that have correctly been classified as non-AD.

False Positives (FP) The number of subjects that have incorrectly been classified as AD, but are actually non-AD.

False Negatives (FN) The number of subjects that have incorrectly been classified as non-AD, but are actually AD.

Sensitivity (true positive rate - TPR) The proportion of the positive (AD) samples that have been correctly classified:

$$\text{Sensitivity} = TPR = \frac{TP}{TP + FN}$$

Specificity (true negative rate - TNR) The proportion of the negative (non-AD) samples that have been correctly classified:

$$\text{Specificity} = TNR = \frac{TN}{TN + FP}$$

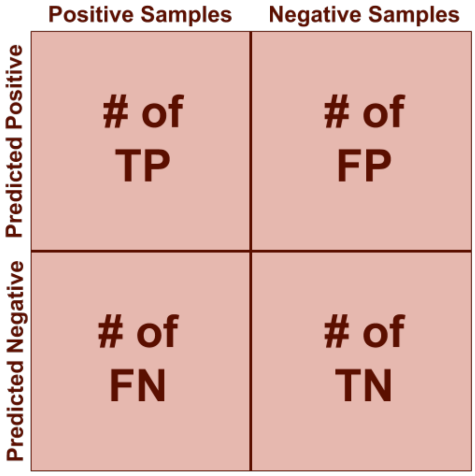
Precision (positive predictive value - PPV) The proportion of the correctly classified samples as positive (AD) among the total number of positively classified samples.

$$\text{Precision} = PPV = \frac{TP}{TP + FP}$$

Confusion Matrix A matrix that summarizes all previous metrics. The rows of the matrix represent the actual labels of the data, while the columns represent the predicted label by the classifier. Each element is a count of the data-points that fall into each category, elements of the diagonal represent the "true rates" (either positive or negative), while all other elements represent the "false rates". Confusion matrices can be helpful even in the cases of multilabel classification. Naturally, it is expected that the matrix has the highest counts along the diagonal and lower counts or even zeros scattered around it.

F1 score The harmonic mean of precision (PPV) and sensitivity (TPR):

$$F1 = 2\frac{PPV \times TPR}{PPV + TPR}$$



	Positive Samples	Negative Samples
Predicted Positive	# of TP	# of FP
Predicted Negative	# of FN	# of TN

Figure 2.14: Confusion Matrix Example

Note: In our experiments we saw it more fitting to calculate the harmonic mean of specificity (TNR) and sensitivity (TPR):

$$F1 = 2\frac{TNR \times TPR}{TNR + TPR}$$

Receiver Operating Characteristic (ROC curve) The ROC curve is a graphical way of showcasing the general performance of a model, while a certain parameter is varied. It is the plot of the true positive rate - TPR (Sensitivity) against the false positive rate - FPR ($1 - specificity$). The ROC curve characterises a better model as it gets closer to the upper-left corner (point $(0, 1)$) of the ROC space indicating the model exhibits high TPR and low FPR. The diagonal in the ROC space ($TPR = FPR$ for all values of the varying parameter) is characterising a random classifier and any curve below the diagonal is characterising a bad classifier (has more misclassifications than correct classifications). In general the closer the curve gets to the upper-left corner the better the model.

When building a model (or a number of models that need to be compared), other than the visual comparison that can be performed on the models' curves (the same parameter should be varying for a more valid comparison), a common practice in order to quantify each model's performance is to calculate the *area under the curve* (AUC) and then compare the numbers.

The ROC space can also be used even in the case where it is not possible or desired to test the performance while varying different parameters. In that case the performance is visualised with one point in the ROC space ((TPR, FPR)). The same "rules" apply here as before, in terms of better/worse performance. The closer the point is to the upper-left corner, the better the performance and of course it is not desired that the point lies on the diagonal line or lower than it. In case different models need to be compared, a common practice is to calculate the Euclidean distance of each model's point from $(0, 1)$. The shorter the distance, the better the model. This practice is used also in the case of the ROC curves when one needs to determine the point of the curve that lies closer to $(0, 1)$, which would give the value of the varying parameter that produces the best results.

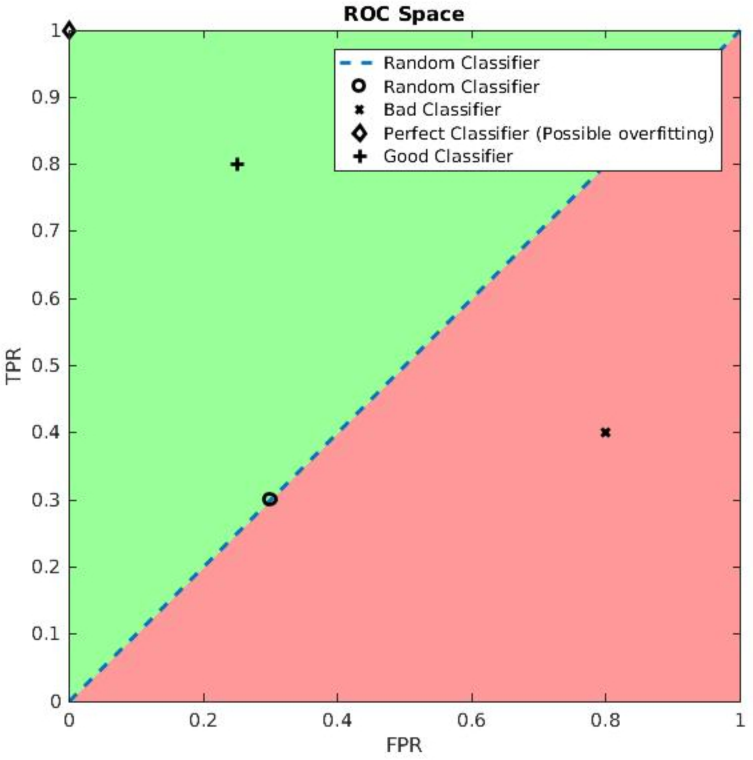


Figure 2.15: ROC Space Example

In Figure 2.15 we can see the ROC space and some of the possible model evaluations that we can get. If the model's performance lies either within the red area or upon the diagonal line, it means that the model i not doing very well. On the other hand we wish that the model lies within the green area. The closer to $(0, 1)$ it lies, the better its performance, however we don't really wish for it to be exactly on the corner. This may indicate a perfect classification but it could also be an indication of the model overfitting to the data.

Diagnostic Odds Ratio (DOR) The diagnostic odds ratio is a metric commonly used in medical research when there is binary classification in the form of positive/negative result (regarding a disease or a condition). It is defined as:

$$DOR = \frac{sensitivity \times specificity}{(1 - sensitivity) \times (1 - specificity)}$$

It can be defined as *the ratio of the odds of the test being positive when the disease is present relative to the odds of the test being positive when the disease is not present.* Its value ranges from 0 to infinity and higher values mean *better discriminatory test performance.* When the metric is 1, then the system fails to discriminate between the diseased and not-diseased subjects (similar to the random classifier of the ROC space). And finally when the value is lower than 1, then the metric indicates that the system has failed to interpret the data correctly (there exist more wrongly than correctly classified subjects).

Chapter 3

Results

This chapter presents the results and the evaluation of the experiments. The different attempts are compared, contrasted and commented upon.

3.1 Experimental Setup

The experiments are designed and executed using Python [12]. In order to build the HMM models, the toolkit *hmmlearn* [30] was used. This toolkit is open-source and offers a number of algorithms and models regarding HMM-learning and usage. As it used to be part of the *scikit-learn* [22] package (scikit-learn version 0.16 and earlier) it follows its API closely, but has been adapted to sequence data. Finally, scikit-learn has been used for the training and testing of the SVM classifier and other machine learning implementations (cross validation, evaluation metrics etc.).

The HMMs are initiated as fully connected. This means that before any training occurs the transitions matrix contains no zero-elements and transitions from all states to all states are possible. The emissions probabilities are constructed with Gaussian emission distributions (other choices are Gaussian Mixture Model and multinomial/discrete emissions) with a "spherical" covariance (each state uses a single covariance value that applies to all the features). The number of the HMM's states varies depending on the implementation and test-run.

When it comes to the SVM classifier, we decided to configure it before the experiments and keep this configuration unchanged for all different methods and approaches. The hyperparameters of a SVM that need optimizing are the type of the kernel, the penalty parameter C and the kernel coefficient gamma. In case of a polynomial kernel we are also interested in its independent term.

There are different approaches when it comes to setting hyperparameters. The usual way is to train the model iteratively over a range of possible values for the parameters. Due to constraints in time and hardware, we decided to do a simpler optimization, by optimizing one parameter at a time and then use it to optimize the rest. This way we gradually optimize all parameters, one at a time. This led us to use a polynomial kernel of degree 3. The penalty parameter C is set to 63.26 (since the data is sparse and few, the classifier needs to be "strict" with misclassifications). Finally the kernel coefficient gamma is set to 0.001 and the independent term of the polynomial function is 3.

Training Data As mentioned in paragraph 2.4.4 the methods that are developed are focused on excluding the MCI subject-group from the training process of both the SVM classifier and the HMM. In practice, when it comes to classification techniques and algorithms it is always better to have as much and as diverse training data as possible, so that the model does not over-fit. For this reason, we decided to run an extra round of our experiments, while including the MCI data in the training process, in an attempt to build a better system and to test if that would indeed improve the performance of the system.

Cross validation is a method widely used to evaluate the performance of different models and their ability to generalize their behaviour on unseen data. One issue that rises when training and testing a model is that during training the model learns how to perform well on the training data (or at least as well as possible), so when evaluating on its performance there, the results cannot be trusted. It therefore needs to show how well it performs on unseen data and that performance is the one that actually matters. k-fold cross validation is a common practice to overcome this issue. The data is divided into k subsets, where the $k-1$ are combined and used as training set and the k-th is used as unseen testing set. The process of training and testing is repeated k times and each one of the subsets is used as testing set once and as part of the training set $k-1$ times. At the end of the cross validation the evaluation metrics or errors that have been produced by each run are averaged to a final performance measure. It is common practice to use $\sim 10-20\%$ of the data as testing set and use the rest as training set ($k = 10-5$ folds respectively).

In our experiments we follow the logic of cross-validation in order to include the MCI data in the training process. Initially the data is already divided into training and testing sets (non-MCI and MCI subject-initial-groups correspondingly). These two sets have almost equal numbers of data-points (309 for the MCI and 322 for the non-MCI). What we aim at, at this point, is to include an part of the MCI subjects (testing set) in the non-MCI subjects (training set).

At this point we decided to divide the MCI group into $k = 3$ folds (103 subjects per fold) and merge that fold with the training set, while the other two are used as testing sets. We decided on this number in order to include a *considerable* amount of MCI subjects into the training set, so that it would actually have some impact on the procedure. If we were to choose one of the usual k's for cross-validation (anything between 5 and 10 folds), it would result in a very low percentage of the final training set being MCI subjects. With this 3-fold division however, the 109 subjects constitute $\sim 25\%$ of the training set (the training set will now have 425 subjects, 109 of which are MCI).

This technique is not cross-validation in its standard form, so from now on we shall refer to the experiments where only non-MCI data is included in the training set as blind and to the experiments where a third of the MCI data is included in the training set as semi-blind.

For the Methods II and III, where SVM training is involved, cross validation has been performed in the selection of the SVM model. In this case, the training data (either just the non-MCI or the combination of non-MCI and $\frac{1}{3}$ of MCI data) is divided into k folds $(5, 7$ or $10)$. k SVM classifiers are trained and tested and k F1-scores are calculated. The SVM classifier with the highest F1-score is the one that gets used by the model and classifies the testing set (MCI data) for the final evaluation of the method. For this cross-validation we regard our training-set from a more common angle and divide it into the widely used numbers of folds.

Experiments In practice, different approaches to the methods have been attempted. All three methods are tested exactly as described in Section 2.4 (no MCI data used in the training process), as well as by means of cross-validation, including a portion of the MCI data in the training process. Cross-validation on SVM was performed in all cases where SVM was actually used. Additionally, because of the nature of the data, the HMMs produce not very stable results (more will be discussed in Section 4.2). For this reason all the different experiments have been performed ten times and their results averaged, in order for the results to be smoother and any outliers to be eliminated. This a common practice when performing evaluation of probabilistic models, so that the results are smoother and more indicative of the actual performance of the models.

3.2 Results

In this section the results of the experiments for all methods are summarized and presented. The following diagrams contain the metrics of the different methods being tested with and without the use of cross-validation on the training data and with $5, 7$ and 10-fold cross-validation on the SVM training (when applicable). Additional metrics for the subjects with the maximum number of follow-up scans (three follow-ups) are produced and shown.

3.2.1 Random Classifier

As mentioned in Section 1.4 the approach that we attempt in this thesis is not an extension of a previous method, hence we have no state-of-the-art results to compare with. We will therefore compare our results with a random classifier, which we will set as the lowest acceptable limit.

For our data, we can define the prior probabilities of the two classes (non-AD and AD) based on the numbers indicated in Table 2.2:

$$p_{\text{non-AD}} = \frac{\text{Number of CN and MCI diagnoses at Last Scan}}{\text{Number of all subjects}} = \frac{391}{631} = 0.62$$

$$p_{\text{AD}} = \frac{\text{Number of AD diagnoses at Last Scan}}{\text{Number of all subjects}} = \frac{240}{631} = 0.38$$

The random classifier, assigns a class to any data-point based on a predefined probability:

$$p_{\text{random}} = \begin{cases} p_{\text{class}}, & \text{where class} \in [\text{non-AD}, \text{AD}] \\ \frac{1}{2}, & \text{equal probability of assigning either class} \end{cases}$$

In the first case, the classifier will classify each data-point with a probability equal to the priors of the two classes, which results in maximum overall accuracy of the classification. In the second case, the classifier slightly favours the minor class, AD in our case. This approach increases the sensitivity of the system, which means maximising the AD detection, at the cost of decreasing the specificity (more AD cases will get caught at the cost of more false positive results).

