DEGREE PROJECT, IN COMPUTER SCIENCE, FIRST LEVEL
STOCKHOLM, SWEDEN 2022

Accuracy evaluation of using Graph Theory metrics for functional connectivity analysis MCI classification

PABLO URQUIJO

KTH ROYAL INSTITUTE OF TECHNOLOGY

CSC SCHOOL

Accuracy evaluation of using Graph Theory metrics for functional connectivity analysis MCI classification

PABLO URQUIJO

Supervisor: Arvind Kumar

Examinator: Pawel Herman

Swedish title: Utvärdering av noggrannheten vid användning av grafteorimätningar för funktionell konnektivitetsanalys MCI-klassificering

DA150x EECS/KTH

2022-06 Sweden, Stockholm

Abstract

Mild Cognitive Impairment (MCI) is an intermediate stage between the physiological cognitive deterioration due to ageing and a nonphysiological stage of dementia. MCI has morbidity between 12% and 18% of the population aged over 65 [1]. Even though MCI is not considered dementia and the symptomology of the patients is not severe, the patients with this disease are at higher risk of subsequently developing dementia and other neurodegenerative illnesses like Alzheimer's (AD).

Functional connectivity analysis has proven helpful in enhancing our knowledge of cognitive disorders such as MCI. The functional connectivity analysis is often combined with Graph Theory to extract metrics about the brain state and improve the classification of MCI patients using Machine Learning. In this report, I studied the classification accuracy improvement of using the Graph Theory metrics alongside the single functional connections for a specific dataset. Additionally, I explored the features that reported higher differences between MCI and healthy patients.

I found that using only Graph Theory metrics for this dataset does not provide enough accuracy but that using them alongside the individual functional connections enhances the accuracy value. In addition, regarding the Graph Theory metrics, the node features proved to be more relevant for the classification than the whole matrix features. Additionally, I observed a significantly altered behaviour in some brain regions.

Sammanfattning

Lindrig kognitiv störning (eng mild cognitive impairment, MCI) är ett mellanstadium mellan den fysiologiska kognitiva försämringen på grund av åldrande och ett icke-fysiologiskt stadium av demens. MCI drabbar mellan 12 och 18 % av befolkningen över 65 år [1]. Även om MCI inte betraktas som demenssjukdom, och patienternas symtom inte är allvarliga, löper patienter med denna sjukdom större risk att senare utveckla demens och andra neurodegenerativa sjukdomar som Alzheimers sjukdom (AD).

Analys av funktionell konnektivitet har visat sig vara till hjälp för att öka vår kunskap om kognitiva störningar som MCI. Den funktionella konnektivitetsanalysen kombineras ofta med grafteori för att få fram mått på hjärnans tillstånd och förbättra klassificeringen av MCI-patienter med hjälp av maskininlärning. I den här rapporten har jag studerat den förbättrade klassificeringsnoggrannheten genom att använda grafteori tillsammans med de enskilda funktionella förbindelserna för ett specifikt dataset. Dessutom undersökte jag vilka egenskaper som gav upphov till större skillnader mellan MCI- och normala patienter.

Jag fann att det inte ger tillräcklig noggrannhet att använda enbart Graph Theory-metriker för detta dataset, men att användningen av dem tillsammans med de enskilda funktionella förbindelserna förbättrar noggrannhetsvärdet. När det gäller grafteorimetrikerna visade sig dessutom nodfunktionerna vara mer relevanta för klassificeringen än funktionerna för hela matrisen. Dessutom observerade jag ett betydligt förändrat beteende i vissa hjärnregioner.

Contents

Chapter 1

Introduction

Mild Cognitive Impairment (MCI) is an intermediate stage between the physiological cognitive deterioration due to ageing and a nonphysiological stage of dementia. MCI produces an early loss of memory and other cognitive capacities; its symptomology can be perceived by their families and friends, but the MCI usually doesn't evoke a state of dependence in the patient to do quotidian activities. The MCI has morbidity between 12% and 18% of the population aged over 65 [1]. Even though MCI is not considered dementia, and the symptomology of the patients is not severe, the patients with this disease are at higher risk of subsequently developing dementia and other neurodegenerative illnesses like Alzheimer's (AD). MCI early diagnosis and treatment could lead to lower chances of developing these other neurodegenerative illnesses; however, only Biogen's aducanumab drug has been approved by the Food and Drug Administration to treat MCI, and its effectiveness has been a controversial topic ever since [21]. To accomplish this early diagnosis and treatment, recently, some studies have correlated the MCI with the signal obtained from the functional connectivity analysis.

The functional connectivity analysis data is obtained from the resting-state functional magnetic resonance imaging (rs-fMRI). As its name implies, the rs-fMRI consists of acquiring functional magnetic resonance imaging (fMRI) while the patient is resting. rs-fMRI can provide some valuable data about the functional connections of the brain regions. To extract the functional connectivity data from the rs-fMRI, first, the 3D image of the rs-fMRI is segmented into ROIs (regions of interest) with an already segmented atlas. After the ROIs segmentation, we obtain these regions' signals across the time domain and compare them with each other to create a correlation array. In the resulting array, the off-diagonal elements represent the correlation between two regions and the diagonal elements are usually set to zero.

The correlation strengths of the functional connectivity arrays reflect excitatory and inhibitory relations. A functional connectivity network of the brain can be represented using a Graph Theory approach from these arrays. Graph Theory can provide multiple metrics that characterise the state of the functional brain network. The implementation of these metrics with the addition of Machine Learning models for classification has been recently used to diagnose patients with diseases that alter their cognitive performance. However, the accuracy enhancement for the classification obtained by these metrics is not well established and variates with different atlas. For this dataset, the relevancy of the Graph Theory metrics is still unknown. [2][3][4][5]

1.1 Research Question

This study aims to see if Graph Theory metrics can improve MCI patients' classification accuracy using different non-ensemble Machine Learning methods and Ensemble Learning. In addition, among these metrics, see which ones are more representative to make this classification and infer biological knowledge based on the obtained results. Additionally, find out which elements of the array are more representative as well; in other words, observe which connections determine the classification of control samples and MCI patients. Studying this will induce physiological information that can be used for medical purposes. I aim to answer the following research questions:

- Can Graph Theory metrics improve the accuracy of classifying MCI patients for this concrete dataset?

- Which features differ the most between MCI and healthy patients?

For studying the accuracy, not all the Machine Learning classifiers were used; to narrow this project's scope, only the most used Machine Learning methods for functional connectivity analysis were used. These are Random Forest, AdaBoost, Gradient Descending, Logistic Regression, Naïve Bayes, Decision Trees, and Support Vector Machines. All these are partially optimized for the sake of reaching their best performance.

The graph metrics studied are metrics that have achieved good results in other studies with this kind of data.

1.2 Approach

Multiple libraries and machine learning methods from Python and MATLAB were used. The data utilized for this project has been provided by Swedish BioFINDER-1; it consists of a 4D volume with a size of 6x94x200x200 and the ground truth of the classification. The volume dimensions are:

1. 6 times delays for each subject.
2. 94 patients, of which 44 of them have MCI and the 50 remainders are control.
3. 200 ROIs
4. 200 ROIs

200x200 numbers that represent the correlation between ROIs

The correlation of the array is directional, meaning that A can inhibit/excite B but not necessarily the other way around, which produces an asymmetrical array where C_{ij} may not be equal to C_{ji}. The biological inference should be afterwards contrasted by a professional in the field. The resulting model with higher accuracy could improve the early diagnosis of MCI subjects.

Chapter 2

Background

2.1 MCI

The MCI or Mild Cognitive Impairment consists of a neurological disease where the memory and other cognitive capacities are more deteriorated than expected from the physiological deterioration due to the ageing of the subject. Even if it doesn't interfere with the daily activities, it can be noticed by the people surrounding the affected and might lead an above-average death rate in a period of six years compared to the rest of the healthy population of the same age range. This death rate is also affected in patients of this kind by other features such as sex, years of education, history of heart disease, and the realisation of moderate physical exercise. Furthermore, the main concern is that patients suffering from this condition have an increased risk of developing more severe types of dementia like Alzheimer's disease. The main symptoms of MCI are memory loss and deterioration of other cognitive capacities such as language, attention, and decision-making [2]. Suffering MCI can also lead to depression, irritability and aggression, anxiety, and apathy.

Mild Cognitive Impairment in older adults has been detected as one of the most critical causes of a higher risk of mortality and morbidity in the modern world. As such, this condition and its consequences increase the health expenses of the society. Cost differences between cognitively normal (CN) and mild cognitive impairment patients don't differ significantly. Still, it does with a prevalent dementia state, which raises to 4,000 $ the annual cost compared with the CN. [1] Therefore, computer-aided diagnosis of the disease can significantly reduce the cost of the disorder and other neurodegenerative illnesses since MCI early detection and correct treatment can considerably decrease and prevent the onset of these other neurodegenerative illnesses.

From the SEN (Spanish Society of Neurology) data, different essential measures of the impact of this disease can be extracted. In Spain, the mean cost of an AD patient goes from 17.100 € to 28.200€. In 2015 there were around 47 million people with dementia worldwide, and it's expected that if the prevalence remains constant, in 2050, there will be 131 millions suffering from this condition. [3]

Since the impact of the MCI and its consequences are undeniable, more reliable diagnosis methods and treatments must be produced to improve the life quality of the patients and reduce the government's expenses.

2.2 rs-fMRI and functional connectivity

Magnetic resonance imaging (MRI) is a non-ionising medical imaging technique that provides high contrast of soft tissue with a high spatial resolution; this makes it a suitable option for studying the brain. With the same hardware, modifying the acquisition parameters, we can obtain different contrast mechanisms: FLAIR, DIFFUSION, ADC…

While the normal MRI gives morphological information, the functional magnetic resonance (fMRI) provides information about brain activity. fMRI can be achieved thanks to the close relationship between neural activity and blood flow. As the region's blood flow increases due to the need for more oxygen, the MRI signal received in that area of the brain slightly increases due to the effect of the haemoglobin on the magnetic field. [4]

MRI is commonly used to observe the brain activity response to specific tasks, this kind of fMRI is called task-based functional MRI. Nevertheless, in 1995, Biswal described the Resting-state fMRI, which, as its name implies, consists of doing fMRI while the patient is not doing any task. In this state, low-frequency oscillation can be obtained and related to spontaneous neural activity. This neural activity is not driven by an external stimulus, which can help classify patients with different neurological disorders. [5]

From the rs-fMRI, the functional connectivity of the brain can be acquired. First, an atlas segments the brain into regions of interest (ROIs). Later, each region's signal across the time domain is correlated with each other, constructing an array where each off-diagonal element corresponds to the correlation between two ROIs. These arrays can be weighted/unweighted and directional/unidirectional. The categorisation can be seen by observing the arrays.

Weighted functional connectivity is when the array's values are not binary. In this case, the array's values determine the strengths of the connections between regions. High positive values relate to excitation, high negative values to inhibition and values close to zero mean no functional connectivity between the two areas of the brain.

Unweighted functional connectivity is binary; there is no distinction between the strength values of the connections; it only states if there is a connection or not.

If the array is directional, there is a discrepancy between source and target. The connections A→B and B→A corresponding to the array elements C_{ij} and C_{ji} have different values, resulting in an asymmetric matrix. This does not happen on unidirectional arrays in which the discrepant behaviour of source and target is not represented, causing symmetrical matrices.

