SAPIENZA
UNIVERSITÀ DI ROMA

Empirical survey on graph classification methods for Brain Networks

Faculty of Information Engineering, Informatics and Statistics
M.Sc. in Data Science

Academic Advisor
Prof. Aris Anagnostopoulos

External Advisor
Dr. Francesco Bonchi

Candidate
Ilaria Taddei
ID 1853733

Academic Year 2020/2021

Abstract

Our brain, as our entire body, is a perfect as complicated machine. For this reason is important to analyse it in an effective and efficient way. An important role in this purpose is given to machine learning and similar algorithmic methods. In fact, there is a wide literature covering the field of implemented methods for brain data. In particular, one fundamental task is the following: brain is represented as a graph through its fMRI, and from that representation the goal is to recognize if a person is affected by a psychiatric disorder, such as Autism or Schizophrenia, or not.

In this work, the aim is to study and compare some of those methods. First of all is important to have in mind which are the basic elements with which the papers work, such as graphs, classification and brain connectomes. Then, the algorithms taken in consideration can be classified according to the technique used, like Neural Networks or features embeddings, so, some methods of different techniques are described. For each method an experimental analysis has been performded, to see how they work, in which case are more useful, and how good are the results. Wxperiments are made with different datasets, to see how the methods adapt and how general they are.

Future works can be implemented, including more state-of-the-art algorithms and crafting new methods that could be easily used by neuroscientists.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

Neuroscientists, to diagnose a psychiatric disorder, use symptom scores from clinical interviews. For a definitive validation, could be useful to study the interactions between brain regions, in order to see the different behaviours of these interactions in a brain of Typical Developed people and people with disorders. This could be seen through Magnetic Resonance Imaging (MRI), in particular functional MRI (fMRI). The neuroimaging data produced are then pre-processed and transformed in data structures that an algorithm can study, in our cases graphs, matrices and time series. From the study of these interactions, the goal is to implement a classifier of the brain representation, able to says whether the brain we give in input is affected by a specific psychiatric disorder. This is called **brain network classification**. Also, through Machine Learning algorithms, we could analyse better brain data. This could be useful to comprehend the causes of the psychiatric disorder, that from the simple interview of neuroscientists could not emerge.

The studies taken in consideration, other than in techniques, differs also in the data that have in input. The inputs could be either graphs, matrices, time series and even images. This work is focused only on methods that take in input graphs, for which many methods have been implemented for their classification. In our particular field, graph classification of brain networks, becomes important to take into account the structure of the graph, because there could be many brain regions that characterize a specific psychiatric disorder. We will see how in some methods are also studied those regions to identify the once that have different interactions in people with disorders, even if the described experiments of this works are concentrated on the simple classification of brain networks.

This thesis contains an explanation of the basics with which the study works, then a description of several methods on this subject and then experiments with some of these methods that more represent our field.

This work was performed during a research internship at ISI Foundation under the supervision of Dr. Francesco Bonchi.

1.1 Outline

The thesis is organized as follows:

- In the current chapter there is an explanation of some basics related to this work.
- Chapter 2 contains the classification of the several methods taken in account, so, an overview of the macro-classes of all the implemented techniques to do *graph* and *brain network classification*. Then, for each method, there is an explanation of the main characteristics.
- Chapter 3 contains the experimental part. Are described the set-up of the experiments, like the datasets and the kind of classification, the results of the experiments and a discussion of the results.
- Chapter 4 contains comments about the use of each method and possible future works.

1.2 Background

To understand better all the methods, we should do some explanation of what we will encounter during the lecture. The main arguments we should consider are **Graphs**, **Classification** and **Connectomes**.

Graphs

There are many subjects in which graphs are used, and consequently many definitions. In mathematics, a graph is a structured set of objects that, in pairs, could have a relation between themselves. Each object is called a *vertex*, and each relation between a pair of vertices is called an *edge*. So, a graph $G(V, E)$ is a pair where V is the set of vertices and E is the set of edges. Graphs are represented in diagram form, where vertices are in circle form and edges are lines that join pairs of vertices with a relation (Figure 1.1).

Graph could be *directed* or *undirected*. In a direct graph (Figure 1.2), edges have an orientation, the links between vertices can be represented by arrows going from one vertex to the other. In an edge (x, y) directed from x to y, the vertices are called respectively *tail* and *head* of the edge.

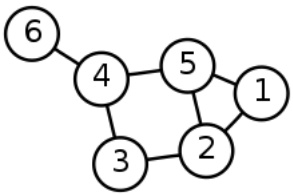

Figure 1.1: Representation of a graph.

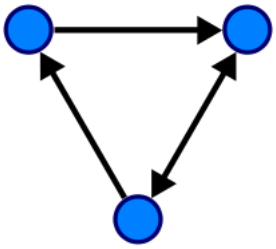

Figure 1.2: Representation of a directed graph

Graphs can be represented by matrices, the matrix representations in which we are interested for this thesis are **Adjacency matrix** and/or **Connectivity matrix**.

An **Adjacency matrix** is a square matrix that represents if there is an edge between two vertices. In particular in an unweighted graph with a vertex set $U = \{u_1, ..., u_n\}$ the Adjacency matrix A is an $n \times n$ matrix in which its element A_{ij} is 1 if there is an edge between the vertices u_i and u_j, because they have no specific weight in each edge. If the graph is weighted the values in the matrix are not 1s and 0s, but depends on the weight of each edge. In fact, a weighted graph is a graph where each branch (or edge) is given a numerical weight (or label). Also, if the graph is undirected the Adjacency matrix is symmetric.