Recall, which is the ratio of the data-points of a class that have been correctly classified, is p_{random} in both cases. In our experimental results, the recall of the AD class is the sensitivity and the recall of the non-AD class is the specificity. Therefore, in the first random classifier we would get:

$$\begin{aligned} \text{Sensitivity} &= 0.38 \\ \text{Specificity} &= 0.62 \\ \text{F1} &= 0.4712, \end{aligned}$$

while in the second random classifier we would get:

$$\text{Sensitivity} = \text{Specificity} = \text{F1} = 0.5.$$

The DOR would be 1 for both versions of the random classifier.

At this point and in the spirit of aiming for the best possible performance, we choose the highest values for Sensitivity and Specificity as our lower limits. We therefore set:

$$\begin{aligned}\text{Sensitivity}_{\text{min}} &= 0.5\\ \text{Specificity}_{\text{min}} &= 0.62\\ \text{F1}_{\text{min}} &= 0.554\end{aligned}$$

3.2.2 Method I

In Figures 3.1, 3.2, 3.3 & 3.4 we show the Sensitivity and Specificity measurements for increasing numbers of states, as well as the harmonic mean of the two (F1-score). What we can deduce from these graphs is that the increasing number of states does not increase the performance of the system. Though it doesn't significantly affect it we can see in Figures 3.1 & 3.3 that sensitivity tends to increase and specificity tends to equally decrease, while maintaining a steady F1-score. A common approach with these two metrics is to try to keep them at relative close numbers and as high as possible. It is of course desirable that they are both very high, but except for a perfect classifier, they tend to have an inverse behaviour. A very good classifier would correctly classify both positive and negative data-points with few, or even no false positives and false negatives. These false positives and false negatives decrease the specificity and sensitivity of the system. If a classifier over-classifies one class, e.g. classifies all data-points as positive, then the sensitivity of this classifier will be almost perfect, but at the same time its specificity will be extremely low. Thus, it is good practice to compromise on either metric, so that we can have a better overall performance.

For the blind experiment in Figure 3.1 we see that sensitivity is well above the acceptable limit, while specificity starts upon the limit and then decreases when more states are used. Similar behaviour occurs for the subjects with 3 follow-ups (Figure 3.3), but the metrics are boosted, thus the specificity follows a better path, staying above the limit for a larger number of states. Also the divergence of the two metrics starts occurring at a later state. In both cases, however the F1-score is much higher than the pre-set limit.

On the other hand the semi-blind experiment produces much more stable metric graphs (Figure 3.2). Sensitivity is still ranging high and specificity is stably upon or above the 0.62 limit, except for one occasion. The stability of the graphs concerns not only variations of each metric individually, but also the closeness of the two when compared. Similarly the subjects with 3 follow-ups exhibit a corresponding behaviour. The metrics are higher, with specificity exhibiting the best improvement, as it stays well above its limit (Figure 3.4).

In Figure 3.5 we can see the progression of the diagnostic odds ratio (DOR) for both the blind and the semi-blind experiment and using the entire MCI group or only subjects with 3 follow-ups. Here we see satisfactory results, all ratios remain well above 2 (when 1 is our limit), with small variation, while the best DOR performance is produced for the blind experiment and using 3 follow-ups.

This leads us to the conclusion that for this method using MCI subjects in the training process, contributes in the stability of the system, but does not improve the classification rates.

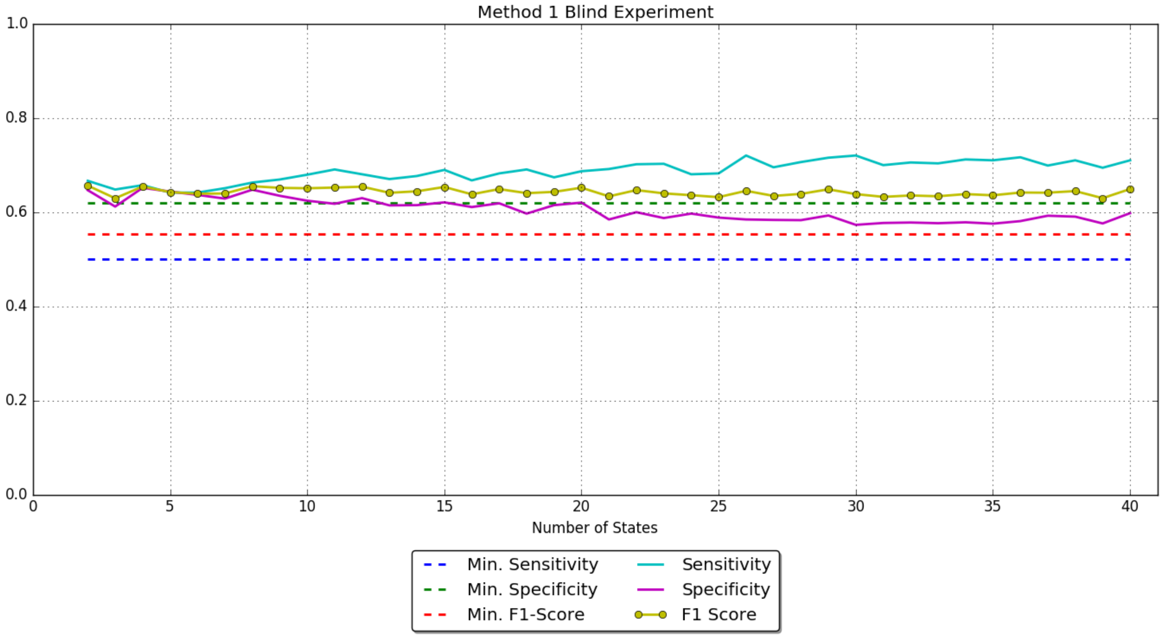


Figure 3.1: Sensitivity and Specificity of Method I without Cross-Validation for an increasing number of HMM states. Average F1 Score = 0.64273

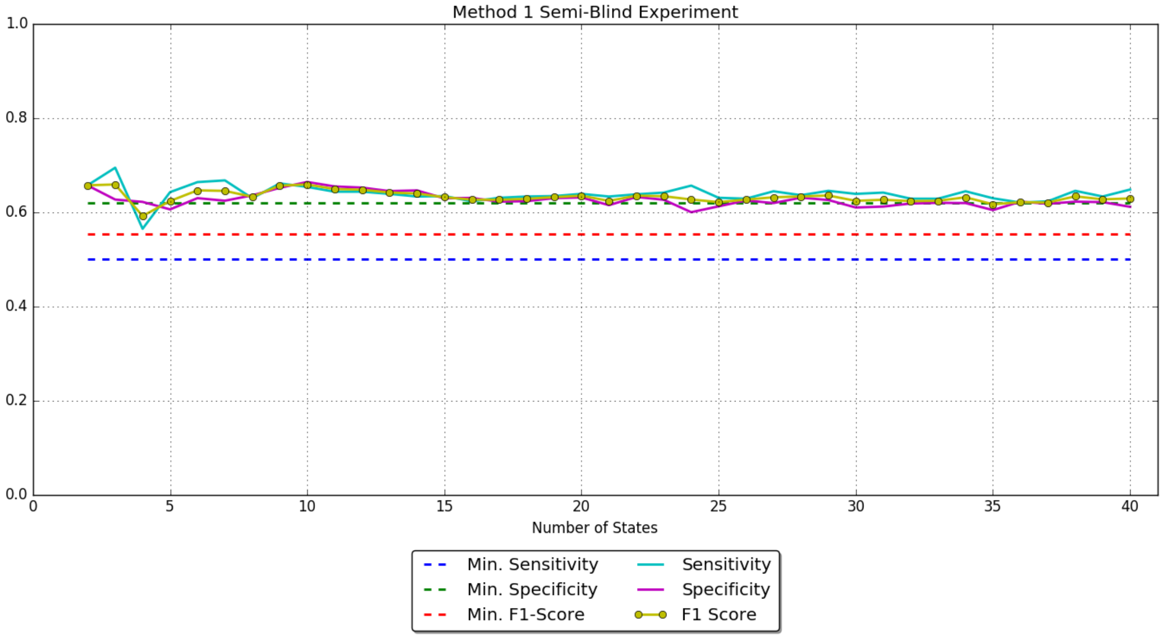


Figure 3.2: Sensitivity and Specificity of Method I with Cross-Validation for an increasing number of HMM states. Average F1 Score = 0.63277

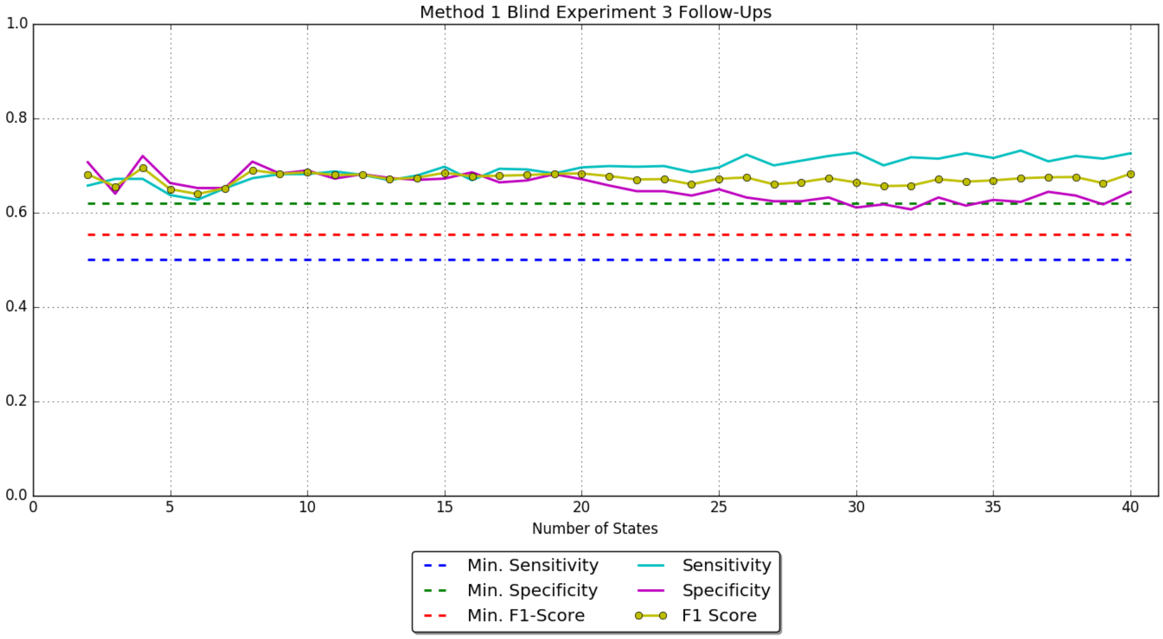


Figure 3.3: Sensitivity and Specificity of Method I for subjects with 3 Follow-ups without Cross-Validation for an increasing number of HMM states. Average F1 Score = 0.671514

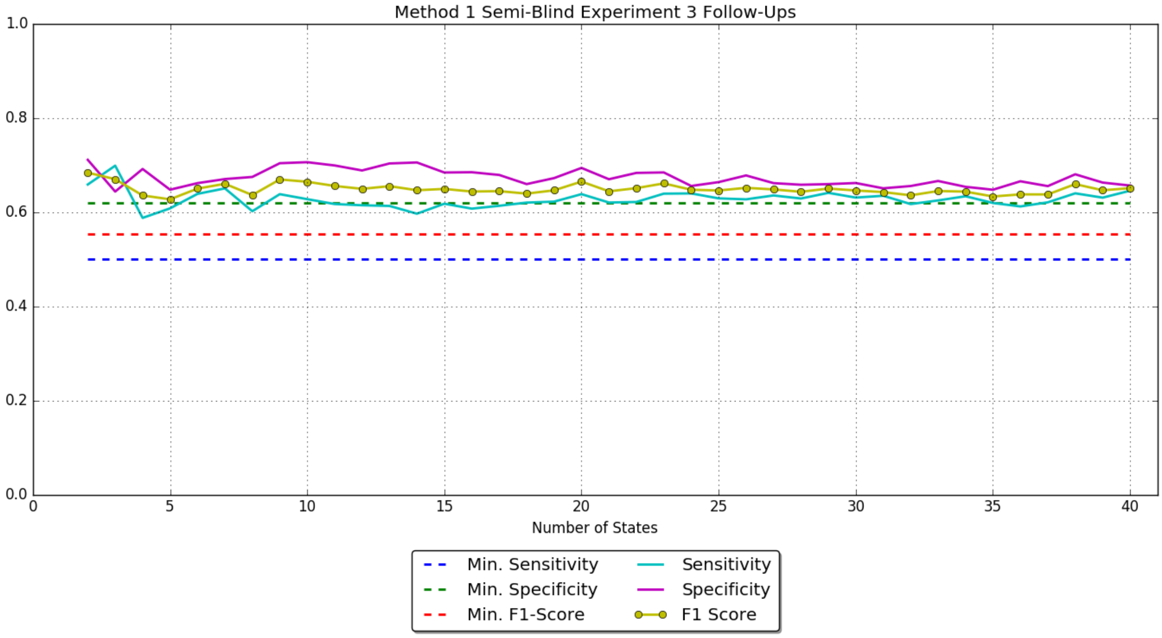


Figure 3.4: Sensitivity and Specificity of Method I for subjects with 3 Follow-ups with Cross-Validation for an increasing number of HMM states. Average F1 Score = 0.649184