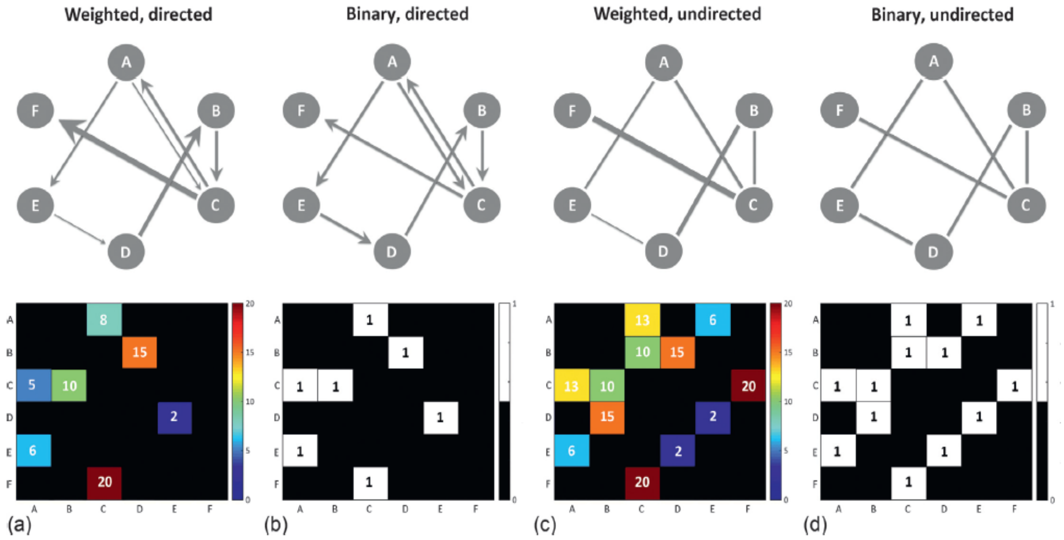

Figure 2.1: Comparison of different types of networks and their corresponding matrices. a) Weighted directed graph and array. b) Unweighted/binary directed graph and array. c) Weighted undirected graph and array. d) Unweighted/binary undirected graph and array. Image courtesy of Alex Fornito, Andrew Zalesky and Edward T. Bullmore [6]

We can obtain unweighted arrays from weighted arrays using a threshold value. However, since weighted matrixes contain more information than unweighted ones it is not possible to convert the other way around. Similarly, directional matrixes can be converted to unidirectional, but the opposite is not possible. [6]

2.3 Graph Theory

A simple definition of a graph would be: a network that helps define and visualise relationships between various components. From a computer science perspective, a graph is: G(V, E), where V is a set of nodes and E is a set of edges. The functional connectivity can be understood from a Graph Theory perspective where each ROI behave as a node, and the correlation strength between two nodes corresponds to an edge. So, brain networks can be studied using the knowledge and tools from this field. Multiple metrics can characterise a Graph, so these can also represent a brain network connectivity.

2.4 Machine Learning

Healthcare is one of the most expensive sectors globally. Its global budget is estimated to be greater than 6 trillion dollars, and it is predicted to reach 12 trillion dollars within seven years [9]. Machine learning is in the spotlight to reduce health expenses and strengthen the relationship between patients and health service workers. Nowadays, Machine Learning in all its shapes can be found in quite a wide range of health fields, such as eHealth, mHealth and image diagnosis.

Machine Learning is a branch of computational algorithms that tries to mimic human behaviour when treating different kinds of inputs to produce a specific outcome. These methods train on past events to predict what will happen given unseen data not used for its training.

The problem solving of Machine Learning algorithms can be differentiated into regression and classification and in supervised and unsupervised. Regression is when the output is continuous, and classification is when the output is discrete. Supervised is when the training phase of the model is done with labelled data and the output for the samples is known, whereas unsupervised is when the outcome is uncertain; clusters of the data are constructed given the likelihood of the sample belonging to that class/cluster.

For this research, classification supervised machine learning was used in pursuance of differentiating MCI patients and control patients, given the features of the functional connectivity arrays.

Machine Learning can provide two sources of information. First, learning; see how the features are related and see which ones are more crucial for correctly modelling the problem. Second, inference; make a prediction of the output given some inputs. I will use both approaches throughout this study.

The Machine Learning methods used in this investigation are superficially explained below.

2.4.1 Ensemble Learning

Ensemble learning is based on the combination of multiple simple methods of Machine Learning. Its goal is to train weak classifiers and combine them with the goal of constructing a more robust model than any of the individual ones. This vision is inspired by the Wisdom of Crowds (J. Surowiecki) principle, which says that a collective is wiser than a single individual. For pursuing this, four requirements must be fulfilled:

1. Diversity of opinion: The individuals must have different training and outputs.

2. Independence: The output of the individual must not be influenced by other individuals.
3. Decentralisation: Each induvial specialises in different features.
4. Aggregation: All predictions can be summed into a single prediction.

Depending on how these single classifiers are aggregated, a binary classification can be done: Bagging and Boosting.

Bagging uses bootstrap replicates, sets of the training data obtained using sampling with replacement. To use this kind of ensemble learning, high variance and low bias classifiers must be used; otherwise, the diversity of opinion would not be satisfied. After training all the individuals, a majority vote is used to determine the classification. It works in parallel. (A. Maki, personal communication, Lecture 10, Machine Learning DD2421).

Boosting, unlike Bagging, uses high bias classifiers. At each training instance, the algorithm is given a weight of how good the previous classifier did the classification. A higher weight means more misclassification, so the next learner must pay more attention to these samples to classify them correctly. [10]

2.4.1.1 Random Forest

Random Forest is a collection of bagged decision trees (2.5.2.3). The feature selection is randomised by inhibiting some dimensions and using replicates for each tree training. This randomness produces the decorrelation of the trees since they will learn from different experiences, in other words, using different training subspaces. If we do not inhibit some of the features, all the trees choose to ask the same questions. They, therefore, would present a similar behaviour, which would not satisfy the diversity of opinion. (A. Maki, personal communication, Lecture 10, Machine Learning DD2421).

2.4.1.2 AdaBoost

This boosting-based algorithm sequentially aggregates the weak learners, unlike Random Forest that works parallelly. After each iteration, the weights and importance of the samples are adjusted to make sure that misclassified samples are taken more into account to be correctly classified in the next iteration. In the end, every weak learner has a reliability coefficient related to the amount of error they produced; this determines the importance of that classifier in the majority vote. When an unseen instance is introduced to the method, the sample will be classified as the majority voted class [16].

2.4.1.3 Gradient Boosting and Extreme Gradient Boosting

Gradient Boosting is similar to the AdaBoost algorithm. However, instead of readjusting the weights, this method tries to fit the next weak learner based on the residual error provided by the previous classifier. It works as an optimization to minimize the loss of these weak learners by employing gradient descent. Extreme gradient boosting is a more regularized form of Gradient Boosting, which provides a better generalization [16].

2.4.2 non-Ensemble learning

Non-ensemble learning refers to traditional machine learning classifiers that do not rely on aggregating learners to perform better.

2.4.2.1 Logistic Regression

Even though it is called regression is used for classification. It relates a vector of features to a distribution that gives us the probability of the sample belonging to one class. We fit a sigmoid function to this distribution and draw a discriminant boundary where the probability of belonging to a class is 0.5. If the sample is placed on one of the sides of this frontier, the sample will be classified with one label, and if it is located on the other side, it will be classified with the other label. (Bob L. T. Sturm, personal communication, Lecture 6, Machine Learning DD2421).

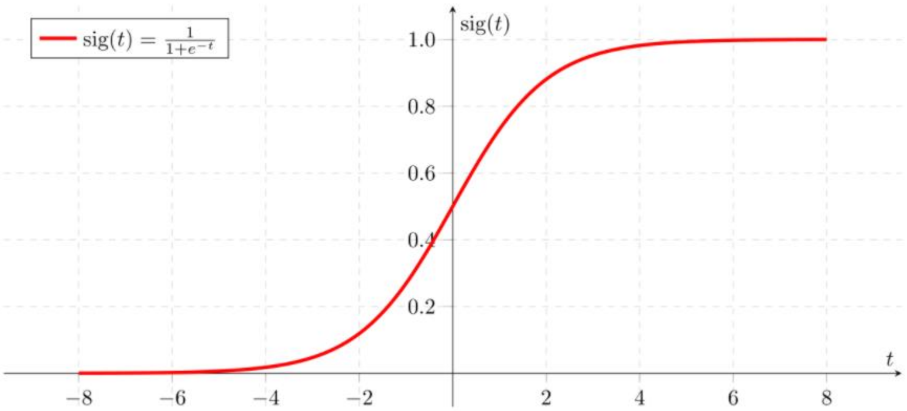

Figure 2.1: Sigmoid function where the 0.5 value is placed on the axis origin of x. In this case, if the input value is negative, it would be classified as 0 and if it is positive to 1. Image courtesy of Saishruthi Swaminathan [11].

2.4.2.2 Naïve Bayes

This method works under the assumption that the dimensions are conditionally independent; the assumption is sometimes violated but still works well under this condition if the dependence is not too strong. We lose some information but save high volume probability space. The Naïve Bayes technique is optimal when the dimensions are at least reasonably independent. To ensure the classifier's good behaviour, it must be examined with the test data. The algorithm uses the Bayes' Rule to determine to which class the sample belongs:

$$P(y|X=x)=\frac{P(x|Y=y)\cdot \Pr(Y=y)}{\Pr(X=x)}$$

Where $P(x|Y=y)$ is the likelihood of observing *x* given that *Y=y*. $Pr\ (Y=y)$ is the prior and represents the previous knowledge of Y before taking any observations. $Pr\ (X=x)$ is the evidence and describes how well the model fits the evidence. Finally, we have $P(y|X=x)$, which is the posterior and describes the probability density of the hypotheses given *x*.

The classification is done by taking the y that maximizes the posterior probability. (Bob L. T. Sturm, personal communication, Lecture 6, Machine Learning DD2421).

2.5.2.3 Decision tree

The training space is sequentially stratified into more simple regions by asking questions about the features until reaching a particular condition. For example, this condition can be related to the state of the leaf nodes or the number of questions, branches generated. Due to the similarity to the trees, this method is called decision tree. The best questions are selected using different split criterions. Among the split criterions, the most used is called information gain. Information gain is the loss of entropy/ unpredictability related to the split of the sample space produced by the asked question. The more certainty we gain about the output, the more information we gain. The questions that provide the highest information gain are located at the top of the tree, therefore being easy to deduce which are the relevant attributes of the dataset that determine the classification. This interpretability is of high esteem to resolve medical-related problems such as the one implicated in this study. (A. Maki, personal communication, Lecture 2, Machine Learning DD2421).

2.5.2.4 SVM

Support Vector Machine's goal is to find a hyperplane that can successfully separate two classes. There is a Structural Risk expected from the variability of finding different hyperplanes that can correctly solve the problem. To solve the Structural Risk, margins, distance to the existing data points of the training are introduced so future samples can be correctly classified and to increase the generalization. Wide margins restrict the possible hyperplanes, so the best hyperplane is the one that maximizes the margin distance to the existing data points. (J. Conradt, personal communication, Lecture 8, Machine Learning DD2421).

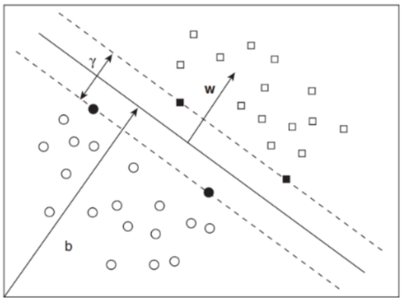

Figure 2.2: Graphic representation of a separating hyperplane with margins. Image courtesy of Mammone, A., Turchi, M., and Cristianini, N. [12]

Figure 2.2 shows how the squares are correctly separated from the circles. If new unseen data falls into the left side of the hyperplane, it is classified as a circle and if it is located on the other side as a square.

Even if one problem cannot be linearly solved in a low dimensional space, it can become separable in a high-dimensional space. However, projecting the samples into higher dimensions can be computationally expensive, so Kernels are used for the purpose of avoiding these expensive calculations. The kernels can exploit the advantages of high-dimensional space without actually projecting it into higher dimensions. The most common kernels are the Linear Kernel, the Polynomial Kernels and the Radial Bases Function (RBF) Kernels:

$$Linear\ Kernel \rightarrow K(\vec{x},\vec{y}) = \vec{x}^T \cdot \vec{y}$$

$$Polynomial\ Kernel \rightarrow K(\vec{x},\vec{y}) = (\vec{x}^T \cdot \vec{y} + 1)^p$$

$$Radial\ Bases\ Function\ Kernel \rightarrow K(\vec{x},\vec{y}) = e^{-\frac{\|\vec{x}-\vec{y}\|^2}{2\sigma^2}}$$

For the polynomial kernel, the exponent p determines the degree, and for the Radial Bases Function (RBF), σ is used to control the smoothness of the boundary. (Ö. Ekeberg, personal communication, Lab 2, Machine Learning DD2421).

2.5 Related work

Even though functional connectivity analysis for disease diagnoses is is a relatively novel field, it has already been used in multiple studies.