Classification

In machine learning, *Classification* is a supervised learning approach in which an algorithm is trained, with some given data, to classify new observations. It is a process. In particular classification predictive modeling is the task of approximating a mapping function $f(X)$ from input variables X to discrete output variables y ([4]).

First we have some data, of which we already know the belonging class, called *labelled* data. We could have two classes, in which case we will have a *binary classification*, or multiple classes, so *multi-class classification*. Our

labelled data are given to the classification model, starting the **training** of the model. In this step the classifier uses the training data to how the given input data relate to the class.

Once trained the classificator, we can give to it new data that we want to classify. This is called the **prediction**. In classification there are two types of learners, *Lazy learners* and *Eager leraners*. Lazy learners simply store the training data and wait until a testing data appear. When it does, classification is conducted based on the most related data in the stored training data. They have less training time but more predicting time. An example of lazy learner is k-nearest neighbour. Eager learners construct a classification model based on the given training data before receiving data for classification. It must be able to commit to a single hypothesis that covers the entire instance space. They take more time in training time and less in predicting time. Some examples of eager learners are Decision Tree, Naive Bayes, Artificial Neural Networks.

There are many classification algorithm, it depends on the dataset which one is better to apply.

Finished the prediction we can evaluate our model through some scores, the main one in our case is the **accuracy**, that calculates the percentage of how many predictions of our classifier are right. Others are Precision and Recall and the ROC curve. To evaluate the model there are also the **loss functions**. They calculate the distance between the prediction of the algorithm and its expected output. There are many measures of loss functions, the most common are the *cross-entropy*, the *Log loss* and the *Mean Square Error* (MSE). Cross-entropy and Log loss function compute the difference between two probability distribution functions. The MSE is the square difference between the current output (y_pred) and the expected output (y_true) divided by the number of output.

Connectomes

The **connectome** is the connection matrix of the human brain [23]. From the anatomic point of view, the connectome is defined by all the axonal origins, terminations and trajectories of all the brain neurons, through the brain regions. This means, the connectivity of neural pathways in the brain.

Being able to have a representation of the human brain allows us to understand fundamental cognitive operations, brain activities, conditional structure-function models of the brain, so, consequently, to understand and

detect brain deceases. From this powerful tool, we can see how brain physiology is correlated to abilities and behaviours, underlying mental anomalies and pathology. This is fundamental to develop treatments ad hoc for each pathology and case.

Other fields, that we do not cover here, but that are very interesting, concern the memories (in fact neuroscientists believe that our memories are stored in the synapses between neurons) and the preservation of the brain (being able to recover the structure and connections of it) [8].

In Figure 1.3 we can see an example of a connectome.

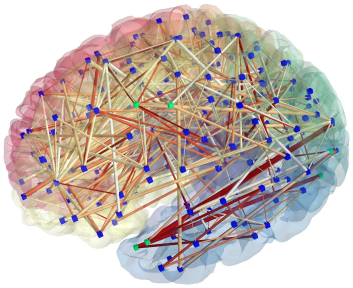

Figure 1.3: Example of a connectome

1.3 Related works

Here we concentrate the study of connectomes. This means that the fMRI of a brain, is elaborated in a graph form, that is called connectome. To make easier the study of the brain, the graph is studied in matrix form (Figure 1.4). The graph have as nodes the brain regions, and as edges the connections between them. It follows that the matrix is an *Adjacency matrix*, in which the values represent the degree of connections between the regions, and each row and each column is a node of the graph, a so called *Region of interest* (**ROI**).

Graph Classification

The technique that studies the classification of graphs is called **Graph Classification**. A basic definition of graph classification is given in [19]: given a set of graphs $\mathcal{B} = \{(\mathcal{G}_1, \ell_1), (\mathcal{G}_2, \ell_2), \cdots, (\mathcal{G}_n, \ell_n)\}$ the aim is to learn a function $f : \mathbf{G} \to \mathcal{L}$, where $\mathbf{G}$ is the input space of graphs, and $\mathcal{L}$ is the

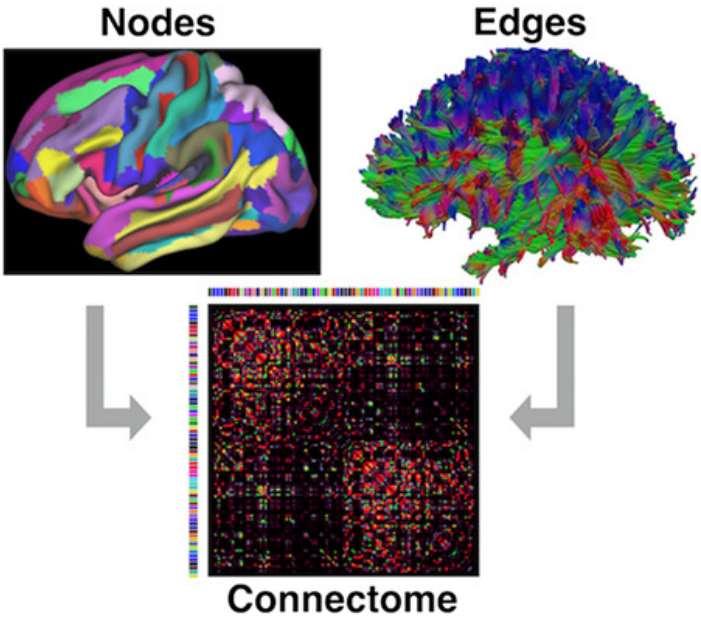

Figure 1.4: Different representations of a brain. Up left: nodes of a connectome. Up right: edges of a connectome. Bottom: Correlation matrix of a connectome.

set of graph labels. Each graph $\mathcal{G}_i = (\mathbf{A}_{\mathcal{G}_i}, \mathbf{B}_{\mathcal{G}_i})$ has an adjacency matrix $\mathbf{A}_{\mathcal{G}_i} \in \{0,1\}^{N_i \times N_i}$ and an attribute matrix $\mathbf{B}_{\mathcal{G}_i} \in \mathbf{R}^{N_i \times N_i}$, where N_i is the number of nodes of the i-graph and B is the number of attributes. Each graph has also a corresponding label l_i.