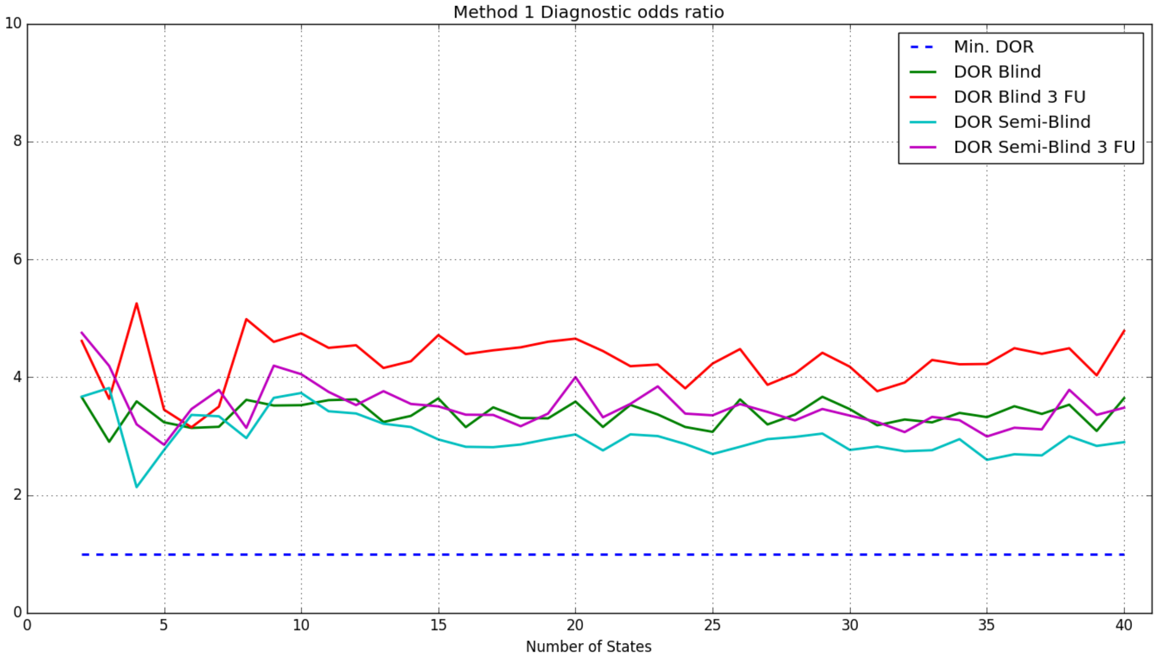


Figure 3.5: Diagnostic Odds Ratio for the different approaches of Method I

3.2.3 Method II

For the second method, when examining Figures 3.6, 3.7, 3.8 & 3.9, the graphs look much more interesting. We can see that the second method gives very high Specificity (almost 1 for subjects with 3 follow-ups), but very low Sensitivity, which consequently causes a very low F1 score. This phenomenon of inverse sensitivity and specificity behaviour was discussed in the previous section. The SVM cross-validation does not seem to have a significant impact on the performance.

After further examining the results and studying the confusion matrices that have been produced (Figures 3.13a & 3.13b), we saw that the classifier classified most data as non-AD, which is the cause of the very high Specificity/low Sensitivity. As mentioned in Section 2.4.3 the state sequences that are produced by the HMMs and used as feature vectors for the SVM are highly variant which turns them into non-separable data. This causes the SVM to fail in finding a separating hyperplane.

When examining the DORs (Figures 3.10, 3.11 & 3.12) of this method we can once again deduce that the cross-validation for SVM has not affected the behaviour of the system, since all DOR graphs are almost identical for the different numbers of folds. In general they are quite low, significantly lower than for Method I, and for the blind experiments, DORs fall as low as below 1.

For this method the use of MCI subjects in the training process slightly improves the performance, in terms of the Sensitivity/Specificity divergence, which becomes smaller, but the results are still disappointing.

Figures 3.13a & 3.13b show two examples of confusion matrices for different runs of the experiments where we can see the amount of data-points classified in each class. As mentioned in Section 3.1 all experiments are ran 10 times for avoidance of outliers, hence the existence of decimal numbers in the confusion matrix. They are effectively showing the *average* amount of data-points classified in each class over the performance of 10 experiments with identical parameters.

These very extreme results in sensitivity and specificity have motivated us to develop Method III.

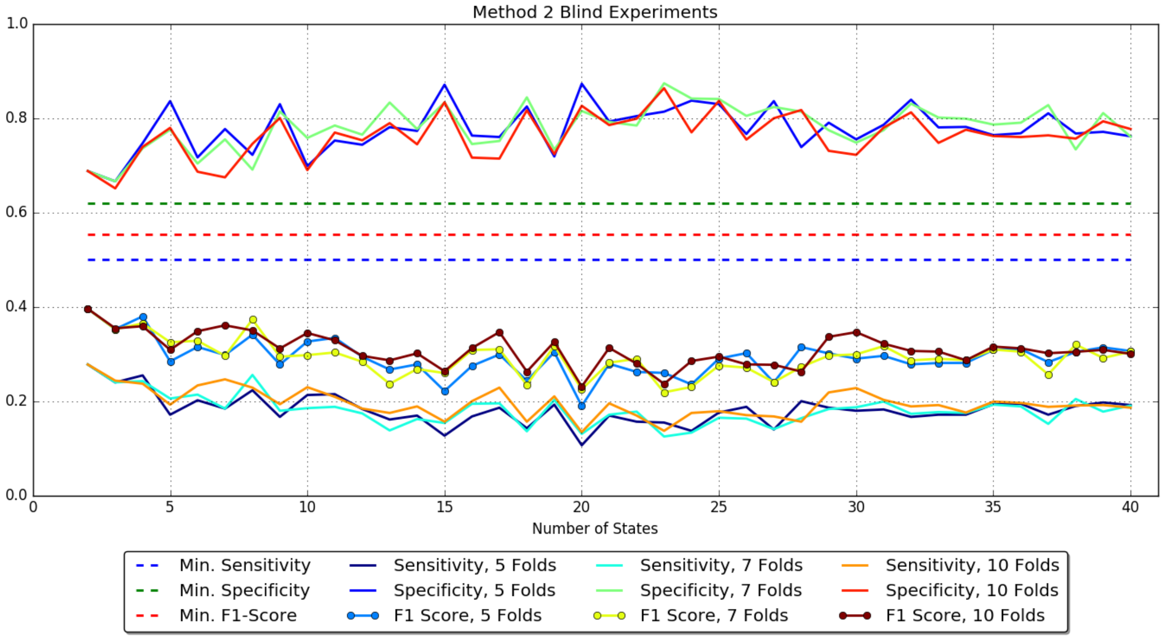


Figure 3.6: Sensitivity and Specificity of Method II without Cross-Validation for an increasing number of HMM states and Different SVM Folds. Average F1 Score for 5, 7 & 10 Folds: $0.292659, 0.292794$ & 0.30927

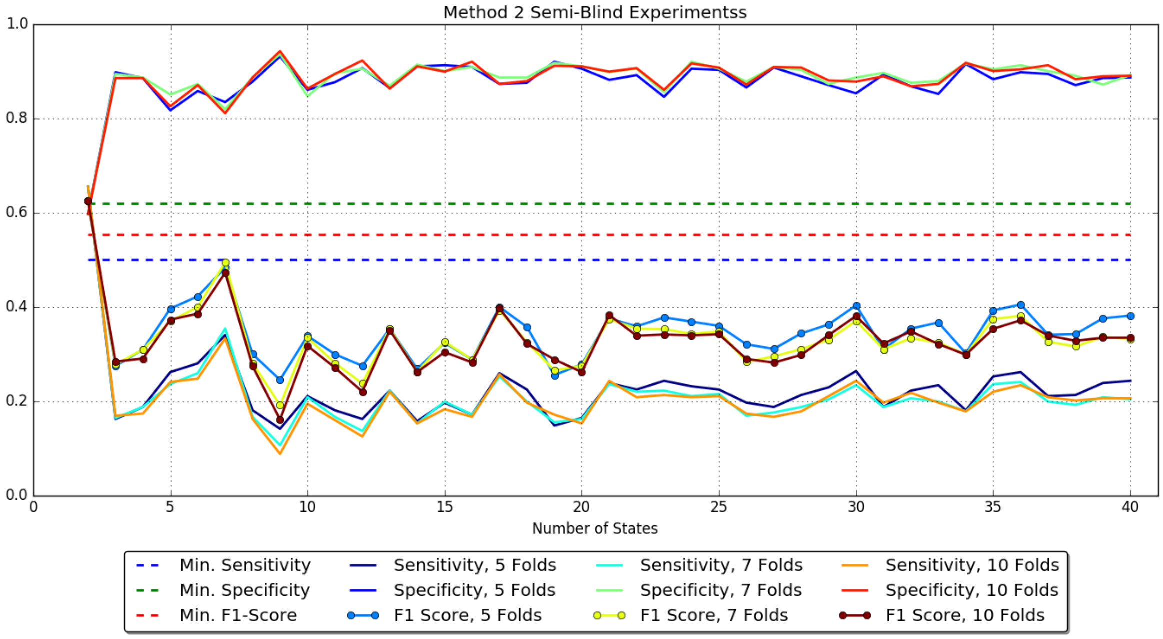


Figure 3.7: Sensitivity and Specificity of Method II with Cross-Validation for an increasing number of HMM states and Different SVM Folds. Average F1 Score for 5, 7 & 10 Folds: $0.34986, 0.332871$ & 0.328869

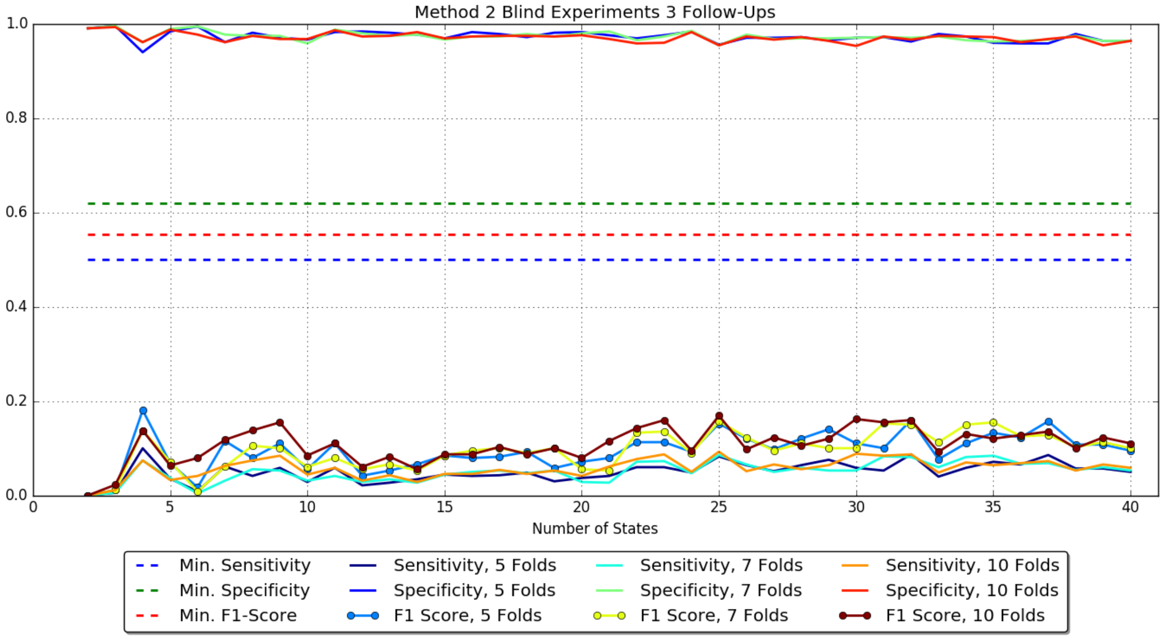


Figure 3.8: Sensitivity and Specificity of Method II for subjects with 3 Follow-ups without Cross-Validation for an increasing number of HMM states and Different SVM Folds. Average F1 Score for 5, 7 & 10 Folds: $0.094648, 0.095525$ & 0.1077

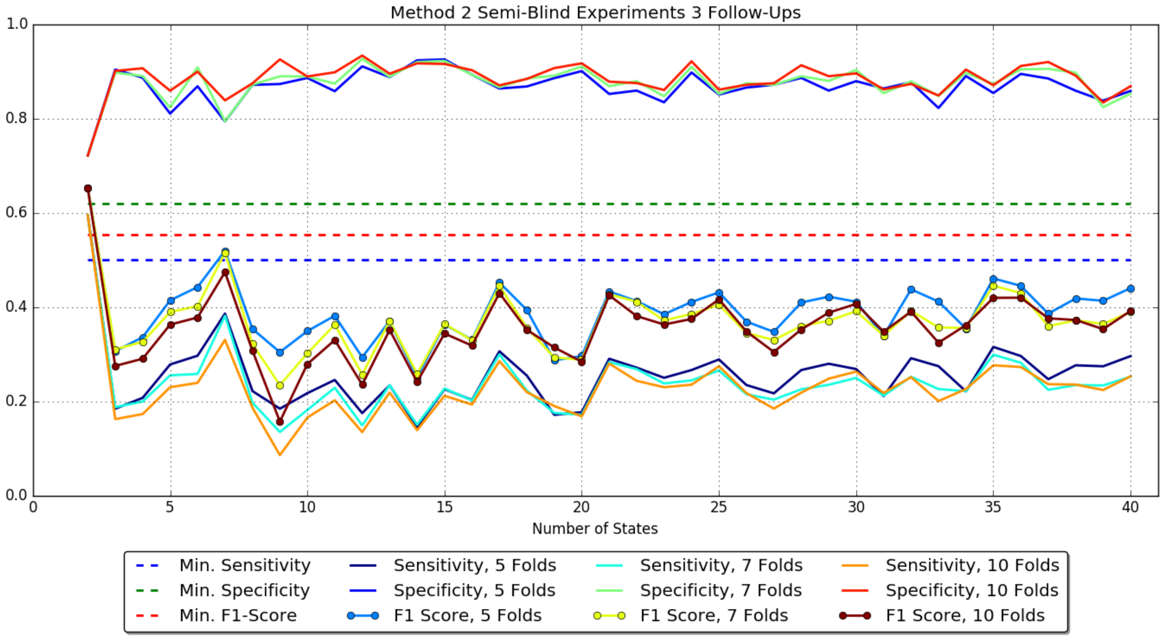


Figure 3.9: Sensitivity and Specificity of Method II for subjects with 3 Follow-ups with Cross-Validation for an increasing number of HMM states and Different SVM Folds. Average F1 Score for 5, 7 & 10 Folds: $0.39075, 0.368816$ & 0.35632