For example, researchers from the Middle Tennessee State University and Southern Arkansas University [13] realised a study to classify autism spectrum disorder (ASD) using the multisite ABIDE dataset. This dataset consists of pre-processed resting-state functional MRI data for 1112 patients, 539 with (ASD) and 573 typically developing (TD) participants. The participants' data was obtained at different international sites. To segment the MRI data, they used different atlas to see which one was more fitted for the classification of this type of patient. Studies with only one site and a small number of subjects achieved high accuracy values of around 97% but from multisite datasets, this accuracy was reduced to 70% with a deep neural network classifier. In this study, they managed to improve the state-of-art for multisite type dataset accuracy to 71.98% using Ridge classifier and CC400 atlas. They optimized the amount of computing time as well. The aim of the study was to increase the highest accuracy for supervised machine learning classifiers using optimal parameters for this dataset.

Another related research was carried out by the Department of Neurology of the Xuanwu Hospital [14] with a dataset of 32 subjective cognitive decline patients (SCD), 37 amnestic mild cognitive impairment (aMCI), 30 fully manifested AD and 40 normal controls. The goal of this study was to collect the functional connectivity changes across this spectrum of cognitive states using graph theories methodologies. The conclusive main differential metrics extracted from this study were that subgraph centrality in SCD patients was observed to decrease in the somatomotor and visual networks. aMCI had a decrease and an increase of the global centrality in the primary motor network as well as decreases in the associative networks, and fully manifested AD had increased global centrality in the seven networks.

An inspiring study of the application of advanced machine learning algorithms on rs-fMRI networks for the classification of MCI [15] was done by Ali Khazaee from the University of Bojnord. In this case, the dataset consisted of 89 patients with MCI, 34 with AD and 45 normal controls. Using the graph-theoretical approach, they try to get the highest accuracy possible to the problem of classifying the three groups. The highest accuracy was achieved with the help of Support Vector Machines and resulted in a value of 87.29%.

Functional connectivity analysis can be supported with other kinds of information; the use of functional connectivity analysis alongside structural connectivity analysis has proved to improve the accuracy value for the MCI classification. Chong-Yaw Wee [23] using both types of connectivity analysis achieved an accuracy of 96,3% by employing multiple-kernel Support Vector Machines. The structural information was registered using diffusion tensor imaging (DTI), which provides microstructural characteristics of water diffusion, while the functional was obtained with rs-fMRI. For this study, the atlas used segmented the brain into 90 ROIs. They concluded that structural connection analysis complements functional analysis and that the use of both can enhance the MCI accuracy classification.

Thomas Welton and Daniel A. Kent [26] made a systematic review of the reproducibility of Graph Theory metrics for the Brain network. He collected different aspects of the Graph Theory metrics sensibility regarding how the functional connectivity arrays were obtained. In addition, the authors collected the most promising Graph Theory metrics for future research use. However, these metrics' behaviour depends on the classification problem and other methodological factors, so they cannot be considered as a Ground Truth for all functional analysis problems.

Chapter 3

Methods

3.1 Dataset

The dataset consists of a 4-dimensional matrix with a 6x94x200x200 size. It is composed of weighted directional functional connectivity arrays where the corresponding dimensions are:

- 94 patients of which 50 of them are control, supposedly healthy in a neurological sense. And the remainder 43 have MCI.
- For each of the 94 patients, there are 6 different delays for the obtention of the correlation arrays.
- Finally, we have a 200x200 array of a weighted directional functional connectivity array, where each off-diagonal element represents the correlation strength between two ROIs of the Brain. Diagonal elements are set to zero since the study of the correlation of an ROI with itself is not relevant as it always will result in a value of 1.

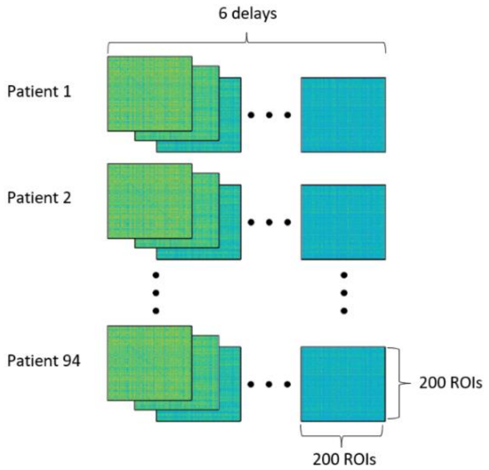

Figure 3.1: Representation of the dataset. A 4-dimensional array where each dimension is: 6 delays, 94 patients, 200 ROIs and 200 ROIs.

The dataset was obtained from the Swedish BioFINDER-1 (no. NCT01208675), study lead by Oskar Hansson, MD, PhD of Lund University [27]. In the study the subjects recruited were between 60 and 80 years old. This range of age was chosen to make the cognitive loss due to the ageing between MCI and healthy subjects the same so that the cognitive difference between the two classes would be due solely to the MCI.

The ROIs were obtained using the Craddock200 atlas to an rs-fMRI; the corresponding labels of the Craddock200 atlas segmentation can be found in Appendix A. After segmentizing the brain, the signals of the ROIs across the time domain were correlated using 6 different delays with the Pearson correlation coefficient, thus obtaining the functional connections. With the Pearson correlation coefficient, the correlation is obtained using the following equation:

$$r = \frac{\sum(x_i - \bar{x})(y_i - \bar{y})}{\sqrt{\sum(x_i - \bar{x})^2 \sum(y_i - \bar{y})^2}}$$

Where:

r = Pearson Correlation Coefficient

x_i = Samples of x

y_i = Samples of y

$\bar{x}$ = Mean of x

$\bar{y}$ = Mean of y

3.2 Feature extraction

Before the feature extraction, the dataset was pre-processed by deleting the patients with NaN values, missing values on the array; this only resulted in the removal of one patient, so the loss of information was not critical.

Metrics from the Graph Theory discipline were extracted using the MATLAB Brain Connectivity Toolbox [Rubinov and Sporns (2010)] [8] and were selected guided by the work of Frank de Vos “A comprehensive analysis of resting-state fMRI measures to classify individual patients with Alzheimer's disease” [7]. In Frank de Vos’s research, they used the same toolbox to classify patients with AD. Using the same metrics is a reasonable procedure since the MCI is an intermediate stage between physiological cognitive loss and an abnormal state like AD.

To extract the features I used both the weighted directional array and the unweighted directional array; to go from the weighted to the unweighted, a threshold of 0.04 was used in order to detect the connections that were greater than zero rejecting the ones that were almost zero as well.

The extracted features can be classified as node features and whole matrix features. For each node feature, I obtained 200 predictors per delay, so 1200 predictors per patient. Each whole matrix feature contributed with one predictor per delay.

The metrics used for the original matrix (non-binarized) were:

- Node features:
  - Weighted Node Betweenness Centrality (BCW): The fraction of all shortest paths in the network that contain a given node, where shortest paths are the routes between two nodes such as the sum of the weights of the edges is minimized. A high value indicates that the node participates in many shortest paths.
  - Weighted Clustering Coefficient (CCW): Geometric mean of all triangles associated with each node.
  - In-strength, Out-strength and Node Strength (IS, OS and S): The in-strength is the sum of weights for the inward connections, the out-strength is the sum of weights for the outward connections and the node strength is the sum of both strengths.
- Whole matrix features:
  - Weighted Transitivity (WT): Fraction of the number of existing triplets. A triplet structure results when two neighbours of a node are neighbours between themselves as well. Is a classical version of the clustering coefficient.
  - Characteristic Path Length (L): Average of shortest paths between all pairs of nodes in the network.
  - Global efficiency (E): Average inverse shortest path length.
  - Network radius (R): Minimal eccentricity.
  - Network diameter (D): Maximal eccentricity. It is a measure of the length of the longest minimal path.

For the binarized matrix:

- Node features:
  - Edge betweenness centrality (BCB): Fraction of all shortest paths in the network that contain a given edge.
  - Binary clustering coefficient (CCB): Fraction of triangles around a node.

    - Degree of each node (DH and DV): Is the number of connections that establishes each ROI. This was done for the inward connections as well as for the outward connections.

- Whole matrix features:
    - Binary Transitivity (BT): Fraction of the number of existing triplets.
    - Degree of the matrix (TD): The sum of all the connections established by the network.

For the feature extraction, three different datasets (Dataset 1, Dataset 2 and Dataset 3) with different features were made to find out the relevance of the Graph Theory metrics for the MCI classification, all of them using MATLAB.

The first set of features (Dataset 1) comprises only the Graph Theory metrics. As previously mentioned (section 2.4), Graph Theory metrics can be used for the study of brain networks such as functional connectivity arrays. However, current network methods cannot reckon the role of negative connections [7]. Therefore, with this methodology, inhibition functional connection information is lost. For this dataset, 10.842 graph features were extracted for each patient resulting in a dataset of 93x10.843 size counting the ground truth of the patients.

The second set of features (Dataset 2) comprises the non-Graph Theory metrics. Dataset 2 is composed of each element of the array, each individual connection, for every delay as a patient feature. This feature was named Pixel Feature since the array can be seen as an image; therefore, each element can be seen as a pixel.

The extraction of the Pixel Feature resulted in a space of 200x200x6=240.000 features for each patient. A prior feature selection process was done to work with this large dataset and be comparable with the other datasets. This feature selection was made with *Relieff*, a MATLAB filtering feature selection tool that gives weights values in concordance to the relevance of the feature. Relevant features are the ones that form well-defined clusters for each class. The filtering was set to return the same number of features as Dataset 1, resulting in another dataset with a size of 93x10.843 counting the ground truth.

The third set of features (Dataset 3) is composed of a mix of the two datasets. This set was obtained by using *Relieff* on both sets of features, obtaining the 5.421 best Graph Theory features and the 5.421 best Pixel Features. This procedure resulted in another dataset with a size of 93x10.843 counting the ground truth.

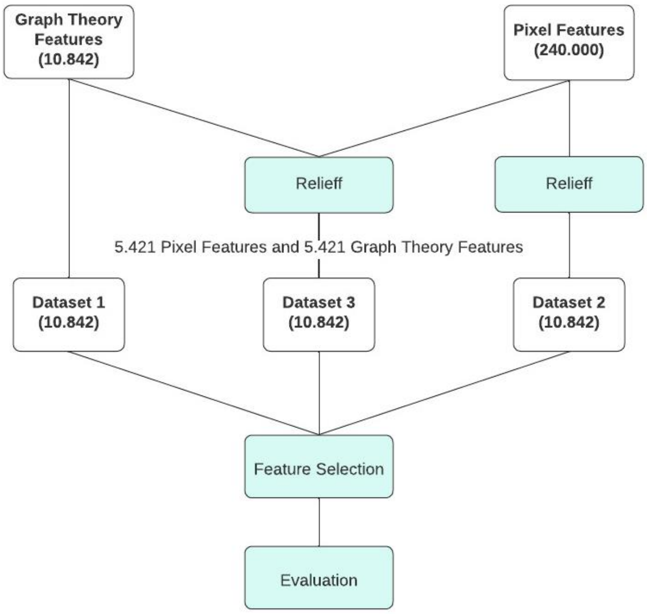

Figure 3.2: Flowchart of the Datasets creation and posterior feature selection and evaluation. The sizes of the Datasets in the image are without considering the ground truth.

Figure 3.2 summarizes the creation and evaluation of the three datasets.

After creating the datasets, these were exported to Python as a comma-separated values (CSV) format with a header indicating the names of the features. The format of the names was the following:

- Pixel feature name structure: "(Pixel Feature) Delay: a | ROIs: b-c".
- Node feature name structure: "(Node feature) Delay: a| Node: b".
- Whole matrix feature name structure: "(Whole matrix feature) Delay: a"

Where "a" can take any value from 1 to 6 and indicates the delay array from which was obtained the feature, while "b" and "c" indicate the functional connection. For the pixel feature structure, "b" is the origin ROI and "c" is the target ROI for the connection.

3.3 Feature selection

Feature selection is an important phase of the Machine Learning classification to determine the dimensions that are more crucial in the distinction of the classes. For such a large quantity of features extracted and a low sampling dataset, this phase is required if we want to avoid the Curse of Dimensionality.

The Curse of Dimensionality refers to the problem that easy to solve problems in low dimensions become harder to solve in higher ones. To avoid the Curse of Dimensionality, Ensemble learning, as well as decision trees, have their intrinsic way of determining the more important features; however, feature reduction methods can boost their performance. More importantly, for the other classifiers, a previous feature selection is essential. Various techniques for dimensionality reduction were used to make the resulting accuracy optimal.

Once in Python, the correlated variables were removed using a threshold of 0.85 so that dimensions correlated more than 0.85 were deleted. The filtering of the correlated variables returned a space of 3.144 uncorrelated dimensions for Dataset 1, 4.174 for Dataset 2 and 4.492 for Dataset 3.