The usual strategy to study graphs is to calculate graph statistics on the entire graph. A popular technique is to count the occurrences of various *subgraphs* on a graph, called graphlet kernel [22]. Then there is the *Morgan algorithm* [20] that consists in an iterative process, that updates the attributes vector of each node by hashing a concatenation of all the attributes of in the node's local neighbourhood. Then from the final attributes of all the nodes in the graph is computed the graph feature. Recently, learning data-driven graph features [11] is becoming more important. This means that given a dataset, the task-relevant features are learned automatically from the graphs. Once we extract these features, independently of which method we would like to use, we use them for the classification. Another method that we can mention is all the literature that regards *Graph Neural Networks* (GNN), a deep learning technique. To build a GNN we need to find out the graph structure, the graph type and its scale, then we should define the loss function, depending on the task, and then we are ready to build a model using computational model [29].

These methods brings us to the field of our interest, **Brain Network Classification**. For this task, we will show that the state-of-the-art is mostly made by methods based on the previous graph classification techniques, but specifically adapted for a more suitable result. In the next chapter we will discuss various methods that inspired this work.

Chapter 2

Survey

2.1 All methods

In this chapter there will be a wide description of the state-of-the-art about *brain network classification*. For sake of clarity, we divide the methods in 3 different classes: Deep Learning, Statistical Fingerprints and Machine Learning. In order to compare them, we have performed an experimental analysis, whose results are described in Chapter 3.

2.1.1 Deep Learning

GroupINN: Grouping-based Interpretable Neural Network for Classification of Limited, Noisy Brain Data

This state-of-the-art of Yan Y. et al [28] proposes a grouping-based interpretable neural network model, GroupINN, that classifies cognitive performance with 85% fewer parameters than baseline deep models, while also identifying the most predictive brain subnetworks within several task-specific contexts. In the design of the neural network is included the idea of node grouping. In this way the model learns the node grouping and extracts the graph features jointly.

The problem statement is: given a set of subjects, each with corresponding fMRI data and a label associated with a certain phenotype, we seek to devise an efficient, interpretable, and parsimonious neural network model that can predict each phenotype with high accuracy.

To reduce the number of parameters used in the model, they adopted the idea of multi-graph clustering (where the goal is to find a common clustering across multiple graphs) to summarize the original graph into a supergraph with each cluster as a supernode.

The neural network is formed by three different types of layers: node grouping layer, graph convolutional layer and fully connected layer. The node grouping layer is designed to hide the non-indicative (noisy) edges by grouping them into a cluster, thus highlighting the indicative edges: two nodes are assigned to different groups if their connection is identified as important. Graph convolutional layers are used to capture the structure of the supergraph.

The neural network is also divided in two branches, one processes the positive graphs and one the negative ones. All in all, the architecture consists of three kinds of layers and two branches. The input graph is the correlation matrix W. The first layer is a dimensionality reduction layer and the output is a matrix W^s representing the supergraph. Following the dimensionality reduction layer, three graph convolutional layers are used. At last, the positive and negative outputs of the previous layer are concatenated, flattened and sent to the fully connected layer (with softmax activation). The design of the model is represented in Figure 2.1.

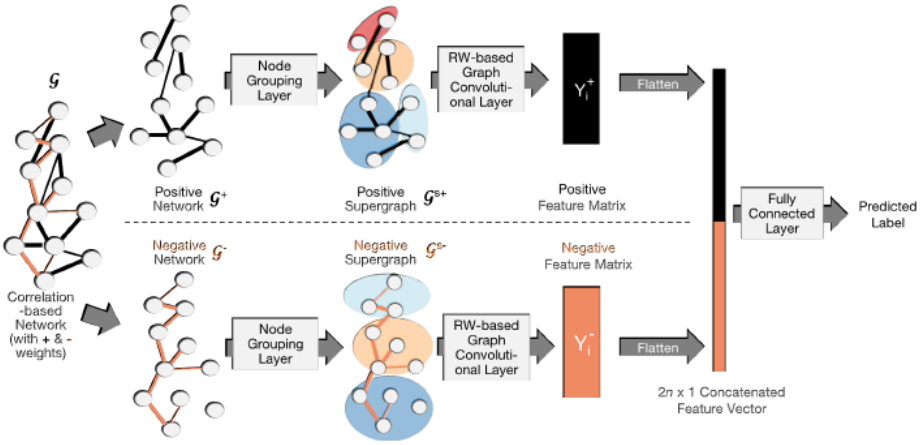

Figure 2.1: Structure of GroupINN Neural Network.

Regarding the experimentation part, they used a dataset taken from Human Connectome Project 1200 (HCPt) [24]. This dataset consists of 966 subjects to which has been measured brain activity, through fMRI, while they were performing specific tasks. The four task-based datasets used in this experiment are: *Emotion, Gambling, Social* and *Working Memory*. It is divided in 90% train/validation set and 10% testing set. For the evaluation they take in consideration *accuracy* and the *runtime*. Comparing their method, they found out that it is faster and with less parameters than other

works, so it is more interpretable, as well as having good accuracy.