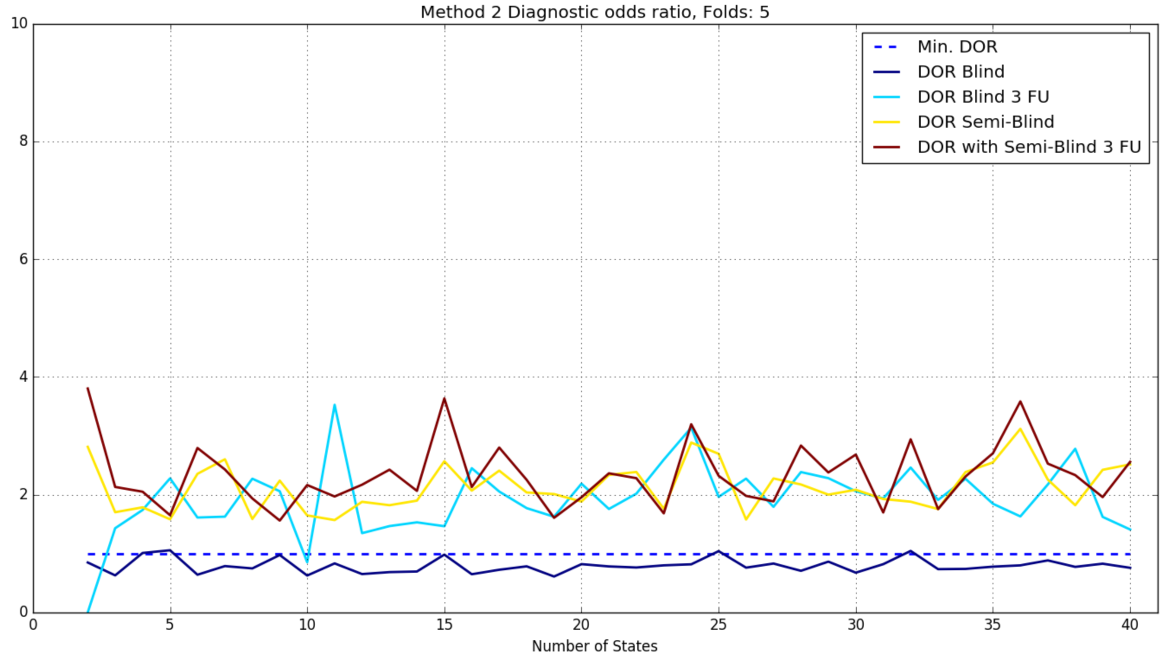


Figure 3.10: Diagnostic Odds Ratio for the different approaches of Method II with 5-Fold Cross Validation

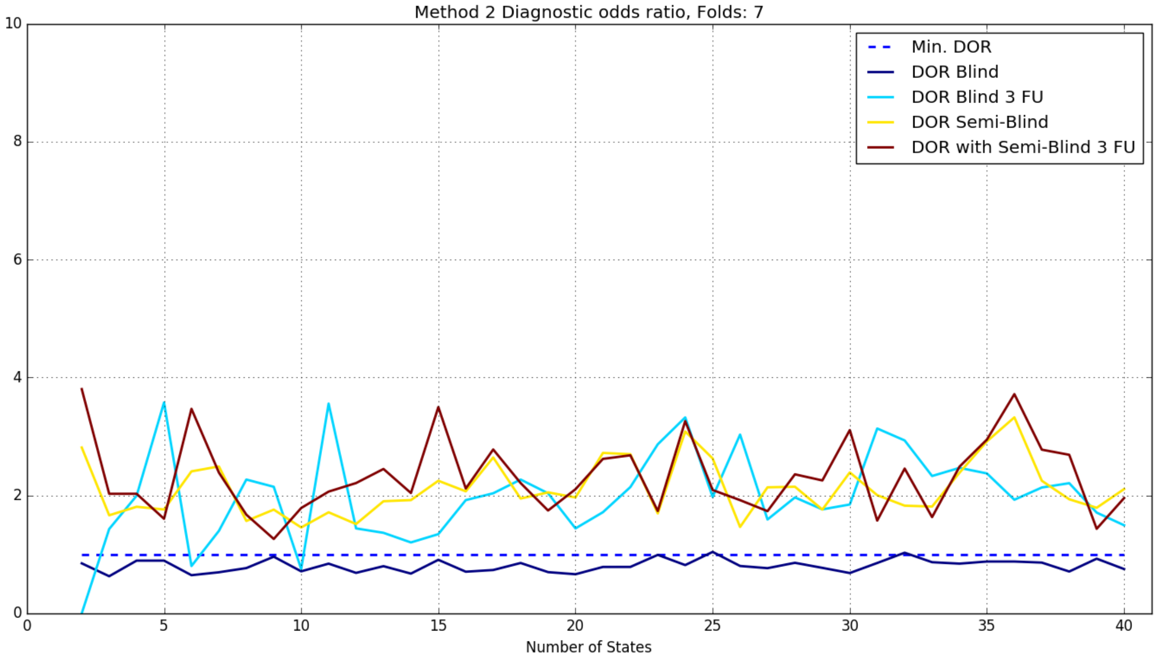


Figure 3.11: Diagnostic Odds Ratio for the different approaches of Method II with 7-Fold Cross Validation

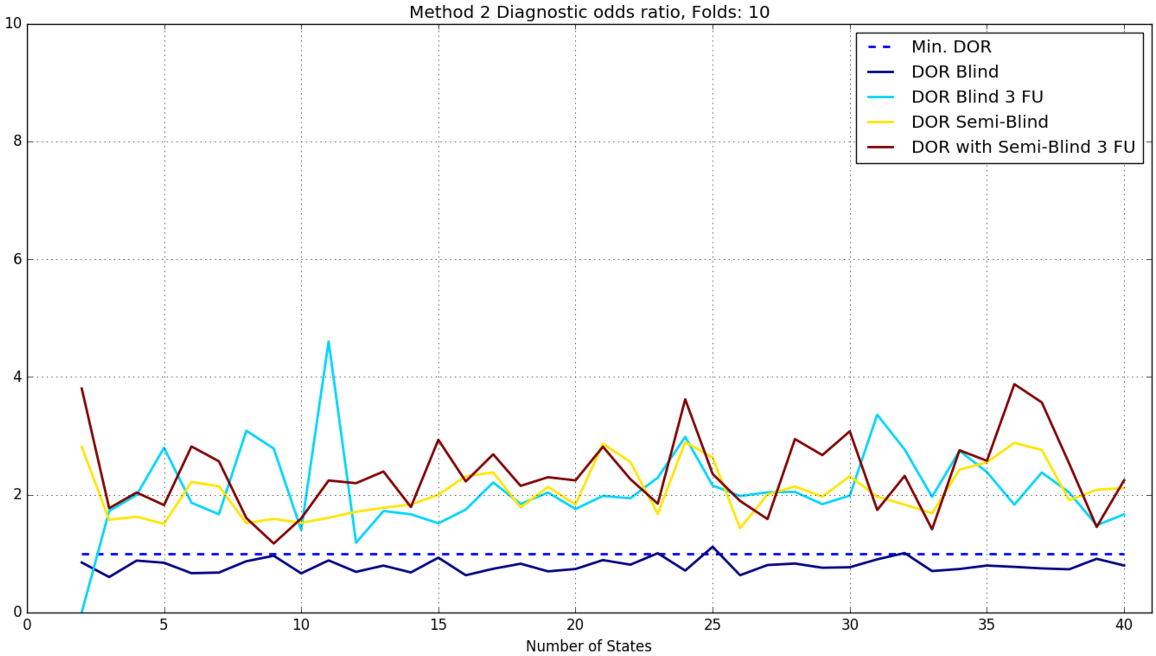
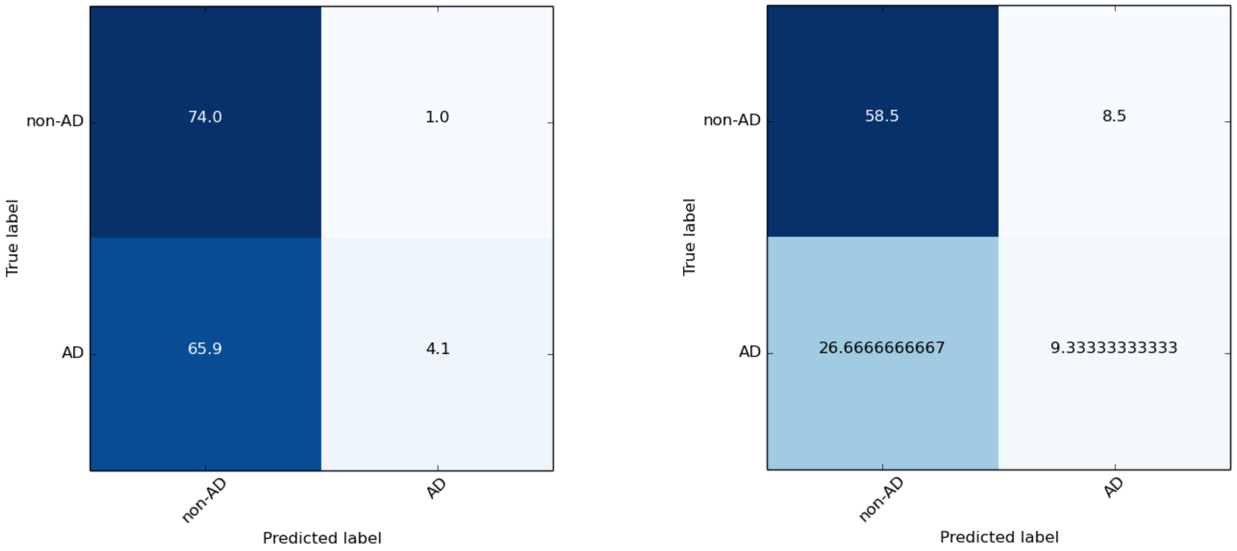


Figure 3.12: Diagnostic Odds Ratio for the different approaches of Method II with 10-Fold Cross Validation



(a) 11 States without Cross-Validation, 10 SVM Folds & Subjects with 3 Follow-Ups

(b) 17 States with Cross-Validation, 5 SVM Folds & with all MCI Subjects

Figure 3.13: Confusion Matrices for Method II

3.2.4 Method III

For this method we see that semi-blind experiments are significantly affected by the extra MCI subjects and, as in Method I, the divergence of Sensitivity and Specificity is also improved. Here we can observe that not only is there no divergence occurring at all, but there is actually convergence (Figures 3.14, 3.15 & 3.17). In all three Figures we see that the two metrics meet at a certain point and then continue very close to each other. In Figure 3.16 sensitivity and specificity are evolving closely together throughout the entire experiment.

Like with the previous two methods, we present the Sensitivity/Specificity graphs, as well as the DOR graphs. Here we can see that Sensitivity remains above the predefined threshold, while Specificity has trouble during the blind experiment, where it reaches and stays upon the lowest threshold. We can observe a tremendous improvement compared to the *second* method indicating that the use of the frequency maps instead of the actual state sequences constitutes the data much more separable and still is able to carry significant information about the progression of the condition.

The semi-blind experiments do indeed help with the Sensitivity/Specificity divergence, though we can also observe in Figures 3.15 & 3.17 that a slight decline of both metrics is caused, as well as the F1 Score.

Regarding the DOR graphs in Figures 3.18-3.20, it is obvious that they are greatly improved since the previous method with values being way higher than 3.0. These Figures showcase a decline similar to the one in the Sensitivity/Specificity graphs when MCI subjects are included in the training.