After deleting the correlated features, two different feature selection methods were used with these filtered datasets: PCA and Univariate Selection.

PCA (Principal Component Analysis) consists of projecting the features into new dimensions called principal components that maximize the variance while minimizing the information loss. But before using PCA the features were normalised since the method works using the standard deviation, so it must be correctly scaled. In this experiment, the PCA was done holding 90 % of its original variance.

Univariate Selection was made with "*SelectKBest*", which is a filtering method for feature selection that transforms the data into a subset of k best attributes. The function used for the selection was *f_classif,* which applies ANOVA F-value between label/feature to see how well the feature discriminates between the labels, in this case, between the MCI and the healthy patients. A feature discriminates better if the distance between classes in the space is maximized and the same class samples' variance is minimized [17]. In this experiment, the Univariate Selection was done by selecting a subset of the 30 best attributes.

3.4 Classification and evaluation

After reducing the high dimensionality of the three datasets, the classification and evaluation phase took place. All the Machine Learning algorithms were extracted like the feature selection methods from the Python Scikit library, except the Extreme Gradient Boosting, which was extracted from the XGBoost library. To compare the accuracy performance between the datasets, the following machine learning classifiers were employed:

Non-ensemble methods (NEM):

- Logistic Regression
- Naïve Bayes
- Decision tree
- Support Vector Machine

Ensemble methods (EM):

- Random Forest
- AdaBoosting
- Extreme Gradient Boosting

The three different datasets were introduced to these algorithms with the two different feature selection methods and without using a feature selection method. In order to evaluate the performance of the features, a stratified k-Fold cross-validation was done with "k" equal to 5. 5-Fold cross-validation practice consists of splitting the dataset into 5 folds, where 4 of these folds are used for training, and the one left is used for the evaluation. In the stratified type, the folds are made by preserving the percentage of samples for each class. This process is repeated 5 times, each time leaving a different fold for the evaluation. Then, the overall accuracy of the model is computed as the mean of the different test folds accuracy. The accuracy is a metric used to evaluate the performance of the method and is computed as the number of correct predictions divided by the total number of predictions.

Chapter 4

Results and Analysis

To evaluate the accuracy of the Graph Theory metrics and the Pixel Features, I performed the classification of the patients with the three datasets using three different EM and 4 different NEM classifiers. But to properly compare the potential of the three datasets, first, the hyperparameters were tuned for each one of the classifiers, extracting the hyperparameters that resulted in the best accuracy for each dataset. To evaluate the accuracies, I used a stratified 5-Fold cross-validation. After computing all the folds, the mean accuracy was extracted for the three datasets.

For the extraction of relevant information regarding the MCI disease, I performed a feature analysis to see which features are the ones that exhibit more distinct values between MCI and control patients. This feature analysis was led using the *Relieff* function from MATLAB.

4.1 Optimized Hyperparameters

4.1.1 Logistic Regression optimized hyperparameters

The parameters that I optimized for the logistic regression are the penalty, L1 or L2, and the C value. The L1 penalty is a regularization technique called Lasso Regression that shrinks the feature space making the weights of the less important features zero or close to zero. In contrast, the L2 penalty, also called Ridge Regression, makes the weight values close to zero but not exactly zero. The C value refers to the strength of the penalty; a higher C value will make the model more underfitted, making it generalize better.

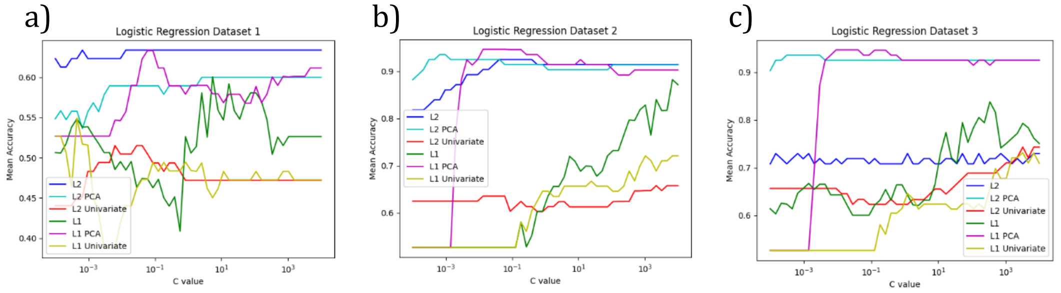

Figure 4.1: a) Mean Accuracy of the Logistic Regression for Dataset 1 for different C values, penalties, and feature selection methods. b) Mean Accuracy of the Logistic Regression for Dataset 2 for different C values, penalties, and feature selection methods. c) Mean Accuracy of the Logistic Regression for Dataset 3 for different C values, penalties, and feature selection methods.

For Dataset 1, Figure 4.1 a) shows that the best combination of the hyperparameters for the Logistic Regression is L2 regularization with a C value equal to 10^{-1} and no feature selection method.

For Dataset 2, Figure 4.1 b) shows that the best combination of the hyperparameters for the Logistic Regression is L1 regularization with a C value equal to 10^{-1} and PCA as the feature selection method.

For Dataset 3, Figure 4.1 c) shows that the best combination of the hyperparameters for the Logistic Regression is L1 regularization with a C value equal to 0.05 and PCA as the feature selection method.

Table 4.1: Optimized Hyperparameters of the Logistic regression for the three datasets and respective accuracy

Dataset	Feature Selection	Regularization	C value	Accuracy
1	None	L2	10^{-1}	0.634
2	PCA	L1	10^{-1}	0.947
3	PCA	L1	0.05	0.947

4.1.2 Naïve Bayes optimized hyperparameters

The hyperparameter that I optimized for the Naïve Bayes classifiers is the var_smoothing. This parameter determines the distribution's variance, widening or narrowing the boundary by collecting more or fewer samples further from the distribution mean.

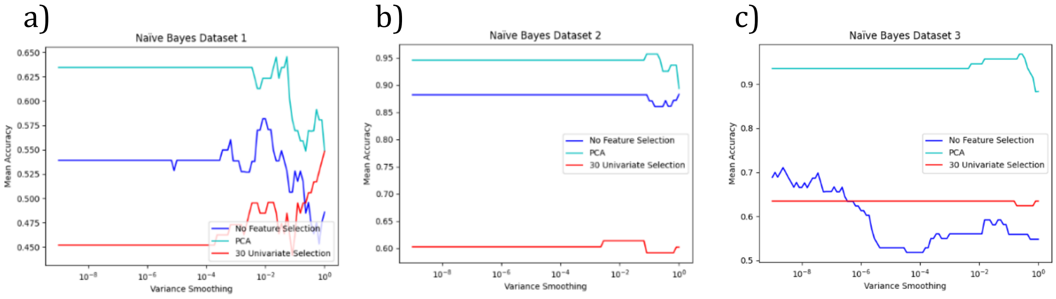

Figure 4.2: a) Mean Accuracy of the Naïve Bayes for Dataset 1 for different Var_smothing values and feature selection methods. b) Mean Accuracy of the Naïve Bayes for Dataset 2 for different Var_smoothing values and feature selection methods. c) Mean Accuracy of the Naïve Bayes for the Dataset 3 for different Var_smoothing values and feature selection methods

For Dataset 1, Figure 4.2 a) shows that the best combination of the hyperparameters for the Naïve Bayes classifier is a Var_smoothing value equal to 0.005 and PCA as the feature selection method.

For Dataset 2, Figure 4.2 b) shows that the best combination of the hyperparameters for the Naïve Bayes classifier is a Var_smoothing value equal to 10^{-1} and PCA as the feature selection method.

For Dataset 3, Figure 4.2 c) shows that the best combination of the hyperparameters for the Naïve Bayes classifier is a Var_smoothing value equal to 0.25 and PCA as the feature selection method.

Table 4.2: Optimized Hyperparameters of the Naïve Bayes for the three datasets and respective accuracy

Dataset	Feature Selection	Var_smoothing	Accuracy
1	PCA	0.005	0.646
2	PCA	0.1	0.957
3	PCA	0.25	0.968

4.1.3 Decision Tree optimized hyperparameters

The parameters that I optimized for the logistic regression are split criterion, Entropy or Gini, and the Max Depth. For the split criterion, Entropy alludes to the information gain and Gini for the Gini Impurity as a measure of the quality of the split. The Max Depth is the condition for the Tree to stop growing. This parameter refers to the maximum depth that the Tree can take. Otherwise, if no parameter is set, it will grow until all the nodes are pure.

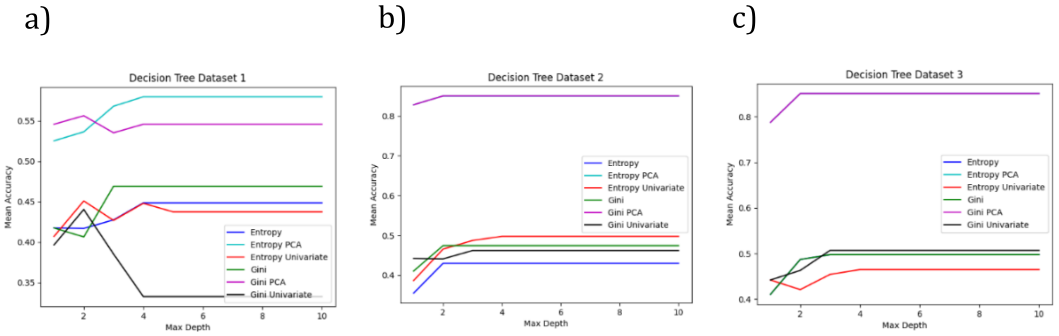

Figure 4.3: a) Mean Accuracy of the Decision Tree for Dataset 1 for different split criteria, max depth values and feature selection methods. b) Mean Accuracy of the Decision Tree for Dataset 2 for different split criteria, max depth values and feature selection methods. c) Mean Accuracy of the Decision Tree for Dataset 3 for different split criteria, max depth values and feature selection methods.

For Dataset 1, Figure 4.3 a) shows that the best combination of the hyperparameters for the decision tree is Entropy as the split criterion and PCA as the feature selection method with a max depth value greater than 4.

For Dataset 2, Figure 4.3 b) shows that the best combination of the hyperparameters for the decision tree is Gini or Entropy as the split criterion since there is no difference in the performance of these two, and PCA as the feature selection method with a max depth value greater than 2.

For Dataset 3, Figure 4.3 c) shows that the best combination of the hyperparameters for the decision tree is Gini or Entropy as the split criterion since there is no difference in the performance of these two, and PCA as the feature selection method with a max depth value greater than 2.

From the optimal max depth values onwards, the accuracy is optimal and remains constant for the three datasets. This occurs since the nodes of the tree become pure with those depth levels so that the optimal value of the accuracy can be reached as well without imposing a Max Depth condition (max_depth=None).

Table 4.3: Optimized Hyperparameters of the Decision Tree for the three datasets and respective accuracy

Dataset	Feature Selection	Split Criteria	Max Depth	Accuracy
1	PCA	Entropy	None	0.580
2	PCA	Gini/Entropy	None	0.850
3	PCA	Gini/Entropy	None	0.851

4.1.4 SVM optimized hyperparameters

For optimizing the behaviour of the SVM, I used 3 different kernels: the linear kernel, the polynomial kernel and the RBF kernel. For the polynomial, I used p=2 and p=3, while for the RBF kernel, I used sigma=0.001, sigma=0.0001 and sigma 0.00001. I optimized the C hyperparameter for each of these models, which regulates how many misclassifications the model is willing to take in the hyperplane selection. Higher C is more rigorous to avoid misclassification, while lower C makes the hyperplane face more misclassification.