Deep Learning-based Pipeline to Recognize Alzheimers Disease using fMRI Data

S. Sarraf et al. [21] built a Neural Network, specifically a Convolutional Neural Network (**CNN**) for classification of clinical data, in particular they tested it for Alzheimer disease. Their dataset is composed of a group of people affected by Alzheimer and a control group. They where scanned with resting-state FMRI, and after a preprocess, the data consisted of images of functional information.

CNNs in general are used to study images, and are composed of Pooling layers, Normalization layers, Fully Connected layers and Convolutional layers. Convolutional layers help to maintain the spacial order of the input they are working on, obviously very important on images, and these layers consist on filters applied to these images. In this case they used an already implemented CNN called LeNet-5 (by Y. LeCun et al [18]), and adjusted it for fMRI data. The data in input were 2D images, that were labeled for binary classification, as shown in Figure 2.2.

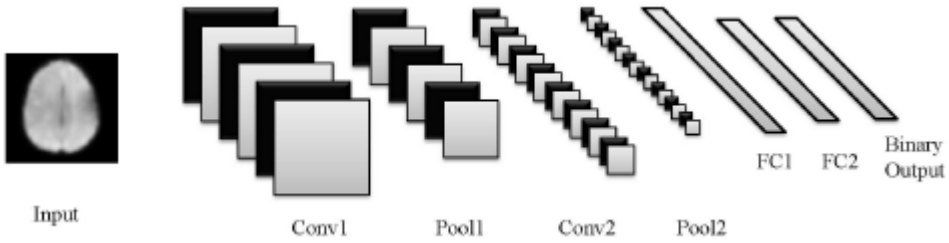

Figure 2.2: CNN LeNet-5 structure adjusted for fMRI input data.

The experiment ended up with a very high accuracy (96.8588%) and a very low learning rate, meaning that it is a valid tool, even if on one hand it is a very complicated model, has many parameters and hyperparameters to tune and could have GPU memory problems.

Functional Brain Network Classification for Alzheimers Disease Detection with Deep Features and Extreme Learning Machine

X. Bi et al [7] designed two deep learning methods of functional brain network classification. More precisely they concentrate their work on Alzheimer disease detection. The first model is a Convolutional learning method, that

learns the deep regional-connectivity features. The second is a Recurrent learning method, learning deep adjacent positional features. So, both the learning methods are Neural Networks. They also implemented an Extreme Learning Machine (ELM) to improve the learning ability, and is implemented in the learning methods.

Both the deep learning methods take as input a graph matrix, i.e. the adjacency matrix of each patient. The Convolutional Learning method (Figure 2.3) is composed of a Convolutional layer, Activation function, Pooling layer, Fully Connected layer and a Decision layer. It is in the Convolutional layer that the features are extracted, while in the decision layer, with a *Softmax* function, are generated the labels of each brain network.

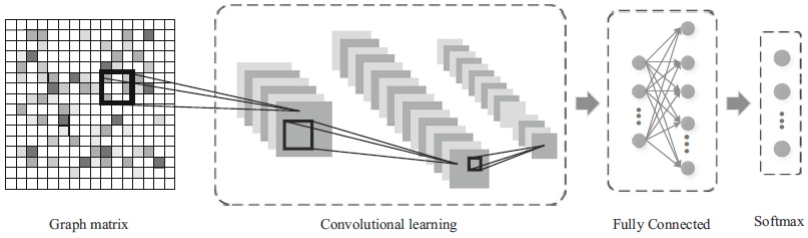

Figure 2.3: Structure of the Convolutional Learning method.

To the Recurrent learning method (Figure 2.4) is given as input a row o more of the graph matrix at each time step, until all the rows are learned. It is mainly composed of two parts, the recurrent structure and the classification structure.

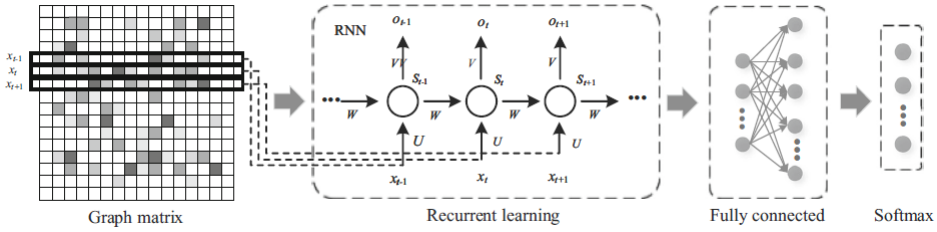

Figure 2.4: Structure of the Recurrent Learning method.

The complexity of these two models is in the fully connected layer, where there is an high computation for the parameters tuning. For this reason is built the ELM layer. It is much faster and gives good generalization performance. So, the deep features are extracted, convolutional or recurrent ones,

and are given to the ELM layer, that produces the output labels (Figure 2.5).

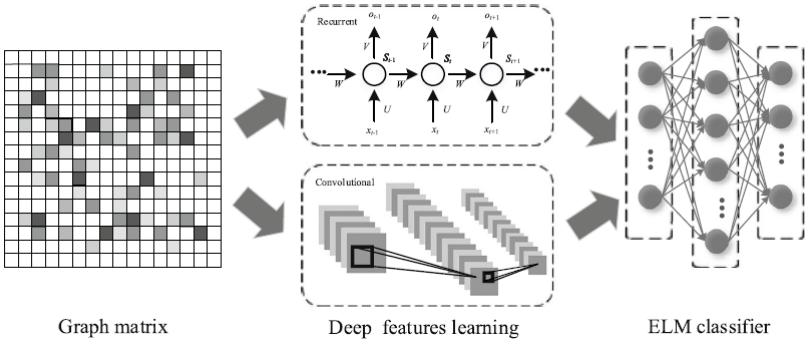

Figure 2.5: Structure of ELM Classifier with both learning methods.