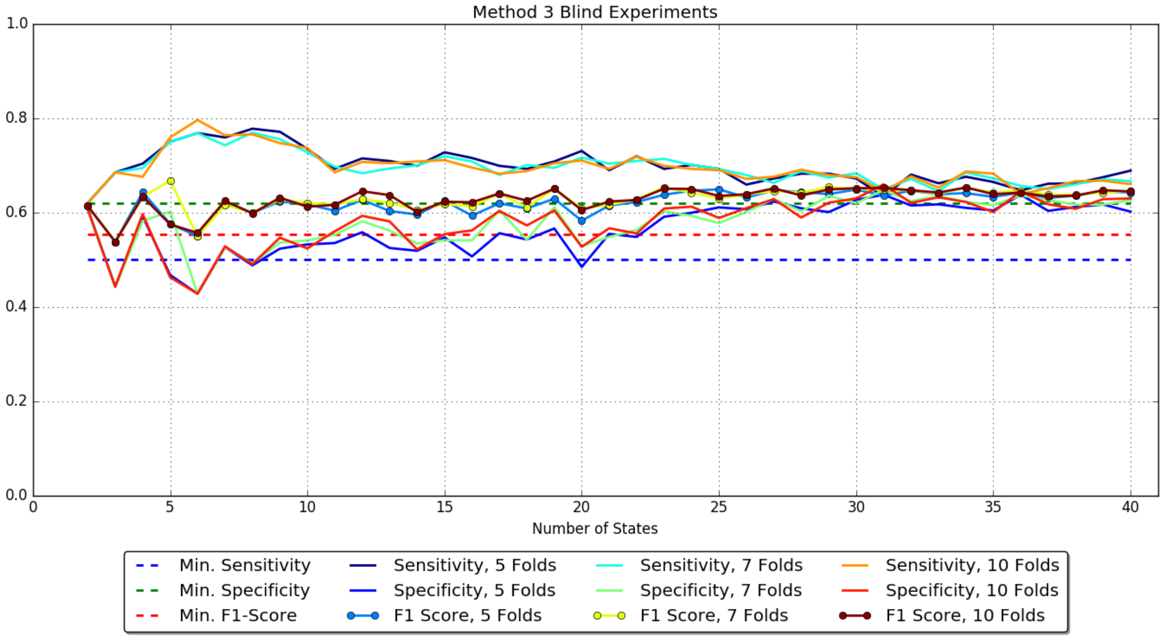


Figure 3.14: Sensitivity and Specificity of Method III without Cross-Validation for an increasing number of HMM states and Different SVM Folds. Average F1 Score for 5, 7 & 10 Folds: $0.621645, 0.628742$ & 0.628219

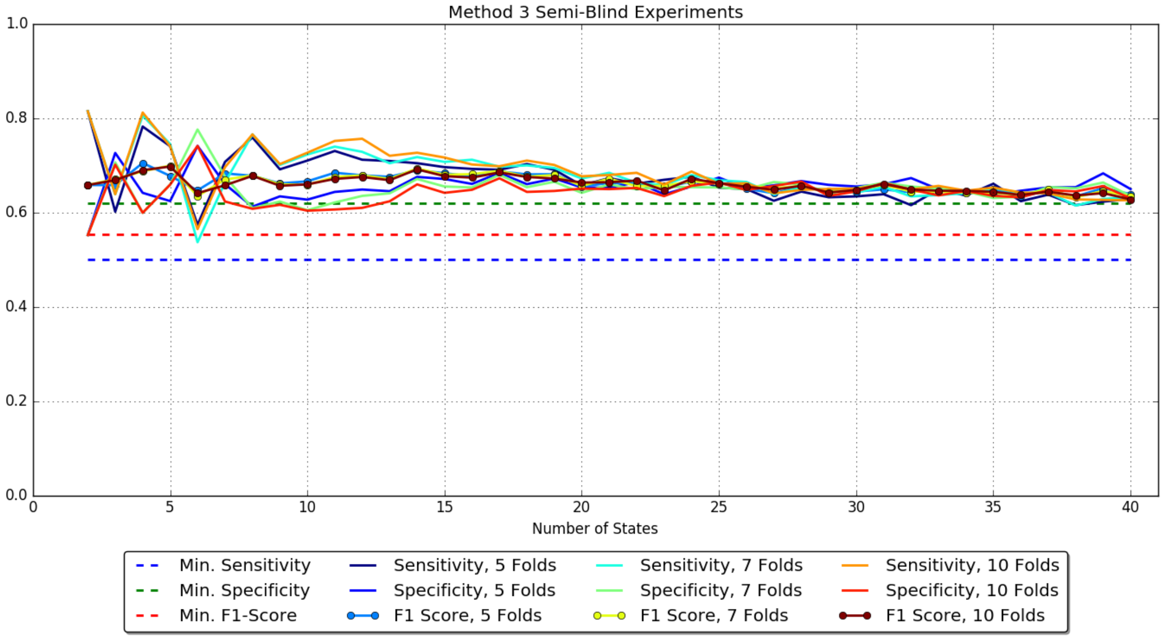


Figure 3.15: Sensitivity and Specificity of Method III with Cross-Validation for an increasing number of HMM states and Different SVM Folds. Average F1 Score for 5, 7 & 10 Folds: $0.661823, 0.661478$ & 0.660475

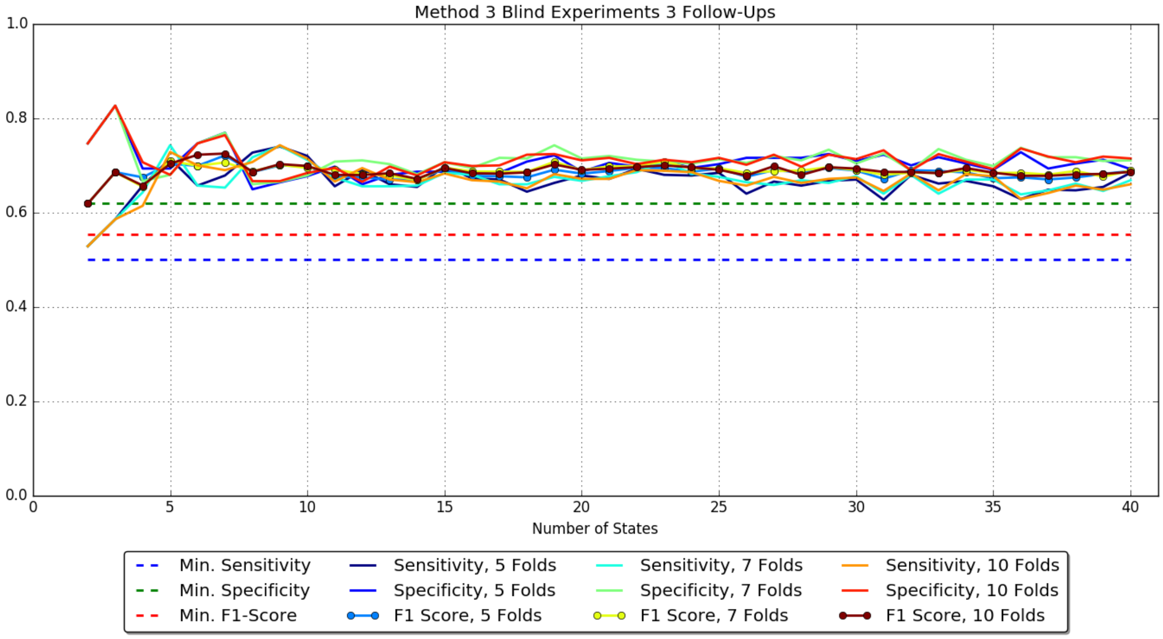


Figure 3.16: Sensitivity and Specificity of Method III for subjects with 3 Follow-ups without Cross-Validation for an increasing number of HMM states and Different SVM Folds. Average F1 Score for 5, 7 & 10 Folds: $0.683631, 0.6875$ & 0.687556

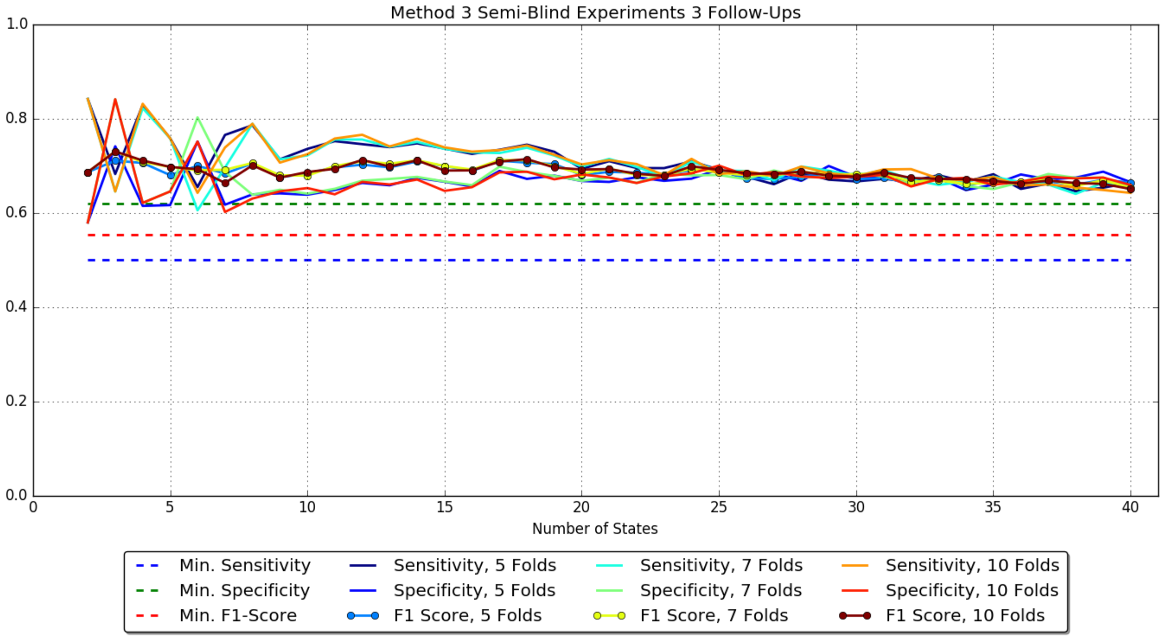


Figure 3.17: Sensitivity and Specificity of Method III for subjects with 3 Follow-ups with Cross-Validation for an increasing number of HMM states and Different SVM Folds. Average F1 Score for 5, 7 & 10 Folds: $0.68485, 0.687042$ & 0.686773

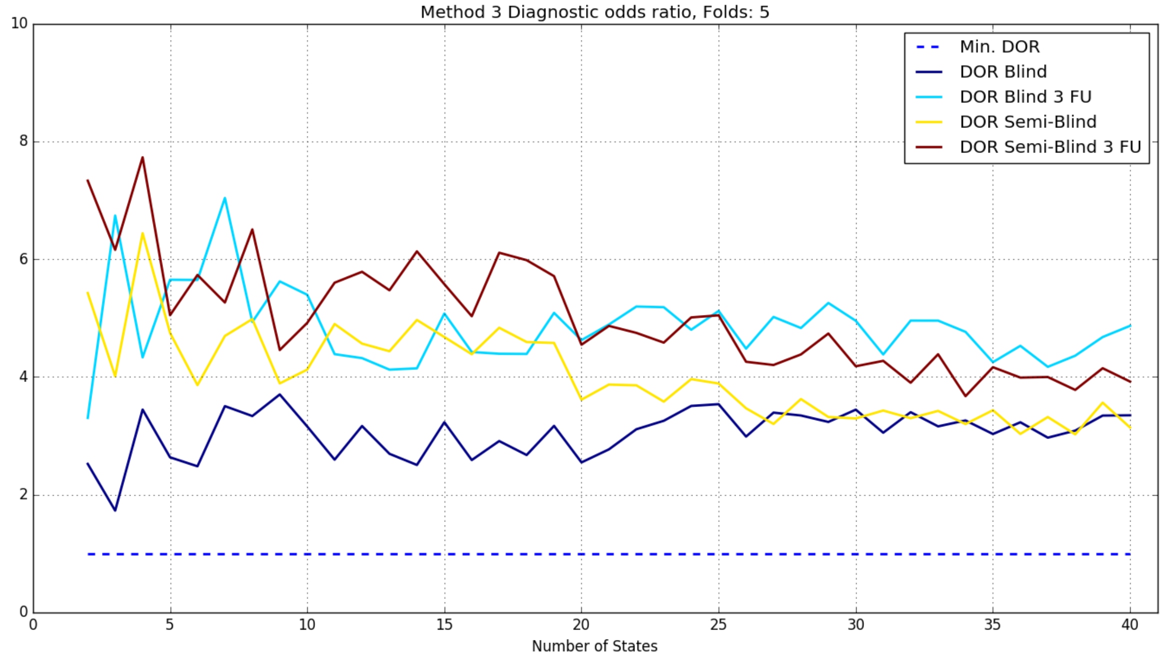


Figure 3.18: Diagnostic Odds Ratio for the different approaches of Method III with 5-Fold Cross Validation

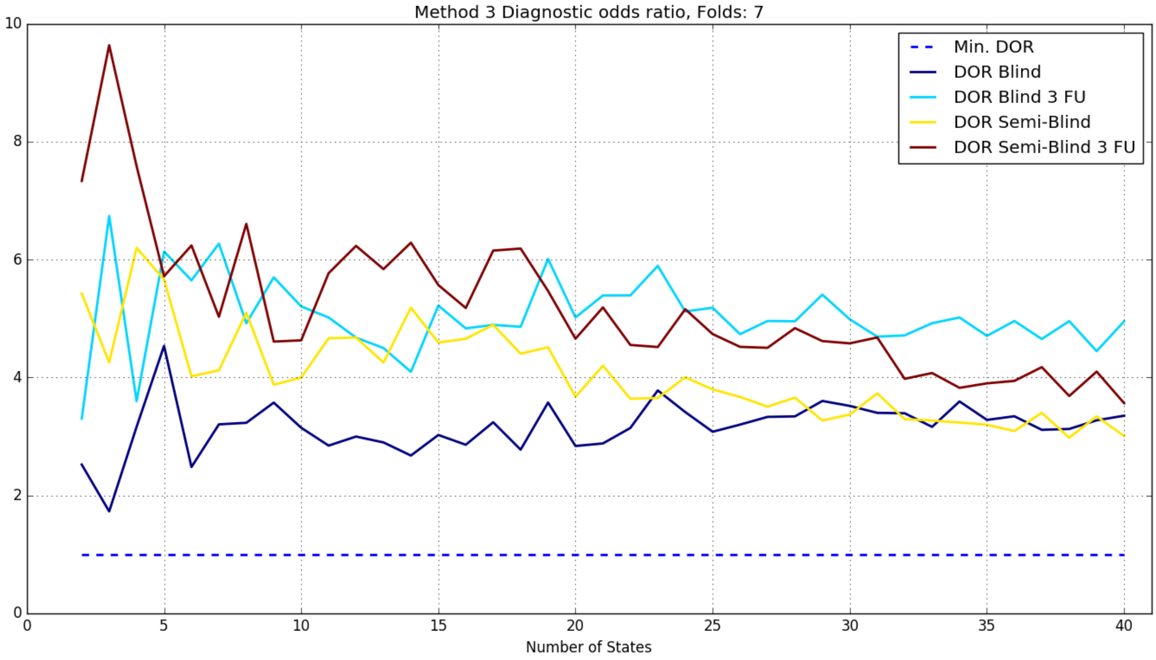


Figure 3.19: Diagnostic Odds Ratio for the different approaches of Method III with 7-Fold Cross Validation