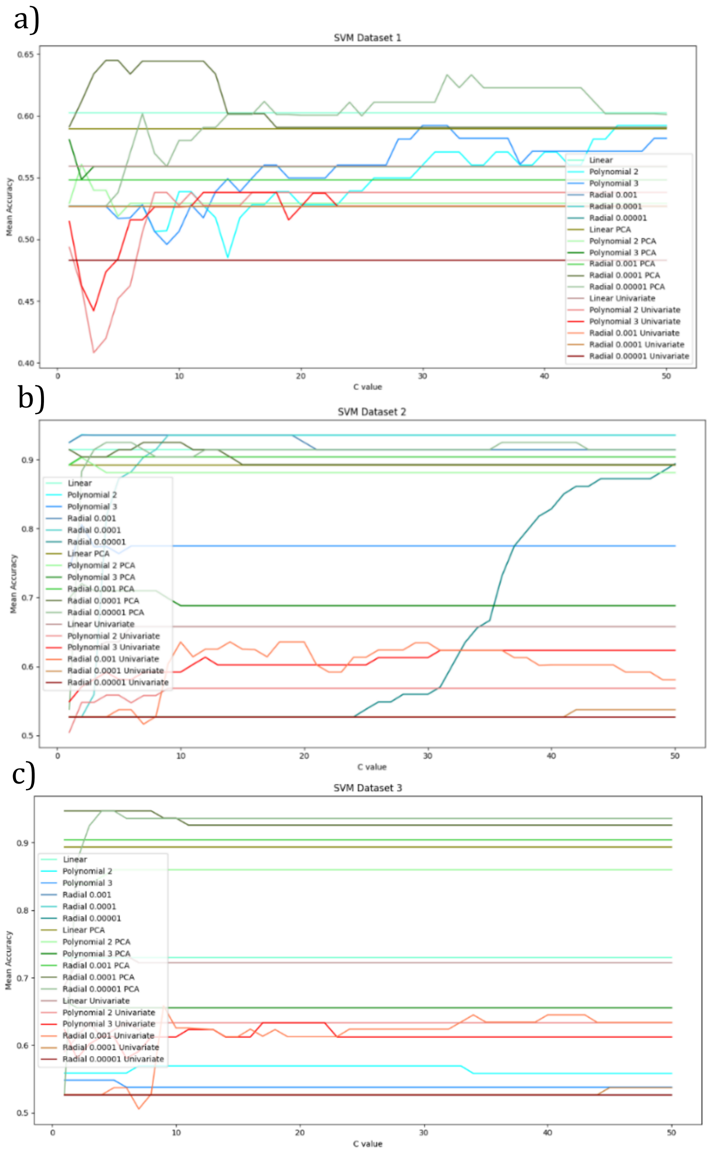

Figure 4.4: a) Mean Accuracy of the SVM for Dataset 1 for different kernels, C values and feature selection methods. b) Mean Accuracy of the SVM for Dataset 2 for different kernels, C values and feature selection methods. c) Mean Accuracy of the SVM for Dataset 3 for different kernels, C values and feature selection methods.

For Dataset 1, Figure 4.4 a) shows that the best combination of the hyperparameters for the SVM is using an RBF kernel with a sigma value equal to 0.0001, a C value equal to 10 and PCA as the feature selection method.

For Dataset 2, Figure 4.4 b) shows that the best combination of the hyperparameters for the SVM is using a polynomial kernel with p equal to 2, a C value equal to 10 and no feature selection method.

For Dataset 3, Figure 4.4 c) shows that the best combination of the hyperparameters for the SVM is using an RBF kernel with a sigma value equal to 0.0001, a C value equal to 5 and PCA as the feature selection method.

Table 4.4: Optimized Hyperparameters of the SVM for the three datasets and respective accuracy

Dataset	Feature Selection	Kernel	C value	Accuracy
1	PCA	RBF 0.0001	10	0.645
2	None	Polynomial 2	10	0.936
3	PCA	RBF 0.0001	5	0.947

4.1.5 Random Forest optimized hyperparameters

To choose an appropriate Random Forest (RF) model, I looked for the number of estimators that resulted in the best accuracy. The number of estimators refers to the number of trees that has a random forest.

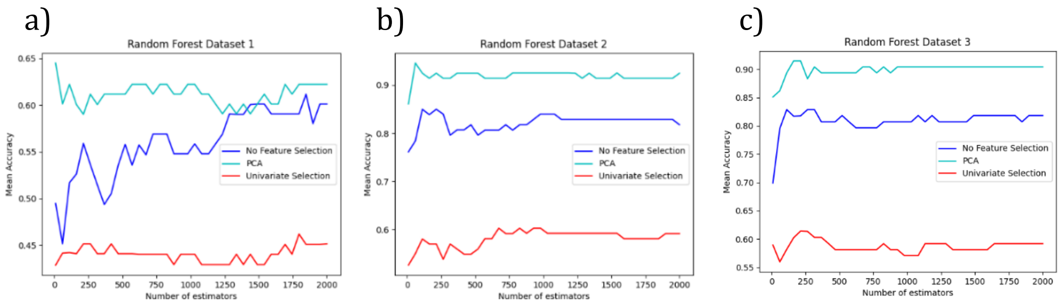

Figure 4.5: a) Mean Accuracy of the Random Forest for Dataset 1 for a different number of estimators and feature selection methods. b) Mean Accuracy of the Random Forest for Dataset 2 for a different number of estimators and feature selection methods. c) Mean Accuracy of the Random Forest for Dataset 3 for a different number of estimators and feature selection methods.

For Dataset 1, Figure 4.5 a) shows that the best combination of the hyperparameters for the RF is using a number of estimators equal to 1 and PCA as the feature selection method.

For Dataset 2, Figure 4.5 b) shows that the best combination of the hyperparameters for the RF is using a number of estimators equal to 60 and PCA as the feature selection method.

For Dataset 1, Figure 4.5 a) shows that the best combination of the hyperparameters for the RF is using a number of estimators equal to 200 and PCA as the feature selection method.

Table 4.5: Optimized Hyperparameters of the RF for the three datasets and respective accuracy

Dataset	Feature Selection	Number of estimators	Accuracy
1	PCA	1	0.645
2	PCA	60	0.946
3	PCA	200	0.915

4.1.6 AdaBoost optimized hyperparameters

For the AdaBoost optimization, the base estimator wasn't changed. I used the default base estimator, a Decision Tree with a max depth of 1. The hyperparameter optimized was the number of estimators like in the Random Forest optimization and the Extreme Gradient Boosting.

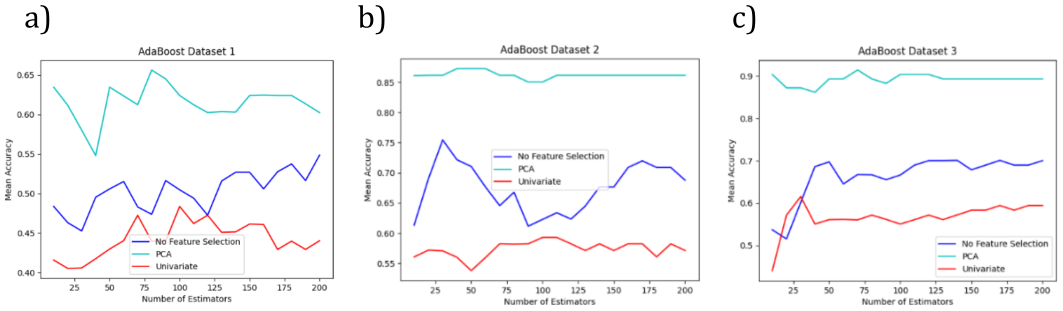

Figure 4.6: a) Mean Accuracy of the AdaBoost classifier for Dataset 1 for a different number of estimators and feature selection methods. b) Mean Accuracy of the AdaBoost classifier for Dataset 2 for a different number of estimators and feature selection methods. c) Mean Accuracy of the AdaBoost classifier for Dataset 1 for a different number of estimators and feature selection methods.

For Dataset 1, Figure 4.6 a) shows that the best combination of the hyperparameters for the AdaBoost uses a number of estimators equal to 80 and PCA as the feature selection method.

For Dataset 2, Figure 4.6 b) shows that the best combination of the hyperparameters for the AdaBoost uses a number of estimators equal to 50 and PCA as the feature selection method.

For Dataset 3, Figure 4.6 c) shows that the best combination of the hyperparameters for the AdaBoost uses a number of estimators equal to 70 and PCA as the feature selection method.

Table 4.6: Optimized Hyperparameters of the AdaBoost for the three datasets and respective accuracy

Dataset	Feature Selection	Number of estimators	Accuracy
1	PCA	80	0.656
2	PCA	50	0.873
3	PCA	70	0.914

4.3.7 XGBoosting optimized hyperparameters

For the Extreme Gradient Boosting (XGBoosting) optimization, two base estimators were tested, a linear and a tree. However, the parameters of the base estimators were maintained by default. The hyperparameter optimized for the ensemble classifier behaviour was the number of estimators.

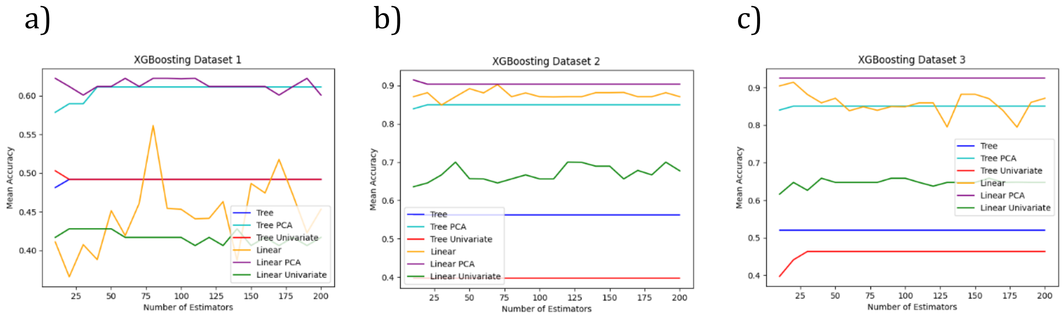

Figure 4.7: a) Mean Accuracy of the XGB classifier for Dataset 1 for a linear base estimator and a decision tree base estimator with a different number of estimators and feature selection methods. b) Mean Accuracy of the XGB classifier for Dataset 2 for a linear base estimator and a decision tree base estimator with a different number of estimators and feature selection methods. c) Mean Accuracy of the XGB classifier for Dataset31 for a linear base estimator and a decision tree base estimator with a different number of estimators and feature selection methods.

For Dataset 1, Figure 4.7 a) shows that the best combination of the hyperparameters for the XGBoosting uses a linear base estimator, with a number of estimators equal to 10 and PCA as the feature selection method.

For Dataset 2, Figure 4.7 b) shows that the best combination of the hyperparameters for the XGBoosting uses a linear base estimator, with a number of estimators equal to 10 and PCA as the feature selection method.

For Dataset 3, Figure 4.7 c) shows that the best combination of the hyperparameters for the XGBoosting uses a linear base estimator, with a number of estimators equal to 10 and PCA as the feature selection method.

Table 4.7: Optimized Hyperparameters of the XGBoosting for the three datasets and respective accuracy

Dataset	Feature Selection	Estimator	Number of estimators	Accuracy
1	PCA	Linear	10	0.622
2	PCA	Linear	10	0.915
3	PCA	Linear	10	0.925

4.2 Comparation of the accuracy

Once selected all the best accuracies for each one of the datasets, the following table collects the accuracy achievable using these datasets (features) with the different Machine Learning classifiers.

Table 4.8: Maximal accuracy achieved for each one of the datasets for all the respective classifiers

Model	Accuracy Dataset 1	Accuracy Dataset 2	Accuracy Dataset 3
Logistic Regression	0.634	0.947	0.947
Naïve Bayes	0.646	0.957	0.968
Decision Tree	0.580	0.850	0.851
SVM	0.645	0.936	0.947
RF	0.645	0.946	0.915
AdaBoost	0.656	0.873	0.914
XGBoosting	0.622	0.915	0.925

Table 4.8 shows that Dataset 1 always shows lower accuracy for each of the Machine Learning classifiers. The results for Datasets 2 and 3 are very similar, except for some cases where the accuracy is slightly improved. The maximum accuracy obtained for the Dataset 1 is 0.656 using AdaBoost, while for Dataset 2 is 0.957 using Naïve Bayes and for Dataset 3 is 0.968 using Naïve Bayes as well. From these results, we can extract that using only graph theory metrics for this particular functional analysis classification problem, does not provide satisfactory outcomes. Nevertheless, using these metrics in addition to the Pixel Feature can provide some improvements to the accuracy.