For the experiments they compared their methods with shallow methods. They saw that, with ELM, both neural networks performed better than without ELM, but ELM also brings performance fluctuation. Anyway, they performed much better than shallow methods. Recurrent method, in particular, is slightly better that the Convolutional one, but takes much more training time. Instead, Convolutional learning method, having more parameters to tune, reaches with more difficulty the optimal performance.

2.1.2 Statistical Fingerprints

Explainable Classification of Brain Networks via Contrast Subgraphs

In this work of Bonchi F, Lanciano T. and Gionis A. [17] they introduce an approach for classifying brain networks based on extracting contrast subgraphs, i.e., a set of vertices whose induced subgraphs are dense in one class of graphs and sparse in the other. The model is extremely simple, with just one parameter, excellent interpretability and good classification accuracy. What they want to improve or add, differently from others methods, are the node-identity awareness, black box effect and high number of parameters. With **node-identity awareness** is meant to take in consideration that a specific vertex corresponds to the same ROI in all the input graphs. This is very important to find similarities among the input networks. The majority of the models have a **black box effect**, meaning that are complicated to understand how to use them, and their parameters. It should be crucial to make it understandable for neuroscientists that need these tools. Even the **high number of parameters** could be a problem, for overfitting and for tuning them.

In their experimentation, each individual is represented by an undirected unweighted graph with $|V| = 116$ vertices, that are the ROIs. They propose two problems regarding contrast-subgraphs, but in the experiments chapter 3 we will employ only the first one. *Problem 1* states that given the observations of condition group - affected by autism - and control group, and the corresponding summary graphs, they seek to find a subset of vertices that maximizes the contrast-subgraph objective, so to find a set of vertices whose induced subgraph is dense in the summary graph of $G^{\mathcal{A}}$ - condition group - and sparse in summary graph $G^{\mathcal{B}}$ - control group. It can be summarised in the following equation:

$$\delta(S) = e^{\mathcal{A}}(S) - e^{\mathcal{B}}(S) - \alpha \begin{pmatrix} |S| \\ 2 \end{pmatrix} \tag{2.1}$$

where $e^{\mathcal{A}}(S)$ and $e^{\mathcal{B}}(S)$ correspond to the number of edges in the subgraph induced by S in the summary graphs $G^{\mathcal{A}}$ and $G^{\mathcal{B}}$, and α is a parameter that penalize large size solutions: larger the value of α, smaller is the optimal contrast-subgraph. The summary graphs $G^{\mathcal{A}}$ and $G^{\mathcal{B}}$ are undirected and weighted graphs, defined over the same set of vertices V as the original observation graphs. A summary graph of a collection of graphs is a single graph where each node (edge) represents at most one node (edge) from each of the graphs it summarizes [15].

Problem 2 is a symmetric variant, wanting to find a subgraph having the largest absolute difference of edge weights between $G^{\mathcal{A}}$ and $G^{\mathcal{B}}$, so that maximize the contrast-subgraph.

As said, in this experiment they want to classify people with autism and typically developed people. For this reason they concentrate on the contrast-subgraph ASD-TD is a subgraph dense in the class of ASD and sparse in TD, and the contrast-subgraph TD-ASD is a subgraph dense in TD and sparse in ASD. Calculating constrast-subgraph TD-ASD and ASD-TD from a dataset they used to make experiments, they were able to observe differences between the two graphs.

They ended up with Figure 2.6, where the top left image is the TD-ASD contrast-subgraph, while the bottom left image is the ASD-TD contrast-subgraph. We can clearly see the patterns differences between the two contrast-subgraphs. Also, from the right part of Figure 2.6, we have a scatterplot with number of edges induces by TD-ASD - x axis - and ASD-TD - y axis - for each patient. With these results they ended up with some important rules:

- If an individual exhibits more than 62 edges among the 15 vertices of

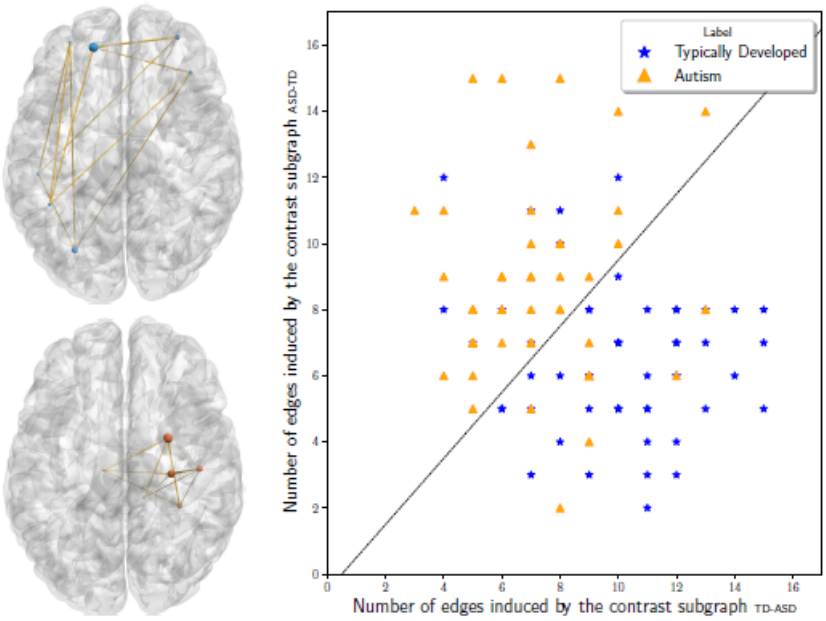

Figure 2.6: Top left: TD-ASD Contrast Subgraph. Bottom left: ASD-TD Contrast Subgraph. Right: scatterplot with at x-axis the number of edges induced by TD-ASD, and at y-axis the number of edges induced by ASD-TD, for each patient.

the contrast subgraph ASD-TD, then there are high chances that the individual is affected by ASD;

- If the number of edges induced by the contrast subgraph ASD-TD is smaller than half of the number of edges induced by the contrast subgraph TD-ASD, then there are high chances that the individual is not affected by ASD;
- If the number of edges induced by the contrast subgraph ASD-TD is smaller than the number of edges induced by the contrast subgraph TD-ASD, then there are high chances that the individual is affected by ASD.