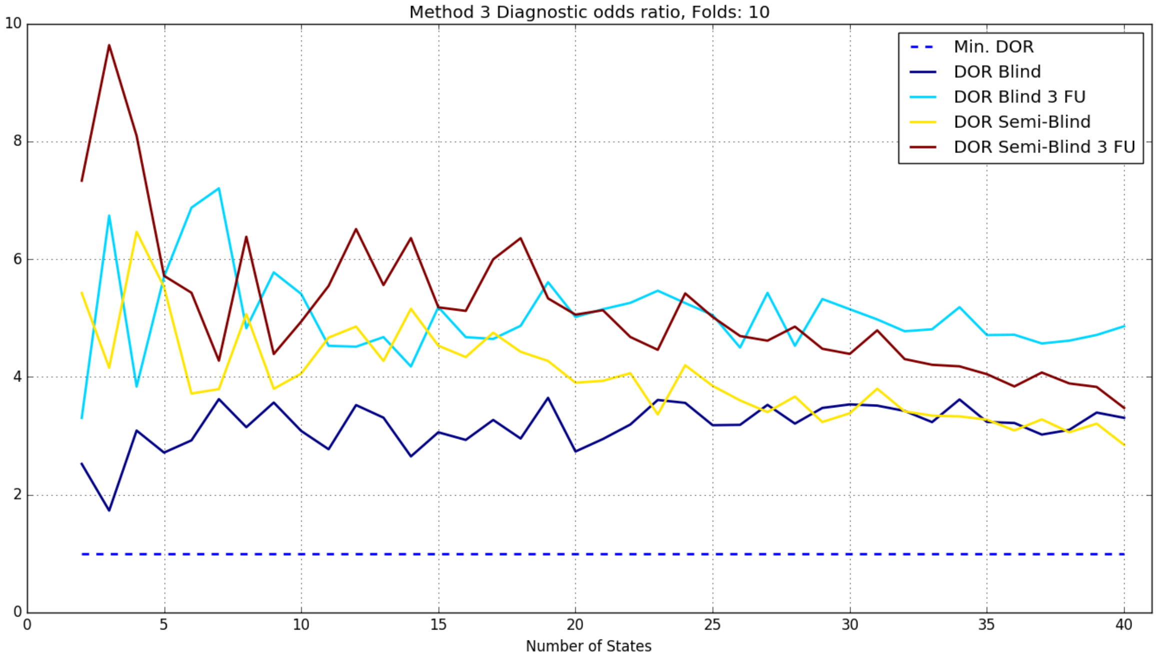


Figure 3.20: Diagnostic Odds Ratio for the different approaches of Method III with 10-Fold Cross Validation

3.2.5 Results' Summary

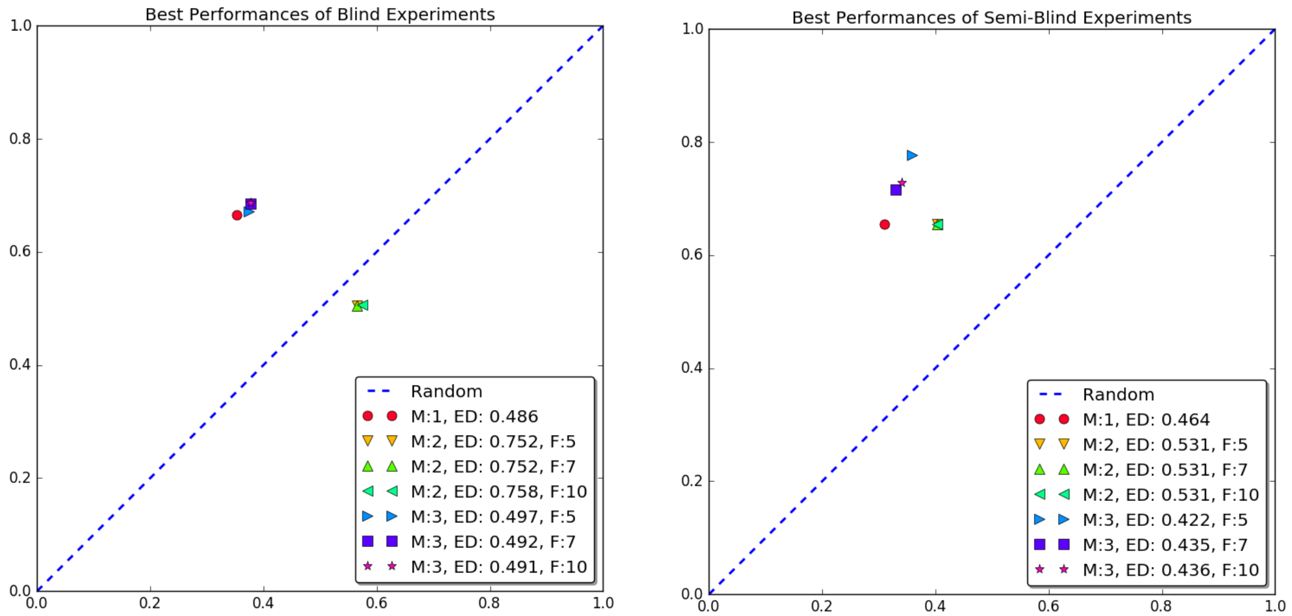
Up until now, the results and the evaluation that has been done was on a more theoretical level and concerned the methods as an overall scientific search (Sensitivity/Specificity and DOR graphs produced for all numbers of states). However, on a more practical level, if these methods were to be used, then we would be more interested in specific instances of the classifiers that have been trained and tested. Meaning that instead of comparing the results produced by varying numbers of states for the HMMs (our main variable), we are going to compare the best "run" of each method.

For this we have produced Figures 3.21 & 3.22, where we visualise an ROC space with the data points of these instances of each method. The different data-points in these Figures correspond to the true positive rate and 1−true negative rate of each method's best instance. In the legend the different methods are indicated by the letter M. We also provide the Euclidean distance of each point (ED) from the top left corner (the shorter the distance, the better the performance). These numbers are also summarized in Tables 3.1 & 3.2. It is easier to compare and contrast the different methods and approaches here, where the performance of all is visualised.

In the Figures we can see that the MCI subjects in the training set improve all the peak points of the tested methods, with Method II exhibiting the most impressive improvement, as it reaches in performance the other two, even though it initially performs very badly (below the random classifier in Figure 3.21a). We also see that Methods I and III performing almost equally good, with Method III slightly exceeding, especially for the subjects with 3 follow-ups.

Another important conclusion that can be drawn from these Figures, but is exhibited even more intensely in the separate results of the methods, is the importance of long MRI sequences. In all provided Figures we can always observe the the results for the subjects with 3 follow-up scans are are higher than the overall. Meaning that among these subjects the rate of detecting progression to AD or progression to MCI or even conversion to CN is higher. This stresses the need for as many follow-ups as possible for each individual so that a more certain prediction can occur.

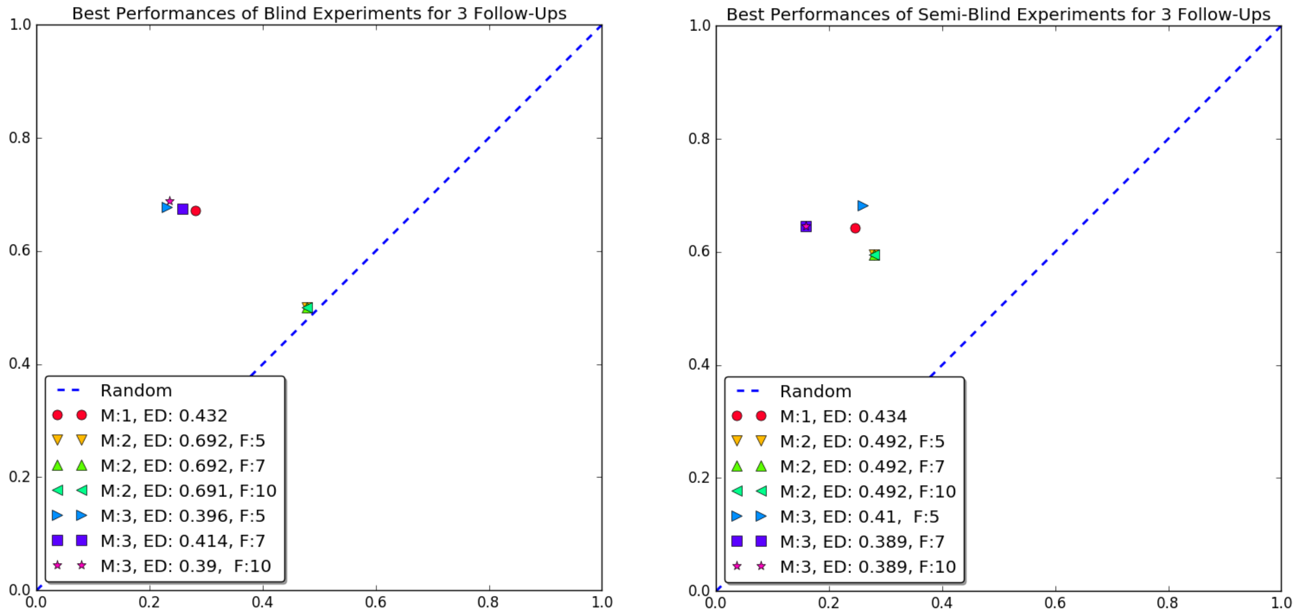
We finally see that the use of different numbers of folds for the training of the SVM has very small impact on the performance, something that was also commented earlier.



(a) All Methods Without Cross-Validation

(b) All Methods With Cross-Validation

Figure 3.21: Best Performances of all Methods



(a) All Methods Without Cross-Validation, Subjects with 3 Follow-ups

(b) All Methods With Cross-Validation, Subjects with 3 Follow-ups

Figure 3.22: Best Performances of all Methods, Subjects with 3 Follow-ups

An interesting note on the results of Tables 3.1 and 3.2 is that, while the Sensitivity and Specificity metrics improve for subjects with 3 follow-ups, they also inverse when it comes to which is higher. So, our system exhibits higher Sensitivity when classifying all subjects, but higher Specificity on the subjects with longer MRI sequences. This can be justified by the unequal distribution of the diagnoses for different numbers of follow-up scans.

Table 3.1: Summary of the best results of each method with respect to its closest point to the upper left corner of the ROC space. The table shows the Specificity and Sensitivity that correspond to that point, as well as its Euclidean distance from the upper left corner. The column *type* corresponds to the type of experiment ran regarding the trainind data (**B** for Blind and **S-B** for Semi-Blind). As a comparison, we also give the average F1 Scores of each method, as well as the average Diagnostic odds ratios (different SVM fold results are provided when possible).

METH.	**TYPE**	**SPECIFICITY**	**SENSITIVITY**	**DISTANCE**	**Avg. F1**	**Avg. DOR**
		5, 7, 10 Folds	5, 7, 10 Folds	5, 7, 10 Folds	5, 7, 10 Folds	5, 7, 10 Folds
I	B	0.646	0.665	0.486	0.643	3.381
I	S-B	0.689	0.655	0.463	0.633	2.998
II	B	0.434, 0.434, 0.425	0.504, 0.504, 0.505	0.752, 0.752, 0.757	0.293, 0.293, 0.31	0.794, 0.809, 0.797
II	S-B	0.596, 0.597, 0.597	0.655, 0.654, 0.654	0.530, 0.530, 0.530	0.35, 0.333, 0.329	2.135, 2.123, 2.066
III	B	0.626, 0.622, 0.622	0.672, 0.685, 0.686	0.496, 0.491, 0.490	0.622, 0.629, 0.628	3.054, 3.172, 3.173
III	S-B	0.641, 0.67, 0.659	0.776, 0.715, 0.728	0.422, 0.435, 0.435	0.662, 0.661, 0.66	4.018, 4.04, 4.015

Table 3.2: Subjects with 3 Follow-ups. Summary of the best results of each method with respect to its closest point to the upper left corner of the ROC space. The table shows the Specificity and Sensitivity that correspond to that point, as well as its Euclidean distance from the upper left corner. The column *type* corresponds to the type of experiment ran regarding the trainind data (**B** for Blind and **S-B** for Semi-Blind). As a comparison, we also give the average F1 Scores of each method, as well as the average Diagnostic odds ratios (different SVM fold results are provided when possible).

METH.	**TYPE**	**SPECIFICITY**	**SENSITIVITY**	**DISTANCE**	**Avg. F1**	**Avg. DOR**
		5, 7, 10 Folds	5, 7, 10 Folds	5, 7, 10 Folds	5, 7, 10 Folds	5, 7, 10 Folds
I	B	0.719	0.671	0.431	0.672	4.289
I	S-B	0.753	0.642	0.434	0.649	3.495
II	B	0.521, 0.521, 0.522	0.5, 0.5, 0.5	0.691, 0.691, 0.691	0.095, 0.096, 0.108	1.939, 1.998, 2.089
II	S-B	0.72, 0.72, 0.72	0.595, 0.595, 0.595	0.491, 0.491, 0.491	0.391, 0.369, 0.356	2.347, 2.326, 2.357
III	B	0.769, 0.742, 0.764	0.677, 0.675, 0.688	0.395, 0.414, 0.39	0.684, 0.688, 0.688	4.855, 5.045, 5.043
III	S-B	0.74, 0.84, 0.84	0.682, 0.645, 0.645	0.410, 0.388, 0.388	0.685, 0.687, 0.687	5.01, 5.177, 5.171

Chapter 4

Conclusions & Future Work

In this chapter we summarize our work and comment on the results. We discuss any issues that come up and affect the performance of our system, while discussing about work that can be done to extend and improve our system and also address the issues that came up.