4.3 Feature Analysis of the Pixel Feature

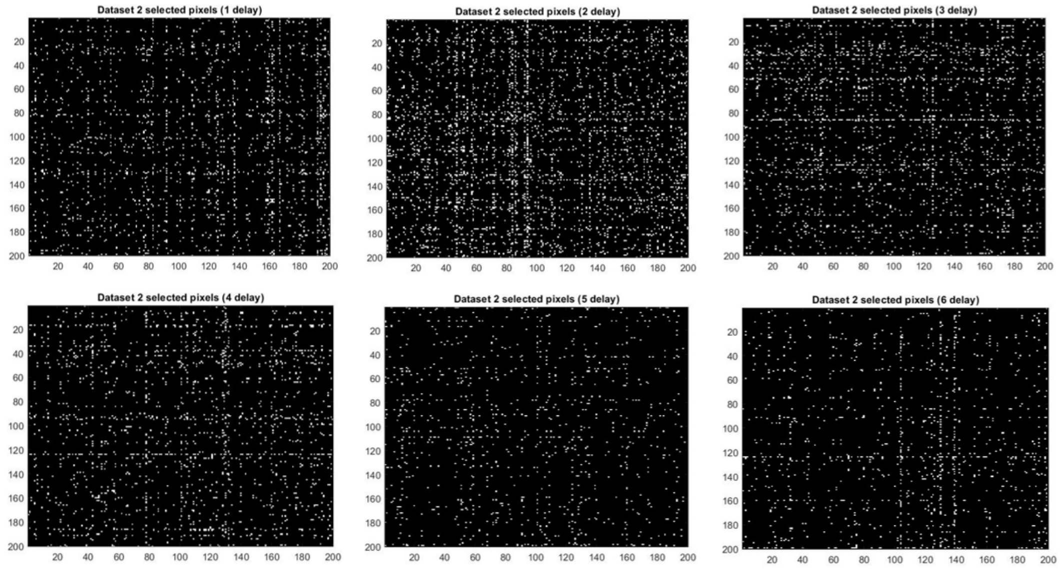

Figure 4.8: Boolean representation of the selected pixels in each one of the delays for Dataset 2. White elements represent selected pixel features and black non-selected pixel features.

To obtain important information about the disease, I studied which features were more different between the control patients and the MCI patients. For doing the feature analysis of the Pixel Feature, I analysed Dataset 2, which consists only of the 10.842 (≈ 4,5%) best pixel features filtered with the MATLAB *Relieff* tool. So, in other words, Dataset 2 is composed of functional connections that present higher differences between control and MCI patients, discrepant pixels. Studying these connections can give us crucial information about the connections that are being altered with the MCI disease.

Figure 4.8 provides an overview of these discrepant pixels between control and MCI for the different delays. Some of these delays have a higher density of white pixels, which indicates that they exhibit more distinct values between the MCI and control. This observation is shown in Figure 4.9 as well.

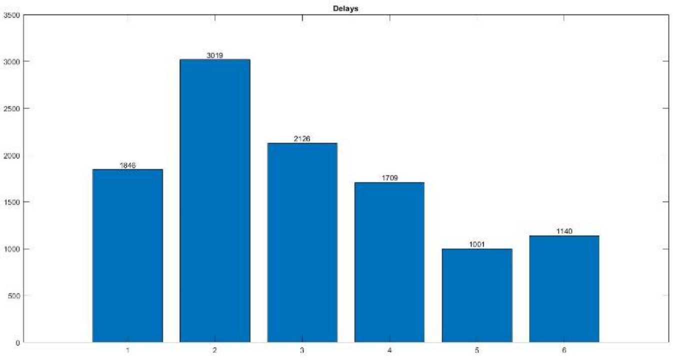

Figure 4.9: Bar graph of the number of select pixels for each one of the delays

Figure 4.9 shows that the second delay is the one from which more pixels are extracted, the one that presents more discrepant pixels, while the fifth delay is the one from which fewer pixels are extracted. This observation matches the results from Figure 4.9 since the second delay shows a higher density of white pixels, and the fifth delay has a lower density of white pixels. As shown in these two figures, the use of different delays can provide helpful information for the classification of MCI patients

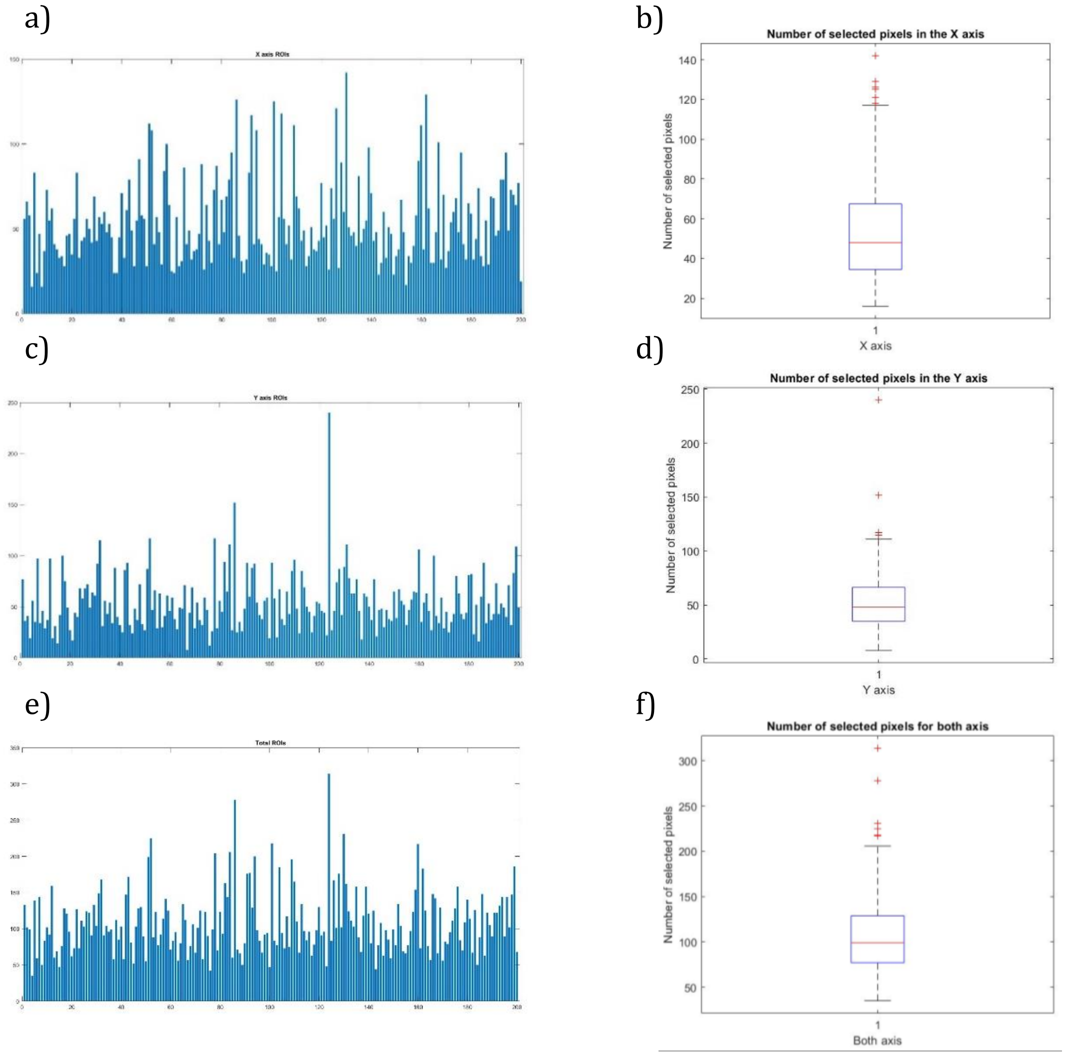

Figure 4.10: a) Bar graph of the number of pixels extracted from the X-axis (the ROIs behaving as a target). b) Boxplot of the number of pixels extracted from the X-axis. c) Bar graph of the number of pixels extracted from the Y-axis (the ROIs behaving as a source) d) Boxplot of the number of pixels extracted from the Y-axis. e) Bar graph of the number of pixels extracted from both axes. d) Boxplot of the number of pixels extracted from both axes.

Figure 4.10 shows from which ROIs are extracted more pixels. From this can be inferred which ROIs have more different values of functional connectivity between MCI and control, in other words, which regions of the brain have a more altered behaviour with the MCI. As shown in Figure 4.10, the number of extracted pixels of the ROI as a target, a), or as a source, b), is significantly different. In both, some ROIs have prominent values of pixels extracted compared to the rest. These prominent values are studied in c) and d) boxplots. The red crosses of the boxplots represent the prominent values, the outliers.

The ROIs that behaving as a target present a large number of discrepant values are in descending order of the number of discrepant pixels: 130, 162, 86, 101, 126 and 104. The ROIs that behaving as a source present a large number of discrepant values are in descending order of the number of discrepant pixels: 124, 86, 52, 78, 32. Among the ROIs as a source, the ROI 124 has an outstanding value of 240, with a distance to the median of 192. That means that the functional connectivity of the ROI 124 is significantly altered in multiple connections with other ROIs.

Since the ROI 86 is present in both boxplots, we can assume that this region has affected both performances, as a source and as a target. Its outlier behaviour is enhanced when both boxplots are summed since its distance to the median is increased, Figure 4.10 f).

To prove that these results weren't just random outliers, I did two randomized analyses. First, the values of the functional connections of each patient were mixed in a random fashion and later filtered with *Relieff* obtaining the 4.5% best pixels. If the outliers are not influenced by *Relieff*, the new bar graph should be mostly uniform, without any outstanding region. The other randomized analysis was to randomly select 4.5% pixels without any filtering. The bar graph should be uniform as well. Both random processes were done 100 times to make them statistically significant.

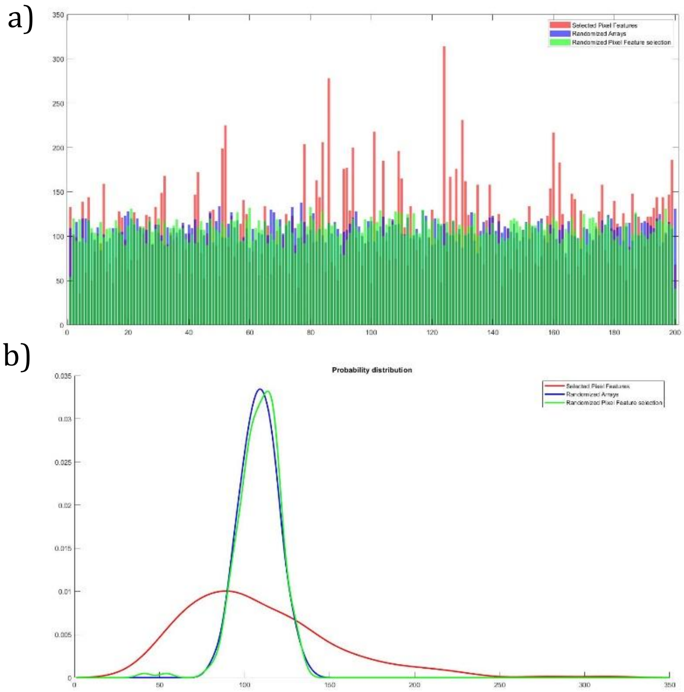

Figure 4.11: a) Bar graph of the non-randomized and the two randomized analyses for both axes. In red, the number of obtained best pixels per ROI. In blue, the number of obtained best pixels per ROI but with the pixels randomized. In green, the number of randomly obtained pixels per ROI. b) Probability distribution for the three analyses.

Figure 4.11 a) and b) confirms the previous hypotheses. The regions with an outstanding number of altered connections cannot be produced randomly. b) shows that the probability of having more than 150 best pixels for an ROI is zero, so using a threshold of 180 is a safe bet to distinct ROI values not produced by a random process.

The three ROIs with a more outstanding number of selected pixels for both axes (Figure 4.10 e) and f)), a higher number of altered connections, are in descending order the 124, the 86, the 52 and the 130. These correspond respectively to the right frontal pole, the right brainstem, the left brainstem and the midbrain. The relation between the numbers and the brain regions can be found in Appendix A.1.

4.4 Feature Analysis of the Graph Theory metrics

For analysing the relevance of the Graph Theory metrics, none of the three datasets was used. Since Dataset 1 is composed of all the Graph Theory metrics, it is not filtered, the importance of these cannot be studied; the Dataset 2 does not have any Graph Theory metric and the Dataset 3 is only filtered by half of its total entries, so the selection is too broad.

So, in order to make the study similar to the one made for the Pixel Feature, the same fraction of the total length is filtered, leaving the 4.5% most important features of the Graph Theory metrics, which results in the selection of the 490 best features.