They used these two features to make classification, based on SVM, and compared the results with other methods of the literature. At the end they have a single parameter α, a very low run-time (less than 30 seconds to extract the constrast-subgraph), a simple explainability, having only two simple features, and high accuracy.

Unsupervised Network Embedding for Graph Visualization, Clustering and Classification

A crucial challenge in mining network-based data is to find effective ways to represent or encode graph structures, in order to make themm efficiently exploited by Machine Learning algorithms. L. Gutirrez et al [12] provide an unsupervised approach to learn embedding representation for a collection of graphs, to use in graph mining tasks. They use an unsupervised neural network on graphs that aims to capture the distribution of the data, to discriminate between different class of networks. With their method, they learn automatically a feature representation of graphs assessing their similarity on an Euclidean space (Figure 2.7), focusing on problems defined on networks that account for *node identity*. They evaluate the method in three network mining tasks: graph clustering, graph classification and visualization. We are more interested in graph classification.

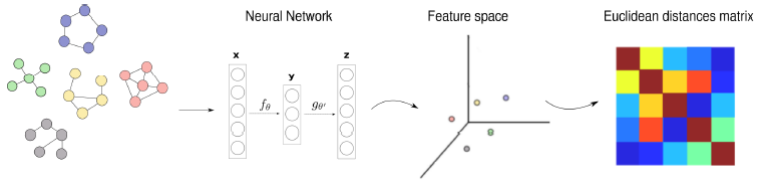

Figure 2.7: Feature representation of graphs on an Euclidean space.

Unlike classical distances in the literature, such as *Hamming* and *Jaccard* distances, this approach performs network comparisons directly on a feature space, through a learned non-linear mapping applied to input graphs. It is composed by some blocks. The first is the **Autoencoder**, that is one of the most popular unsupervised neural network approaches. Unsupervised approaches aim to uncover hidden patterns or learning representations from unlabeled data. In particular, the autoencoder allows to compress the representation of input data, removing redundancy and reducing the dimension of the input. A traditional autoencoder learns a non-linear mapping which encodes an input example in a smaller dimensional latent vector. Unfortunately, this method could just learn the training data, meaning that it will not work on unknown data. For this reason they train a **Denoising Autoencoder** (**DAE**) (Figure 2.8), that reconstruct a clean or repaired version from corrupted input.

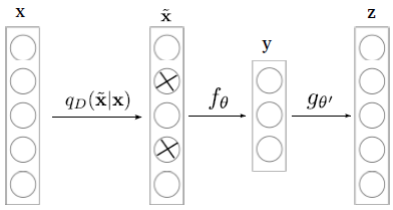

Figure 2.8: Structure of the Denoising Autoencoder (DEA).

Giving in input an adjacency matrix to the DEA could be insufficient, so they compute higher powers of the matrix to capture multiple path relationships. Also, this method remains invariant to the node ordering when the same node permutation is assigned to the graph, having a collection of networks with node correspondence across graphs. A main advantage of transforming graphs into feature vectors is that it allows to compare easily networks computing only Euclidean distances between their embeddings.

For the experiment part, we will see the graph classification results. Their task is to classify connectomes according to gender, male or female. The input is a dataset built from MRI, structural and diffusion, so a collection of graphs. The two steps of the model are learning graph embedding through DAE and computation of a pairwise Euclidean distance matrix. Comparing their method with classic ones of the literature, they saw that it outperformed them, at accuracy level, remaining competitive only with DeltaCon model. Regarding the runtime, it is much faster than all the others.

Supervised classification of structural brain networks reveals gender differences

Another work base on statistical fingerprint is the one of Chiem B. et al [10]. The work aims to study individual differences in the structural connectome, not the functional one, with perfect node correspondence property. This property means that each node corresponds to the same anatomical location in each connectome. They propose three new methods based on SVM.

The first contribution regards the feature extraction, crucial point for the classification part. They introduced the **Bag-of-Edges**, it consists in the application of the *Recursive Feature Eliminatio* (RFE). It trains an SVM with linear kernel, sorts features according to the weights granted by the SVM, and reduces the size of the feature vector keeping only a percentage of the most discriminative features. It stops when a given number of features is reached (Figure 2.9). At the end we have a feature vector with only the most discriminative edges of the graph.

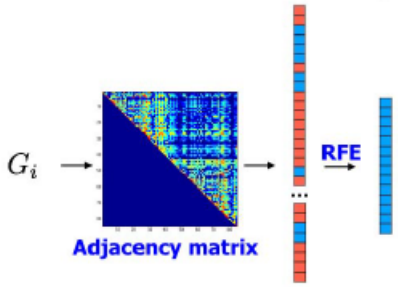

Figure 2.9: Structure of the Bag-of-Edges.

To take advantage of perfect node property they designed two new graph kernels. First they used the **DeltaCon Kernel**, introduced by Koutra et al [16], but never used as a kernel. As first step they compute for each graph a node-to-node affinity matrix, that encodes a particular measure of similarity between nodes of the graph. The similarity measure taken in consideration is Fast Belief Propagation (FaBP). Then is defined the DeltaCon kernel similarity between two graphs (Figure 2.10).