4.1 Conclusions

The objective of the thesis was to implement a model that would be able to predict the progression of the condition of MCI patients by examining their longitudinal MRI scans and at the same time study the information that can be retrieved by these scans without using any other means of diagnosis (cognition tests etc.). For this we want to create a model that is able to predict whether patients that are diagnosed with MCI will progress to AD or not.

We developed three methods, all of which use HMMs as their basis. Each method is considered an extension of the previous:

M. I The first method trains two HMMs with observations of subjects progressing to AD and non-AD, thus guiding each HMM to specialise on the corresponding progression. The HMMs are then used as a means for the classification. We feed unseen observations to both HMMs and produce the probability of the observation having been generated by each HMM.

M. II The second method trains one HMM with observations of subjects progressing to both AD and non-AD, thus enforcing the HMM to retrieve more generalized structure regarding the disease's progression. The HMM then produce state sequences of all the observations. These state sequences are used as features for an SVM classifier which is trained on separating AD and non-AD subjects.

M. III The third method again trains one HMM with observations of subjects progressing to both AD and non-AD, thus enforcing the HMM to retrieve more generalized structure regarding the disease's progression. We once more produce state sequences of all the observations. These state sequences are converted into frequency maps, i.e. matrices of size $N \times N$ (N being the number of the HMM's states), where each element a_{ij} is a counter of the transitions occurring from state i to state j. These matrices are vectorised and used as feature vectors for the SVM classifier, like in Method II.

By examining the results of the experiments we can conclude that Methods I and III produce satisfactory models achieving Sensitivity which is much higher than the predefined threshold and achieving Specificity that manages to surpass the even stricter predefined specificity-threshold, while Method II seems to be problematic (Tables 3.1 & 3.2). Method III seems to be performing best, though Method I's results are quite close.

If the best classifier would be used (Method III, Semi-Blind, 5-fold SVM training), we would be able to detect 77.6% of the subjects that progress to AD and 64.1% of the subjects that either remain stable with MCI or convert back to CN.

Furthermore the results show, especially Method I's results, that there is significant structural information found in the MRI scans concerning the progression of the cognitive state of a patient. This is highly indicated in the first Method, where only the structural and temporal information of the scans is used for the classification by means of the HMMs and no other strong classifier is used.

Another important aspect that emerged through the experiments is the significance of the length of the longitudinal sequences. The results show that among the longest sequences (3 follow-ups) the systems performed much better than with shorter ones. This stresses the importance of having large numbers of follow-up scans.

4.2 Issues & Future Work

At this point, it is imperative to discuss the difficulties that the dataset poses on completing the objective. One important aspect is the size of it. In computer science it is always important to have very large datasets on which we can train our models, however that is difficult to achieve for medical data for a variety of reasons. In the case of our datasets, one issue is that the MRI scan itself is costly and not easy to obtain, therefore it is not easy to create a large dataset. Additionally, the process of segmenting the scans in order to obtain the volumetric features that we need, performed by Freesurfer, is very time-consuming. Each scan needs $12 - 14$ hours of preprocessing before producing our features, making it also hard to enlarge our dataset.

Another factor hindering the acquisition of MRI scans is the patients themselves. As mentioned before, the MRI scans are costly, therefore a patient may not be able to cover for the expenses of getting MRI scans frequently and for a long time, unless there is a means of financing that supports the creation of the scans (e.g. financing for a research project). Additionally, a patient may not feel comfortable with the whole procedure of MRI scans (many patients can feel claustrophobic during the procedure). Moreover, since our research focus on AD, a severe condition, it is also possible that patients suffering from the disease, may not be able to follow with their scans (either because of the severity of their disease, or even because of death).

However, the length of the sequences is very important, because the HMM works on a probabilistic basis, which means that it needs to observe large quantities of data in order for it to be able to stabilize its behaviour, and in this particular case the quantity of the data refers to the number of observations for each subject, rather than the number of the subjects. It is highly possible that with long enough observation sequences, the first method would outperform both of the other two and could produce much better results.

One more issue, that consequently affects the integrity of the obtained data, is the fact that a *definite* AD diagnosis can only occur post-mortem with a brain biopsy. There have been occasions, where the patient's diagnosis was proven incorrect by a post-mortem examination, meaning that a diagnosis in vivo cannot be absolutely trusted, at least with the diagnostic means available today. When it comes to our dataset, our diagnosis labels are not provided with information regarding the certainty of the diagnosis. This means that we can't be sure that the diagnosis is *correct*. Additionally we have no information regarding the moment of the diagnosis. If we knew that the diagnosis was taken in vivo, we could use that fact to improve our system as best as possible, by setting an "uncertainty" flag, which we could use to set extra weight for diagnoses performed post-mortem, for which we are sure of their correctness. However, this brings about a type of "chicken & egg" problem: in order to get better data, we need to find a better way diagnosing AD in vivo with certainty, but in order to achieve that we need better data.

There are different ideas and ways to attempt, in order to improve the performance achieved by this thesis, as well as, ways to address the issues discussed in the previous paragraphs, or at least soften their impact.

A speed-up modification can be attempted on the Freesurfer package, potentially by parallelizing its procedure, so that the scan processing time can be significantly reduced. This can provide researchers with more and longer longitudinal sequences.

A small change in the dataset can also help with the uncertainty over the diagnoses. A flag can be provided with each scan indicating whether it bears a verified diagnosis or not. Using this flag the model can be adjusted by putting weight factors on the data, and weighing more the subjects with definite diagnoses, or even implementing a semi-supervised method in order to take advantage of more data, even unlabelled.

Additional pre-processing can be applied on the 55-dimensional feature vectors, in the form of clustering or any dimension reducing technique, so that more compact and representative features can be used for the prediction process.

Finally, in the field of neural networks, there are many exciting ways to address the thesis' objective, for example with the use of Recurrent Neural Networks (RNN), that have the ability to model temporal and sequential data, without the demand of fixed length for the input. The RNNs are currently used in certain applications of speech recognition [14]. Deep neural networks (DeepNets) and convolutional neural networks (ConvNets) are also a very interesting approach. DeepNets and ConvNets have the advantage, because of their complexity and size, that they are able to model more complex data with higher accuracy. A ConvNet is so strong as a model that it can possibly handle the initial MRI scan as input, without any preprocessing, and yet be able to make correct predictions about the course of a condition, this way bypassing the time-consuming step of extracting the features with Freesurfer. However an important demand of these types of networks is the size of the data provided, which needs to be *very large*. This fact renders their use out of reach for medical research, at least for the time being.

4.3 Society & Ethics

Medical research is a field that is very widely discussed, for which there exist many arguments both for and against it, regarding society and ethics. The most common arguments concern the aspect of *consent* and the protection of the *patients' privacy*. It is obvious that no research can be possibly done without the informed consent of a patient, or in general any person that will participate in an experiment[1]. With proper care and handling, personal data can be protected and research can be done, when the objective is in the best interest of people.

Moreover, people argue on the matter of AI replacing doctors, which is something that can be unsettling. However, if proper methods are developed they can, instead of replacing a doctor, constitute a significant aid in the process of diagnosing or predicting certain conditions early enough, so as to render them treatable or even avoidable. There is already a number of medical occasions where AI is used, e.g. in certain surgical procedures, where the accuracy and stability of a machine (handled by a trained doctor) can significantly affect the outcome of the operation. In the specific case of the thesis' objective, a method that can predict progression to AD can be extremely valuable to a doctor and a patient.

[1] For this thesis ethics approval was obtained at each institution involved in the study. Written consent is obtained from all subjects and/or authorized representatives

Bibliography

[1] Carlos Aguilar, Eric Westman, J-Sebastian Muehlboeck, Patrizia Mecocci, Bruno Vellas, Magda Tsolaki, Iwona Kłoszewska, Hilkka Soininen, Simon Lovestone, Christian Spenger, Andrew Simmons, and Lars-Olof Wahlund. Different multivariate techniques for automated classification of MRI data in alzheimer's disease and mild cognitive impairment. *Psychiatry Research: Neuroimaging*, 212:89–98, May 2013.

[2] Anne Brown Rodgers. *Alzheimer's Disease: Unraveling the Mystery.* National Institute on Aging, September 2008.

[3] Ying Chen and Tuan D Pham. Development of a brain mri-based hidden markov model for dementia recognition. *BioMedical Engineering OnLine*, 12:S2, April 2013.

[4] Ying Chen and Tuan D Pham. Sample entropy and regularity dimension in complexity analysis of cortical surface structure in early alzheimer's disease and aging. *Journal of Neuroscience Methods*, 215:210–217, May 2013.

[5] Simon Duchesne, Anna Caroli, Christina Geroldi, D. Louis Collins, and Giovanni B. Frisoni. Relating one-year cognitive change in mild cognitive impairment to baseline MRI features. *Neuroimage*, 47:1363–1370, October 2009.

[6] Javier Escudero, John P. Zajicek, and Emmanuel Ifeachor. Machine learning classification of MRI features of alzheimer's disease and mild cognitive impairment subjects to reduce the sample size in clinical trials. In *2011 Annual International Conference of the IEEE Engineering in Medicine and Biology Society*, pages 7957–7960. IEEE, 2011.

[7] Farshad Falahati, Daniel Ferreira, Hilkka Soininen, Patrizia Mecocci, Bruno Vellas, Magda Tsolaki, Iwona Koszewska, Simon Lovestone, Maria Eriksdotter, Lars-Olof Wahlund, Andrew Simmons, and Eric Westman. The effect of age correction on multivariate classification in alzheimers disease, with a focus on the characteristics of incorrectly and correctly classified subjects. *Brain Topography*, 29(2):296–307, October 2015.

[8] Farshad Falahati, Eric Westman, and Andrew Simmons. Multivariate data analysis and machine learning in alzheimer's disease with a focus on structural magnetic resonance imaging. *Journal of Alzheimer's Disease*, 41:685–708, March 2014.

[9] Bruce Fischl. Freesurfer. *NeuroImage*, 62(2):774–81, August 2012.

[10] Laboratory for Computational Neuroimaging. Freesurfer. Athinoula A. Martinos Center for Biomedical Imaging, 2013. `https://surfer.nmr.mgh.harvard.edu/`.

[11] Wellcome Trust Centre for Neuroimaging. SPM. `http://www.fil.ion.ucl.ac.uk/spm/`.

[12] Python Software Foundation. Python language reference. `http://www.python.org`.

[13] Zoubin Ghahramani. An introduction to hidden markov models and bayesian networks. *Journal of Pattern Recognition and Artificial Intelligence*, 15:9–42, 2001.

[14] Alex Graves, Abdel-rahman Mohamed, and Geoffrey E. Hinton. Speech recognition with deep recurrent neural networks. *CoRR*, abs/1303.5778, 2013.

[15] Charles M. Grinstead and J. Laurie Snell. *Introduction to probability*, chapter 11, pages 405–470. John Ewing, 2nd edition.

[16] Trevor Hastie, Robert Tibshirani, and Jerome Friedman. *The Elements of Statistical Learning*. Springer Series in Statistics. Springer New York Inc., New York, NY, USA, 2001.

[17] IMAIOS SAS. Mri step-by-step, interactive course on magnetic resonance imaging. Online Course. `https://www.imaios.com/en/e-Courses/e-MRI`.

[18] M. Jenkinson, C.F. Beckmann, T.E. Behrens, M.W. Woolrich, and S.M. Smith. FSL. *NeuroImage*, 62:782–790, 2012. `https://http://fsl.fmrib.ox.ac.uk/`.

[19] Kota Katanoda, Yasumasa Matsuda, and Morihiro Sugishita. A spatio-temporal regression model for the analysis of functional MRI data. *NeuroImage*, 17:1415–1428, November 2002.

[20] Arne Leijon and Gustav Eje Henter. *Pattern Recognition - Fundamental Theory & Exercise Problems*. KTH - School of Electrical Engineering, 2015. Lecture Notes for the course EQ2340.

[21] Susanne G. Mueller, Michael W. Weiner, Leon J. Thal, Ronald C. Petersen, Clifford R. Jack, William Jagust, John Q. Trojanowski, Arthur W. Toga, and Beckett Laurel. Ways toward an early diagnosis in alzheimers disease: The alzheimers disease neuroimaging initiative (adni). *Alzheimer's & Dementia*, 1(1):55–66, July 2005.

[22] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay. Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, 12:2825–2830, 2011.

[23] Alicia Quiros, Raquel Montes Diez, and Dani Gamerman. Bayesian spatiotemporal model of fmri data. *NeuroImage*, 49:442–256, January 2010.

[24] Lawerence R. Rabiner. A tutorial on hidden markov models and selected applications in speech recognition. In *Proceedings of the IEEE*, volume 77, pages 257–286. IEEE, February 1989.

[25] Susan M. Resnick, Dzung L. Pham, Michael A. Kraut, Alan B. Zonderman, and Christos Davatzikos. Longitudinal magnetic resonance imaging studies of older adults: a shrinking brain. *Journal of Neuroscience Methods*, April 2003.

[26] Amiya Sarkar. Understanding the basic principles of nuclear magnetic resonance imaging. Online Blog, June 2010. `http://physiology-physics.blogspot.se/2010/06/understanding-basic-principles-of.html`.

[27] Gabriela Spulber, Andrew Simmons, J-Sebastian Muehlboeck, Patrizia Mecocci, Bruno Vellas, Magda Tsolaki, Iwona Kłoszewska, Hilkka Soininen, Christian Spenger, Simon Lovestone, Lars-Olof Wahlund, and Eric Westman. An MRI-based index to measure the severity of alzheimers disease-like structural pattern in subjects with mild cognitive impairment. *Journal of internal medicine*, 273:396–409, January 2013.