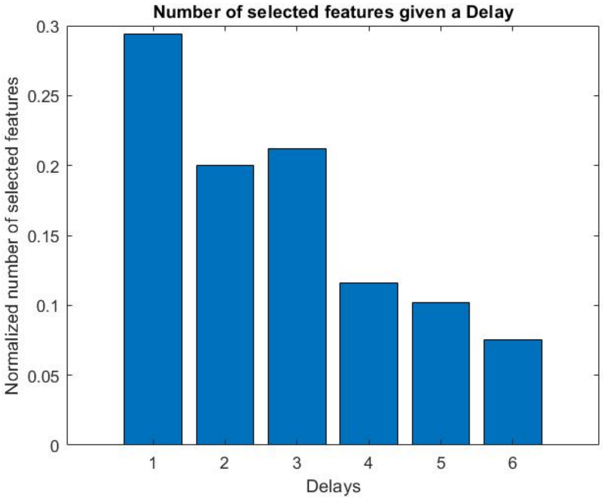

Figure 4.11: Bar graph containing the number of selected best features by delay. This is normalized by dividing the number of selected features by the total number of best features (490).

Figure 4.11 displays that the delays that provide more features are the first one and the third one. This shows no relation with the results obtained for Figure 4.9, where the delay that provided more features was the second one. However, like Figure 4.9, the fourth, fifth and sixth remain the ones that provide fewer features.

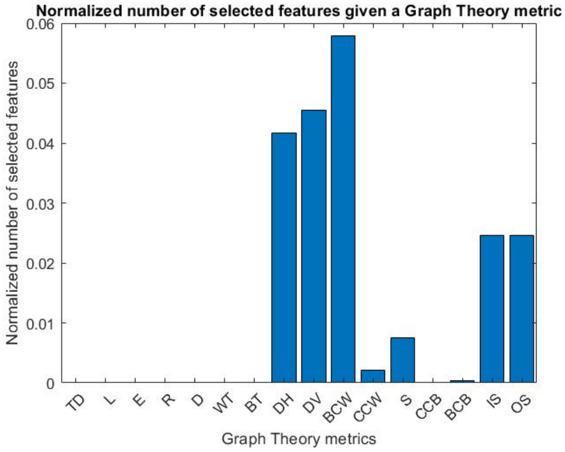

Figure 4.12: Bar graph containing the number of selected best features by Graph Theory metric. The node features are normalized by the total number of nodes per patient (2.400) and the whole matrix features by the total number of delays (6).

Figure 4.12 reveals that a high amount of the best features of the Graph Theory metrics are composed of the Weighted Betweenness Centrality, the Horizontal Degree and the Vertical Degree. Part of the in-strength and out-strength features are selected as well. Also, Figure 4.12 shows an imbalance between the importance of the whole matrix features and the node features since any of the 490 best features relate to whole matrix features.

Chapter 5

Discussion

In the experiments, the following classifiers were used in order to see if there was an improvement in the accuracy when using Graph Theory metrics for the classification of MCI patients: Logistic Regression, Naïve Bayes, Decision Tree, SVM, RF, AdaBoost and XGBoosting. Besides studying the accuracy, the different features were analysed to provide information about the features exhibiting higher differences between the healthy patients and the MCI patients.

5.1 Accuracy comparison of using Graph Theory metrics

The results of the accuracy comparison (Table 4.8) showed that only using Graph Theory metrics is not enough for this particular dataset to provide high accuracy outcomes. Instead, using each element of the functional connectivity array as a feature, the Pixel Feature, evidenced high accuracy. The addition of Graph Theory metrics to the Pixel Feature resulted in a slightly enhanced performance of the accuracy for all the classifiers except the RF.

The fact that for this specific dataset, the Graph Theory metrics are not enough for the correct classification is mainly produced by the small size of the ROIs. This concords with the research conducted by Andreotti that using an array consisting of 154 ROIs, proved that a small ROIs size, in other words, a higher number of ROIs, produce less reliable and repeatable Graph Theory metrics [18]. Thus, Graph Theory metrics by themselves cannot provide sufficient information to produce high accuracy for this dataset, where the number of ROIs is elevated, 200.

5.2 Feature analysis

The analysis of the features that were more discrepant between MCI patients and healthy patients produced interesting new information about the disease and other outcomes that align with previous research results.

An outstanding behaviour of the ROI 124, the ROI 86, the ROI 52 and the ROI 130 was registered for the Pixel Feature. These correspond to the right frontal pole, both sides of the brainstem and the midbrain. For these ROIs, multiple functional connections with other ROIs have abnormal values compared with a healthy patient. Therapies targeting the improvement of the functional connectivity of that region could lead to successful treatments. Treatments like non-invasive brain stimulation have shown promising results in alleviating the AD and MCI symptoms. However, there is no consensus about the best stimulation sites [20]. The altered locations detected in my study, such as the right frontal pole, the brainstem and midbrain might be potential stimulation locations for the non-invasive brain stimulation treatment. These three regions of the brain have memory-related tasks. The results are reasonable since the MCI is a disease that affects cognitive capacities, such as the loss of memory, the most highlighted symptom.

Other treatments aim to improve the whole brain connections. For example, in a study carried out by Noelia Sánchez Pérez and Alberto Inuggi [19], they successfully improved brain functional connectivity by training schoolchildren with computer-based programs that trained them in cognitive tasks. Similar cognitive training could be used in MCI patients as well to improve the strength of the functional connections.

Figures 4.9 and 4.11 showed that using various delays to acquire different functional connectivity arrays for each patient produced new information. In Figure 4.9, delays 2 and 3 provided more discrepant functional connections than delay 1, which would have been the only one obtained without the use of the delays. Future research studies could investigate the accuracy enhancement of using these different delays; analysing this would have gone beyond this research scope.

For the Graph Theory metrics, whole matrix features proved to be of no help for the classification of the MCI patients. Among the 490 best Graph Theory features, no one belonged to the whole matrix category. The Weighted Betweenness Centrality was the Graph Theory metric that had more features among the 100 best features. Eun Hyun Seo and Dong Young Lee concluded that MCI patients had a significantly lower Betweenness Centrality compared with healthy patients [22]. This significantly lower Betweenness Centrality is why so many features were selected from this Graph Theory metric.

Other dominant Graph Theory metrics in this feature analysis were the Vertical and Horizontal degrees. This means that the number of functional connections produced in an MCI brain is different from the number of functional connections produced in a healthy brain.

Having more relevant pixels on a delay does not entail that there will be relevant Graph Theory features for that same delay. This is proved by the fact that the delay from which more relevant Pixel Features are extracted (Figure 4.9), delay number 2, does not match the one from which more relevant Graph Theory features are extracted (Figure 4.11), delay number 1. There is no direct relation between the importance of the single functional connections and the local and global networks formed by those connections.

5.3 Classification of MCI patients

The highest classification accuracy reached for this particular dataset is 96.8% with Naïve Bayes, which is an elevated accuracy value. Other research studies provided similar accuracy values. For example, Chong-Yaw Wee [23] achieved a 96,3% accuracy with an SVM when using structural connectivity added to the functional connectivity analysis. Xiaohong Cui [24] attained a 98.3% accuracy in the classification of MCI vs normal controls using Subnetwork Selection, Graph Kernel Principal Component Analysis and a Minimum Spanning Tree. So, even though the accuracy reached in this research was elevated. It is not outstanding compared to other studies’ accuracies for the classification of MCI patients.

5.4 Limitations

Some limitations were observed during the implementation of this research. First of all, the size of the data. The data consist of relatively few samples, so the results obtained might not have sufficient statistical sigificance. Moreover, because of the small data size, the implementation of one of the most used methods in the state-of-art for the classification of MCI patients, the artificial neural networks (ANN), was not viable. This is due to the deep learning rule of thumb, which says that a minimum sample size of 5.000 labelled samples per class is needed to correctly train the classifier [25]. For this research, only 43 MCI patients and 50 healhty control samples were available.

Another limitation observed is that not all Graph Theory metrics were tested, only the ones that proved relevant in previous research. So, there might be other relevant not tested Graph Theory metrics for the classification of the MCI patients with this particular dataset. Since the procurement of the functional connectivity arrays defines the grade of the importance of the Graph Theory metrics.

5.5 Future Research

Even though this research has answered some questions regarding the classification of MCI patients, there are still some questions left unanswered. Future Research might test other Graph Theories metrics not tested for this specific dataset as well as other Machine Learning classifiers.

Furthermore, in this research, I observed which are the features that are more relevant for the classification between MCI and healthy patients, but the values of these features have not been analysed since doing so could lead to a whole new other research.

Another possible research could be analysing the effect of using different delays. The delay effect has been superficially studied throughout this paper, but more underlying information could be extracted from this, which might provide future standards for the MCI classification using functional connectivity analysis.

Chapter 6

Conclusion

The use of Graph Theory metrics for classifying MCI patients and healthy control patients provides a slightly enhanced performance in the accuracy value. However, for this particular dataset, the use of only Graph Theory metrics does not deliver enough accuracy for the state-of-art standards for this kind of classification.

Among the classifier methods used for the evaluation of using Graph Theory metrics, Naïve Bayes proved to be the best method in terms of accuracy when using Graph Theory metrics in addition to the functional connections as a feature (the ones called Pixel Features through this research). Naïve Bayes resulted in an accuracy of 96.8%, which is not enough to replace previous research Machine Learning classifiers.

In the feature analysis of the Pixel Feature, the right frontal pole and the brainstem showed a high presence of discrepant functional connections between MCI and healthy patients. These ROIs could be potential targets for deep brain stimulation. In addition, the second delay revealed more discrepant functional connections, thereby demonstrating the importance of using different delays in the acquisition of the functional connection arrays.

For the Graph Theory metrics analysis, whole matrix features proved to be of less help than node features for this dataset for the classification of the patients. Weighted Betweenness Centrality, as well as Vertical and Horizontal degree, evidenced to be the Graph Theory metrics with more information for the classification.

Bibliography

1 Leibson, C. L., Long, K. H., Ransom, J. E., Roberts, R. O., Hass, S. L., Duhig, A. M., Smith, C. Y., Emerson, J. A., Pankratz, V. S., & Petersen, R. C. (2015). Direct medical costs and source of cost differences across the spectrum of cognitive decline: A population-based study. *Alzheimer's & Dementia : The Journal of the Alzheimer's Association*, *11*(8), 917. https://doi.org/10.1016/J.JALZ.2015.01.007

2 Vassilaki, M., Cha, R. H., Geda, Y. E., Mielke, M. M., Knopman, D. S., Petersen, R. C., & Roberts, R. O. (2015). Mortality in mild cognitive impairment varies by subtype, sex and lifestyle factors. The Mayo Clinic Study of Aging. *Journal of Alzheimer's Disease : JAD*, *45*(4), 1237. https://doi.org/10.3233/JAD-143078

3 *21 de septiembre: Día Mundial de la Enfermedad de Alzheimer*. (n.d.).