The second graph kernel is the kernel based on **regularized Laplacian**. It follows the same principle of the DeltaCon kernel, but uses a different similarity measure, the regularized Laplacian.

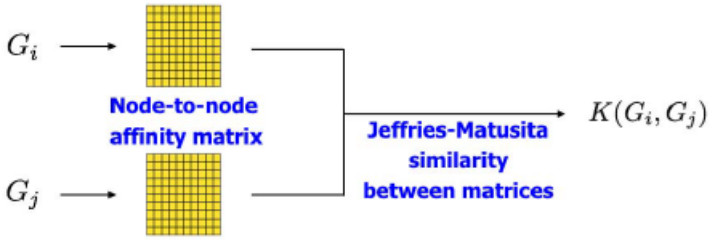

Figure 2.10: DeltaCon Kernel similarity.

The experiments aim to classify connectomes, detecting the gender of the brain in input. The method that gave better results is the Bag-of-edges, even if even the other two models performed better than other graph kernels take from the literature. In particular, the DeltaCon kernel worked better with regularized Laplacian similarity.

Sub-network Kernels for Measuring Similarity of Brain Connectivity Networks in Disease Diagnosis

The innovation of the literature of B. Jie [14] is in the fact that they take in consideration both global and local properties of brain regions to construct graph kernels for measuring the similarity of brain networks. They propose a novel sub-network kernel on brain networks for brain decease classification. They first construct a group of sub-networks on each node to reflect the multi-level connectivity properties of brain networks. Then, they define the similarity of a pair of brain networks, by calculating the similarities of all corresponding pairs of sub-network groups when considering the uniqueness of nodes. The total contribution of this work comprehend three steps: a novel sub-network kernel for measuring the similarity between brain networks, a sub-network kernel based learning (SKL) framework for automated brain disease diagnosis based on fMRI data and finally an implementation for performing inference on brain network data.

The proposed **sub-network kernel based learning** (**SKL**) framework for brain disease classification in composed by the following steps:

1. Image preprocessing and connectivity network construction;
2. Network thresholding and sub-network kernel construction;
3. Classification.

It is illustrated in Figure 2.11.

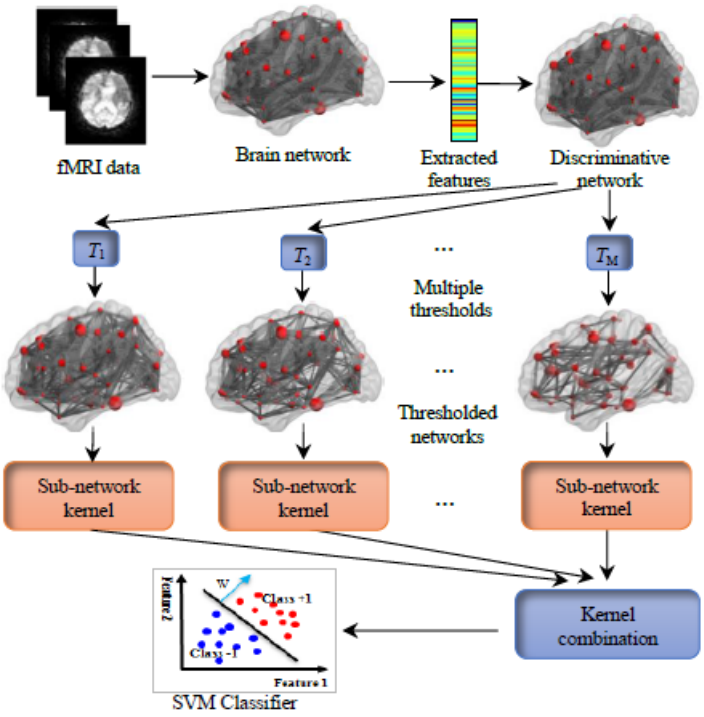

Figure 2.11: Sub-network kernel based learning framework (SKL) structure.

After the preprocess of the images, there is the construction of sub-network kernel. First a group of sub-networks is constructed on each node to reflect the multi-level connectivity properties of the brain. Then is calculated the similarity between brain networks. It is done calculating the similarity of all pairs of sub-networks groups from the same node across different brain networks, since each node on each brain network corresponds to the same ROI.

The sub-network kernel based Learning starts with a *discrimination* of the nodes to construct more discriminative networks. It is done with t-test and a thresholding with p-value. The result is a feature vector that represents the discriminative network. Then there is a *network thresholding*, with different thresholds, that will remove edges with zero weights, to reflect the topological properties of discriminative networks. Eventually, having more thresholds, they adopt a *multi-kernel SVM classification* with grid-search approach. Once found the optimal parameters, the traditional SVM can be applied for classification.

Compared with others state-of-the-art kernels, this method performs much better, with higher accuracy and AUC. They also experimented on

different parameters. The two parameters to tune are d, the number of iterations to compute the mathematical representation of sub-networks, and h, the size of a sub-network set. They saw that h is very important, because they reached the best performance at $h = 2$, while for d the method is robust. Another important evidence is that, without constructing discriminative networks from the originals brain network, the model is not so accurate compared to the model with it.

Integration Of Network Topological Features And Graph Fourier Transform For Fmri Data Analysis

Very interesting is the new approach of J. Wang et al [25]. Their challenge is to evaluate differences of functional connectivity networks between different age groups. The novelty is in the fact that the brain networks are constructed combining commonly used topological features from complex network analysis, with **Graph Fourier Transform** (GFT). GFT contributes to find the significant subspace of the original signal, so it could be a complementary information, given the fact that topological features reveal the morphological structure of the brain network.