[28] M Symms, H Jager, K Schmierer, and T Yousry. A review of structural magnetic resonance neuroimaging. *Journal of Neurology, Neurosurgery, and Psychiatry*, 75, September 2004.

[29] Johan Trygg and Svante Wold. O2-pls, a two-block (xy) latent variable regression (lvr) method with an integral osc filter. *Journal of Chemometrics*, 17(1):53–64, January 2003.

[30] hmmlearn. `http://hmmlearn.readthedocs.org`.

[31] Bing Wang and Tuan D Pham. MRI-based age prediction using hidden markov models. *Journal of Neuroscience Methods*, 199:140–145, July 2011.

[32] Ying Wang, Ssan M. Resnick, and Christos Davatzikos. Spatio-temporal analysis of brain MRI images using hidden markov models. *Medical Image Computing and Computer-Assisted Intervention MICCAI 2010*, pages 160–168, 2010.

[33] Narada Dilp Warakagoda. A hybrid ANN-HMM ASR system with NN based adaptive preprocessing. Master's thesis, Norges Tekniske Høgskole, May 2010.

[34] Chong-Yaw Wee, Pew-Thian Yap, Dinggang Shen, and Alzheimer's Disease Neuroimaging Initiative. Prediction of alzheimer's disease and mild cognitive impairment using baseline cortical morphological abnormality patterns. *Hum Brain Mapp.*, December 2013.

[35] S. Wold, A. Ruhe, Wold H., and W. J. Dunn III. The collinearity problem in linear regression. the partial least squares (pls) approach to generalized inverses. *SIAM Journal on Scientific and Statistical Computing*, 5(3):735–743, 1984.

[36] Svante Wold, Johan Trygg, Anders Berglund, and Henrik Antti. Some recent developments in pls modeling. *Chemometrics and Intelligent Laboratory Systems*, 58(2):131–150, October 2001.

[37] Daoqiang Zhang, Yaping Wang, Luping Zhou, Hong Yuan, Ginggang Shen, and Alzheimer's Disease Neuroimaging Initiative. Multimodal classification of alzheimer's disease and mild cognitive impairment. *Neuroimage.*, April 2011.

[38] Andrew Zisserman. Lecture notes, 2015. University of Oxford, Information Engineering, Machine Learning Course.

Appendix A

Basic Brain Anatomy

The brain is divided into different parts, all responsible for different functions [2].

Cerebral Hemispheres Initially the brain is divided to **cerebral hemispheres**, most commonly referred to as left and right hemisphere, taking up almost 85% of its total weight. The two hemispheres appear to implement different functionalities. The left hemisphere seemingly focuses on details (e.g. recognizing a person in the crowd), whereas the right hemisphere focuses on broad background (e.g. understanding the relative position of objects in space). The outer layer of the hemispheres is called the **cerebral cortex** and is responsible for the processing of incoming information as well as regulating cognitive functions.

Cerebellum Located at the base of the brain, the **cerebellum** takes up about 10% of the total weight. It is also divided into two hemispheres and it is responsible for the body's balance and coordination. It receives information from the eyes, ears, muscles and joints regarding the body's movements and position.

Brain stem The **brain stem** is placed close to the cerebellum and it is the connection between the brain and the spinal cord. It is the part that controls all the unconscious functions and behaves as the messenger for signals sent between the brain and other organs.

Other Parts Within the cerebral hemispheres, we have located several other important parts, such as the hippocampus, the hypothalamus etc., all of which are responsible for different bodily functionalities and as a whole are crucial for a normal living.

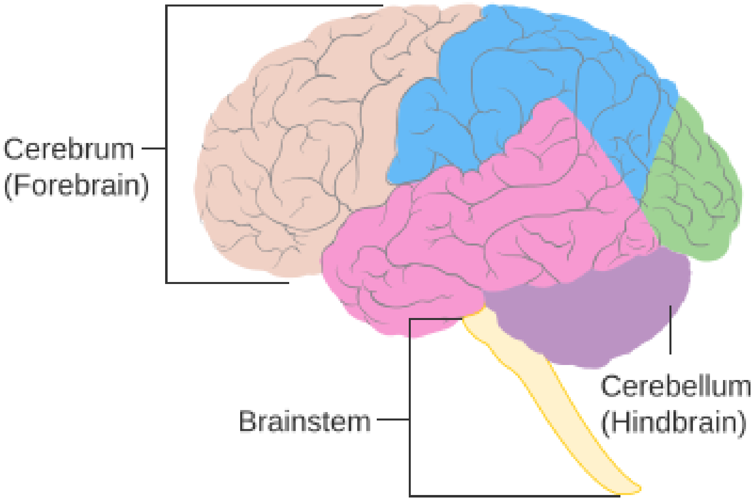


Figure A.1: Basic Brain Anatomy from *Wikipedia*

Appendix B

Magnetic Resonance Imaging

Magnetic Resonance Imaging (MRI) is an imaging technique, which is becoming more and more popular in the medical field both in diagnostics and in research and also in many cases in producing images of non-living objects. Due to its non-invading nature and lack of use of ionizing radiation, it is favoured over other techniques. The high resolution images, the ability to portray different physiologies and other features (functional MRI, real-time MRI etc.) only add to the list of its benefits.

The Science Behind the MRI

The MRI was first used in vivo by the end of the 1970's [28] and has ever since been constantly researched in order to improve its efficiency (speed, image resolution etc.) and to also expand its applicability and features. However, the phenomenon on which the MRI is based, the Nuclear Magnetic Resonance (NMR), was first observed in 1945.

Nuclear Magnetic Resonance

As is widely known, the atoms are composed by the electrons (negatively charged particles) that are orbiting around the atom's nucleus (its core). The nucleus is composed by the protons (positively charged particles) and the neutrons (charge-less particles). The protons (hydrogen nuclei), due to their charge, behave like tiny rotating magnets, producing a microscopic magnetic field around them. This is exactly how we can perceive them in a macroscopic world, by their charge, their nuclear spin and they are commonly represented by a vector that coincides with the rotation axis of the nucleus. The sum of the microscopic magnetic fields generated by all the protons is called **net magnetization** [17].

Normally, the different fields are arbitrarily oriented, resulting in a **null net magnetization** (charge-less) (Figure B.1). When an external magnetic field (B_0) is applied to the protons, then all the spins align to that external field, the same way that small magnets would align to a similar field. As seen in Figure B.2, some spins align with the direction of the field (**parallel**) and some align with the opposite direction (**anti-parallel**) (also known as spin-up and spin-down positions respectively).

While the aligned protons spin, they actually revolve (**precess**) around the direction of the magnetic field (B_0) at an angle, while at the same time they rotate around their own axis. This phenomenon is shown in Figure B.3. The precessional frequency or resonance frequency is called **Larmor frequency** and is proportional to the external magnetic field's strength:

$$\boldsymbol{\omega_0 = -\gamma B_0}$$

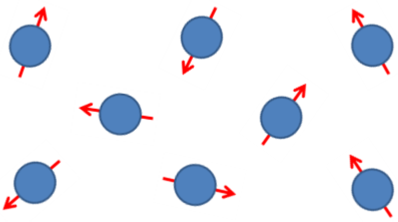


Figure B.1: Null net magnetization [17]

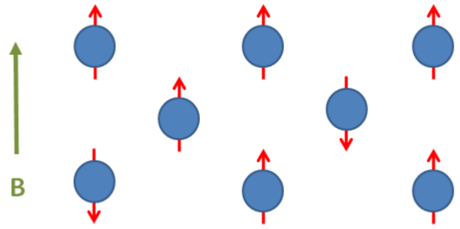


Figure B.2: Nuclei aligned to external field [17]

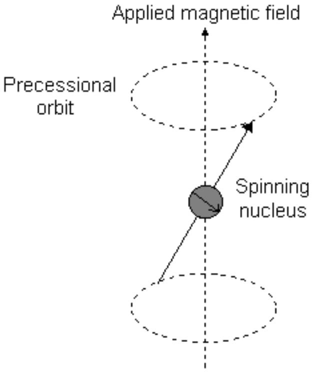


Figure B.3: Larmor frequency [26]

So, the proton's vector can be broken down into two components, a longitudinal and a transverse component. The rotation of the transverse component is what gives the precession.

In general protons tend to align parallel to the field in larger numbers than anti-parallel to the field, resulting in a net magnetization of a certain strength along the longitudinal axis (Z axis). On the other hand, since the protons are not rotating in phase, the sum of their transverse spins results in a null magnetization on the XY plane.

At this point if a second stronger magnetic field (B_1), with perpendicular direction to B_0, is applied to the protons, they align to it. This means that we now have a new longitudinal net magnetization of a certain strength (along the direction of B_1) and a new transverse null magnetization. In other words the original longitudinal magnetization becomes null and the original transverse magnetization increases. This is called spin excitation.

After B_1 stops being applied, then the protons return gradually to their initial state of alignment with B_0 (spin relaxation). While relaxation happens, a small but detectable amount of electromagnetic energy is emitted by the protons (**NMR** signal). We can observe this phenomenon by two different points-of-view: the increase of the longitudinal magnetization to its original state and the decrease of the transverse magnetization to null.

When observing these changes, we can calculate the times needed for them to happen. We call $T1$ the time needed for the longitudinal magnetization to reach 63% of its final value and $T2$ the time needed for the transverse magnetization to reach 37% of its original value (lose 63% of its original value). $T2$ is shorter than $T1$ and since it is tissue-specific, it is unrelated to the strength of the applied field, while $T1$ gets longer as the field's intensity increases.

Table B.1: Relaxation times for various tissues at 1.5 T

	$T1$ (ms)	$T2$ (ms)
Water	3000	3000
Gray matter	810	100
White matter	680	90
Liver	420	45
Fat	240	85

MRI Signal Recording

As said previously, the MRI takes advantage of the NMR. Specifically it takes advantage of the magnetic energy emitted by the protons during relaxation. When the patient enters the MRI machine, a large magnet that envelopes the patient produces the B_0 magnetic field. At the same time an electromagnetic *radio frequency* (RF) signal transmitter and receiver produces the B_1 magnetic field.

An important factor at this point is that the signal transmitted is oscillating with a specific frequency (**Resonance**[1]). This way, only the protons that spin with the desired frequency will respond to the signal causing the excitation and relaxation phenomena described in the previous section. When excitation and relaxation occur, this means that the targeted protons get dephased and back into phase with B_0. Initially a 90° RF pulse causes the protons to dephase after a specific time length, defined as $\frac{TE}{2}$, where TE stands for *Echo Time*. Then a 180° RF pulse causes the protons to get back into phase in time TE. In other words, *Echo Time* (TE) is the time between the 90° RF pulse and the sampling of the MR signal, while another parameter that can be tuned is the *Repetition Time* (TR), which is the time between two excitation pulses (the time between the application of two consecutive 90° RF pulses).

When the electromagnetic energy is emitted from the protons, the RF receiver detects and imprints it on an image. Since different tissues have different responses to the RF pulse, they emit energy of different strength. Additionally by knowing the $T1$ and $T2$ of different tissues and adjusting the strength of the RF pulse, as well as the TE and TR accordingly, scientists are able to produce images where different tissues are distinct, or even adjust the contrast of the produced image, according to their current needs.

For example, a tissue with long $T1$ and $T2$ times (like water) is depicted dark when the RF pulse is low (called $T1$-weighted image) and bright when the RF pulse is high (called $T2$-weighted image). On the other hand, a tissue with short $T1$ and long $T2$ times (like fat) is bright in a $T1$-weighted image and gray in a $T2$-weighted image.

[1] Resonance is called the frequency at which one system will interact with another causing it to oscillate with a maximum amplitude. In this particular case, since we have an electromagnetic RF, it is called magnetic resonance.

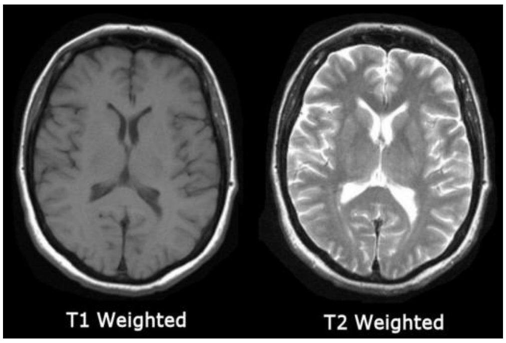


Figure B.4: MRI scan of a brain

Appendix C

Regional Features

3rd Ventricle
4th Ventricle
Accumbens area
Amygdala
Brain Stem
Caudate
Cerebellum cortex
Cerebellum white matter
Corpus callosum anterior
Corpus callosum central
Corpus callosum midanterior
Corpus callosum midposterior
Corpus callosum posterior
Hippocampus
Inferior lateral ventricle
Lateral ventricle
Pallidum
Putamen
Thalamus proper
Sulcal CSF
Ventral diencephalon

Table C.1: Volumetric Measures (mm^3)

Banks of superior temporal sulcus	Parahippocampal gyrus
Caudal anterior cingulate gyrus	Parsopercularis gyrus
Caudal middle frontal gyrus	Parsorbitalis gyrus
Cuneus cortex	Parstriangularis gyrus
Entorhinal cortex	Pericalcarine cortex
Frontal pole	Postcentral gyrus
Fusiform gyrus	Posterior cingulate gyrus
Inferior parietal cortex	Precentral gyrus
Inferior temporal gyrus	Precuneus cortex
Insular cortex	Rostral anterior cingulate gyrus
Isthmus cingulate cortex	Rostral middle frontal gyrus
Lateral occipital cortex	Superior frontal gyrus
Lateral orbito frontal cortex	Superior parietal gyrus
Lingual gyrus	Superior temporal gyrus
Medial orbito frontal cortex	Supramarginal gyrus
Middle temporal gyrus	Temporal pole
Paracentral sulcus	Transverse temporal gyrus

Table C.2: Thickness measures (mm)