4 Logothetis, N. K., & Pfeuffer, J. (2004). On the nature of the BOLD fMRI contrast mechanism. *Magnetic Resonance Imaging*, *22*(10), 1517–1531. https://doi.org/10.1016/J.MRI.2004.10.018

5 Lv, H., Wang, Z., Tong, E., Williams, L. M., Zaharchuk, G., Zeineh, M., Goldstein-Piekarski, A. N., Ball, T. M., Liao, C., & Wintermark, M. (2018). Resting-State Functional MRI: Everything That Nonexperts Have Always Wanted to Know. *AJNR: American Journal of Neuroradiology*, *39*(8), 1390. https://doi.org/10.3174/AJNR.A5527

6 Connectivity Matrices and Brain Graphs. (2016). *Fundamentals of Brain Network Analysis*, 89–113. https://doi.org/10.1016/B978-0-12-407908-3.00003-0

7 de Vos, F., Koini, M., Schouten, T. M., Seiler, S., van der Grond, J., Lechner, A., Schmidt, R., de Rooij, M., & Rombouts, S. A. R. B. (2018). A comprehensive analysis of resting state fMRI measures to classify individual patients with Alzheimer's disease. *NeuroImage*, *167*, 62–72. https://doi.org/10.1016/J.NEUROIMAGE.2017.11.025

8 Rubinov, M., & Sporns, O. (2010). Complex network measures of brain connectivity: Uses and interpretations. *NeuroImage*, *52*(3), 1059–1069. https://doi.org/10.1016/J.NEUROIMAGE.2009.10.003

9 Bhardwaj, R., Nambiar, A. R., & Dutta, D. (2017). A Study of Machine Learning in Healthcare. *Proceedings - International Computer Software and Applications Conference*, *2*, 236–241. https://doi.org/10.1109/COMPSAC.2017.164

10 Yang, Y. (2017). Ensemble Learning. *Temporal Data Mining Via Unsupervised Ensemble Learning*, 35–56. https://doi.org/10.1016/B978-0-12-811654-8.00004-X

11 *Logistic Regression — Detailed Overview | by Saishruthi Swaminathan | Towards Data Science*. (n.d.). Retrieved March 3, 2022, from

https://towardsdatascience.com/logistic-regression-detailed-overview-46c4da4303bc

12 Mammone, A., Turchi, M., & Cristianini, N. (2009). Support vector machines. *Wiley Interdisciplinary Reviews: Computational Statistics*, *1*(3), 283–289. https://doi.org/10.1002/WICS.49

13 Yang, X., Islam, M. S., & Khaled, A. M. A. (2019). Functional connectivity magnetic resonance imaging classification of autism spectrum disorder using the multisite ABIDE dataset. *2019 IEEE EMBS International Conference on Biomedical and Health Informatics, BHI 2019 - Proceedings*. https://doi.org/10.1109/BHI.2019.8834653

14 Wang, Z., Qiao, K., Chen, G., Sui, D., Dong, H. M., Wang, Y. S., Li, H. J., Lu, J., Zuo, X. N., & Han, Y. (2019). Functional connectivity changes across the spectrum of subjective cognitive decline, amnestic mild cognitive impairment and alzheimer's disease. *Frontiers in Neuroinformatics*, *13*, 26. https://doi.org/10.3389/FNINF.2019.00026/BIBTEX

15 Khazaee, A., Ebrahimzadeh, A., & Babajani-Feremi, A. (2016). Application of advanced machine learning methods on resting-state fMRI network for identification of mild cognitive impairment and Alzheimer's disease. *Brain Imaging and Behavior*, *10*(3), 799–817. https://doi.org/10.1007/S11682-015-9448-7/FIGURES/11

16 Chen, C. H., Tanaka, K., Kotera, M., & Funatsu, K. (2020). Comparison and improvement of the predictability and interpretability with ensemble learning models in QSPR applications. *Journal of Cheminformatics*, *12*(1), 1–16. https://doi.org/10.1186/S13321-020-0417-9/FIGURES/8

17 Kim, T. K. (2017). Understanding one-way ANOVA using conceptual figures. *Korean Journal of Anesthesiology*, *70*(1), 22. https://doi.org/10.4097/KJAE.2017.70.1.22

18 Andreotti, J., Jann, K., Melie-Garcia, L., Giezendanner, S., Dierks, T., & Federspiel, A. (2014). Repeatability Analysis of Global and Local Metrics of Brain Structural Networks. *Https://Home.Liebertpub.Com/Brain*, *4*(3), 203–220. https://doi.org/10.1089/BRAIN.2013.0202

19 Sánchez-Pérez, N., Inuggi, A., Castillo, A., Campoy, G., García-Santos, J. M., González-Salinas, C., & Fuentes, L. J. (2019). Computer-Based Cognitive Training Improves Brain Functional Connectivity in the Attentional Networks: A Study With Primary School-Aged Children. *Frontiers in Behavioral Neuroscience*, *13*, 247. https://doi.org/10.3389/FNBEH.2019.00247/BIBTEX

20 Liu, J., Zhang, B., Wilson, G., & Kong, J. (2019). New perspective for non-invasive brain stimulation site selection in mild cognitive impairment: Based on meta-and functional connectivity analyses. *Frontiers in Aging Neuroscience*, *11*(AUG), 228. https://doi.org/10.3389/FNAGI.2019.00228/BIBTEX

21 *Inside the Controversy Around Biogen's Alzheimer's Drug, Aduhelm | Time*. (n.d.). Retrieved May 8, 2022, from https://time.com/6081333/biogen-alzheimers-drug-aduhelm-fda-controversy/

22 Seo, E. H., Lee, D. Y., Lee, J. M., Park, J. S., Sohn, B. K., Lee, D. S., Choe, Y. M., & Woo, J. I. (2013). Whole-brain Functional Networks in Cognitively Normal, Mild Cognitive Impairment, and Alzheimer's

Disease. *PLoS ONE*, *8*(1). https://doi.org/10.1371/JOURNAL.PONE.0053922

23 Wee, C. Y., Yap, P. T., Zhang, D., Denny, K., Browndyke, J. N., Potter, G. G., Welsh-Bohmer, K. A., Wang, L., & Shen, D. (2012). Identification of MCI individuals using structural and functional connectivity networks. *NeuroImage*, *59*(3), 2045–2056. https://doi.org/10.1016/J.NEUROIMAGE.2011.10.015

24 Cui, X., Xiang, J., Guo, H., Yin, G., Zhang, H., Lan, F., & Chen, J. (2018). Classification of alzheimer's disease, mild cognitive impairment, and normal controls with subnetwork selection and graph kernel principal component analysis based on minimum spanning tree brain functional network. *Frontiers in Computational Neuroscience*, *12*, 31. https://doi.org/10.3389/FNCOM.2018.00031/BIBTEX

25 Goodfellow, I., Bengio, Y., & Courville, A. (2016). *Deep Learning*. MIT Press. http://www.deeplearningbook.org

26 Welton, T., Kent, D. A., Auer, D. P., & Dineen, R. A. (2015). Reproducibility of Graph-Theoretic Brain Network Metrics: A Systematic Review. *Brain Connectivity*, *5*(4), 193. https://doi.org/10.1089/BRAIN.2014.0313

27 *The Swedish BioFINDER Study - Full Text View - ClinicalTrials.gov*. (n.d.). Retrieved May 18, 2022, from https://clinicaltrials.gov/ct2/show/NCT01208675

Appendix A

Appended Material

A.1 ROIs

Table A.1: Corresponding regions of the brain to the numbers of the arrays.

Number	ROI
1lIOG	Inferior Occipital G
2lANG	Angular G
3lPRECU	Precuneus
4lINS	Insula
5lANT	Anterior cingulate
6rPRECU	Precuneus
7rANG	Angular G
8lPOST	Postcentral G
9lCEREB	Cerebellum
10rCEREB	Cerebellum
11lMTG	Middle temporal G
12rFP	Frontal pole
13rMEDFG	Medial Frontal G
14rANG	Angular G
15rPUT	Putamen
16lIOG	Inferior Occipital G
17rPRECE	Precentral
18rTHA	Thalamus
19lPRECU	Precuneus
20lINS	Insula
21lPOST	Postcentral G

22rANT	Anterior cingulate
23lMFG	Middle frontal G
24lSTG	Superior temporal G
25rMFG	Middle frontal G
26rIOG	Inferior Occipital G
27lPARAH	Parahippocampal G
28rPOST	Postcentral G
29rCING	CINGULATE G
30rBRAINSTEM	Brainstem
31rSPG	Superior Parietal Lobule
32rTP	Temporal pole
33lSUPRAM	Supramarginal G
34lPRECE	Precentral
35rINS	Insula
36lCEREB	Cerebellum
37lTHA	Thalamus
38rMFG	Middle frontal G
39rMTG	Middle temporal G
40rANT	Anterior cingulate
41rCEREB	Cerebellum
42lFP	Frontal pole
43lTP	Temporal pole
44rOP	Occipital pole
45lTHA	Thalamus
46rPCG	Posterior Cingulate
47lCAU	Caudate
48rPARAH	Parahippocampal G
49rMTG	Middle temporal G
50lSFG	Superior frontal G
51lANT	Anterior cingulate
52lBRAINSTEM	Brainstem
53rFOG	Frontal Orbital G
54lCUN	Cuneus
55lANT	Anterior cingulate
56lANG	Angular G
57lFOG	Frontal Orbital G
58rPRECU	Precuneus
59rINS	Insula
60rPOST	Postcentral G
61lMFG	Middle frontal G
62rFUS	Fusiform
63lFUS	Fusiform
64rMFG	Middle frontal G
65rSPG	Superior Parietal Lobule

66rSTG	Superior temporal G
67lPUT	Putamen
68rMIDB	Midbrain
69rMTG	Middle temporal G
70lLING	Lingual G
71rFOG	Frontal Orbital G
72lMTG	Middle temporal G
73lPRECE	Precentral
74lFOG	Frontal Orbital G
75rFP	Frontal pole
76lPCG	Posterior Cingulate
77rCEREB	Cerebellum
78lTP	Temporal pole
79lANT	Anterior cingulate
80lCEREB	Cerebellum
81rCUN	Cuneus
82lANG	Angular G
83rSTG	Superior temporal G
84lCAU	Caudate
85rIOG	Inferior Occipital G
86rBRAINSTEM	Brainstem
87rFUS	Fusiform
88lPOST	Postcentral G
89rLING	Lingual G
90lPRECE	Precentral
91lMEDFG	Medial Frontal G
92lAMY	Amygdala
93rSUPRAM	Supramarginal G
94rCAU	Caudate
95lSFG	Superior frontal G
96lPOST	Postcentral G
97lSOG	Superior Occipital G
98rPRECE	Precentral
99lITG	Inferior Temporal G
100rITG	Inferior Temporal G
101lITG	Inferior Temporal G
102rIOG	Inferior Occipital G
103lCEREB	Cerebellum
104lFP	Frontal pole
105rLING	Lingual G
106rMFG	Middle frontal G
107rMTG	Middle temporal G
108lLING	Lingual G
109rFP	Frontal pole

110rTP	Temporal pole
111rPRECE	Precentral
112lFOG	Frontal Orbital G
113rMFG	Middle frontal G
114lSOG	Superior Occipital G
115rPRECE	Precentral
116lPOST	Postcentral G
117lMTG	Middle temporal G
118rCEREB	Cerebellum
119rIFG	Inferior Frontal G
120lCEREB	Cerebellum
121lINS	Insula
122lPARAH	Parahippocampal G
123lPARAC	Paracentral lobule
124rFP	Frontal pole
125lMFG	Middle frontal G
126rCEREB	Cerebellum
127rMFG	Middle frontal G
128rIPG	Inferior Parietal G
129lSTG	Superior temporal G
130lMIDB	Midbrain
131lIOG	Inferior Occipital G
132rSPG	Superior Parietal G
133lSFG	Superior frontal G
134rPRECE	Precentral
135rCAU	Caudate
136lSOG	Superior Occipital G
137rINS	Insula
138rFUS	Fusiform
139rFP	Frontal pole
140rMTG	Middle temporal G
141lIFG	Inferior Frontal G
142rMOG	Middle Occipital G
143rIOG	Inferior Occipital G
144rIFG	Inferior Frontal G
145lPARAH	Parahippocampal G
146lSTG	Superior temporal G
147lPRECU	Precuneus
148rTHAL	Thalamus
149lMEDFG	Medial Frontal G
150lIOG	Inferior Occipital G
151lMFG	Middle frontal G
152rCEREB	Cerebellum
153rSTG	Superior temporal G

154lPOST	Postcentral G
155rPARAH	Parahippocampal G
156lSPG	Superior Parietal Lobule
157rPOST	Postcentral G
158lOP	Occipital pole
159lFUS	Fusiform
160lANT	Anterior cingulate
161rSFG	Superior frontal G
162lCEREB	Cerebellum
163rPRECU	Precuneus
164rIFG	Inferior Frontal G
165lPRECE	Precentral
166rANG	Angular G
167lIFG	Inferior Frontal G
168rMFG	Middle frontal G
169lIFG	Inferior Frontal G
170rSOG	Superior Occipital G
171lSPG	Superior Parietal Lobule
172rFUS	Fusiform
173lSFG	Superior frontal G
174lPRECU	Precuneus
175rIOG	Inferior Occipital cortex
176rCEREB	Cerebellum
177lLING	Lingual G
178rPUT	Putamen
179lLING	Lingual G
180rPOST	Postcentral G
181rTP	Temporal pole
182lSUPP	Supplementary motor cortex
183lFP	Frontal pole
184lINS	Insula
185rTTG	Transverse Temporal G
186rSFG	Superior frontal G
187rSFG	Superior frontal G
188lSPG	Superior Parietal Lobule
189lFUS	Fusiform
190rBRAINS	Brainstem
191lSFG	Superior frontal G
192lCEREB	Cerebellum
193rSFG	Superior frontal G
194rMIDB	Midbrain
195rLING	Lingual G
196lINS	Insula
197lPRECU	Precuneus

198rFUS	Fusiform
199lBRAINS	Brainstem
200lTTG	Transverse Temporal G