In *Graph signal*, resting-state fMRI data can be viewed as time series graph signals defined on the parcellated brain regions. On each graph can be computed the GFT, and the eigenvalues are taken in consideration for the construction of the network. To represent the original graph signal, are selected the low frequency components, corresponding to the first several eigenvalues calculated by GFT.

For the topological features they calculate the *centrality* and the *segregation*. The centrality of the nodes is calculated with two measures, the degree and the betweeness centrality. The segregation refers to the existence of specialized neurons and brain areas, organized into distinct neural populations and grouped together to form segregated cortical areas. Functional segregation in the brain is the ability for specialized processing to occur within these areas. A simple measure of functional segregation is defined based on the number of triangles in the network, with a high number of triangles implying segregation.

To construct the graph, once calculated the topological features, they concatenate the data into a $264 \times (TM)$ matrix, where M is the number of subjects. Each row is normalized into a unit norm. To estimate the adjacency matrix is used the Gaussian radial basis functional kernel. Then they can get the features from the frequency domain by GFT.

After the construction of the input data, they use a regularized least-square regression using lasso algorithm for feature selection. The features are then given to a linear SVM classifier. The results of the experiment gives high accuracy, but they where not compared with other state-of-the-art methods. The steps of the model are illustrate in Figure 2.12.

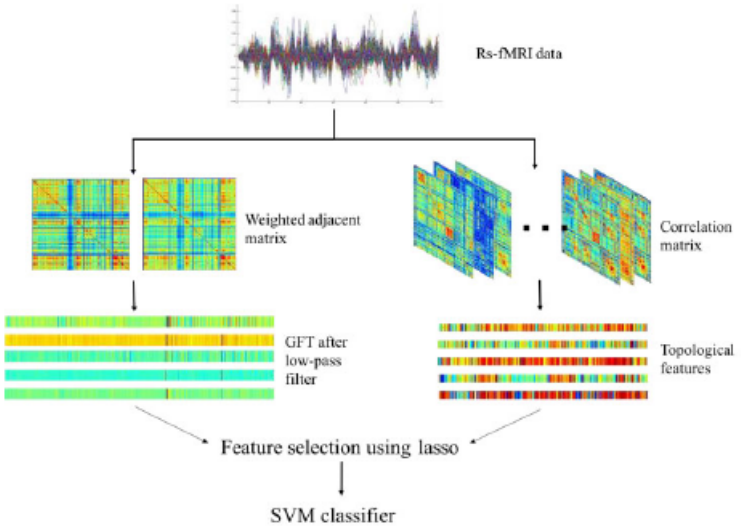

Figure 2.12: Steps of the experiment.

2.1.3 Machine Learning

Network Classification With Applications To Brain Connectomics

The goal of Arroyo-Relin at al [3] is to develop a high-dimensional network classifier that uses all the individual edge weights but also respects the network structure of the data and produces more interpretable results.

Regarding the network structures they aim to find nodes or subnetworks with good discriminative power, and hence select both the most informative nodes and edges. To capture structural assumptions on important predictive edges, they focus on convex structured sparsity penalties (Bach et al. [5]) that encourage a small number of active nodes, meaning nodes attached to at least one edge with a non-zero coefficient. For this purpose they use a group lasso penalty, that eliminate a group of variables simultaneously. To enforce spacial smoothness in the solution, they aim for a regularization that can be applied to any kind of network data.

To solve this penalty problem they use an optimization algorithm with two common approaches to convex optimization, proximal algorithms and alternative direction method of multipliers (ADMM). They use an accelerated version of the proximal algorithm (Beck and Teboulle [6]) to solve the main problem. In each step, they need to calculate a proximal operator, which is a further convex optimization problem solved with the ADMM algorithm.

They first experiment their method with synthetic graphs, comparing it with other state-of-the-art models, with some that takes in account the network structure and some that does not. With small number of communities the method outperforms the others in terms of classification. When it is increased, it could be comparable to methods that do not use network structures. The runtime is of an average of ten minutes for this method, while for most of the others is of ours. Experimenting on Schizophrenia datasets they outperformed all methods except the SVM model, that has not variable selection. Still it has a very high accuracy, and can also correctly identify brain regions that are suspected to be involved in the study of the Schizophrenia disease.

Stable Biomarker Identification For Predicting Schizophrenia in the Human Connectome

Stable Biomarker Identification model [13] is one of the methods we will take in account for the experimentation part. They adopt a machine learning approach that aims at discovering the most relevant set of biomarkers for discriminating subjects groups and thus quantitatively describing the group differences, both in terms of classification accuracy and stability of selected features.

Biomarkers discovery consists on the identification of regions or connections of interest associated with a neural disorder. From machine learning perspective, the choice of biomarkers can be addressed as a feature selection problem. They perform an automatic feature selection procedure in order to identify biomarkers that are relevant for the diagnosis of schizophrenia from brain connectivity data. As a classifier they use an RFE-SVM, integrated into an embedded feature selection approach. The aim of the present work is threefold:

- First, they investigate the effect of structural, functional, and multimodal (structural+functional) connectome with different resolutions in the classification performance of schizophrenia.
- Second, they perform a careful feature selection procedure across modal-

ities in order to assess the robustness of the selected features providing the best trade-off between high accuracy and stability.

- Finally, the analysis of retrieved biomarkers allows to identify a distributed set of brain regions engaged in the discrimination of patients and control subjects.

As we can see in Figure 2.13 the model has two Cross Validation. The outer CV, represented by the left image, is used to evaluate the performance of the model. The inner CV, at the right part of the image, is used to choose the best parameters for the classification.

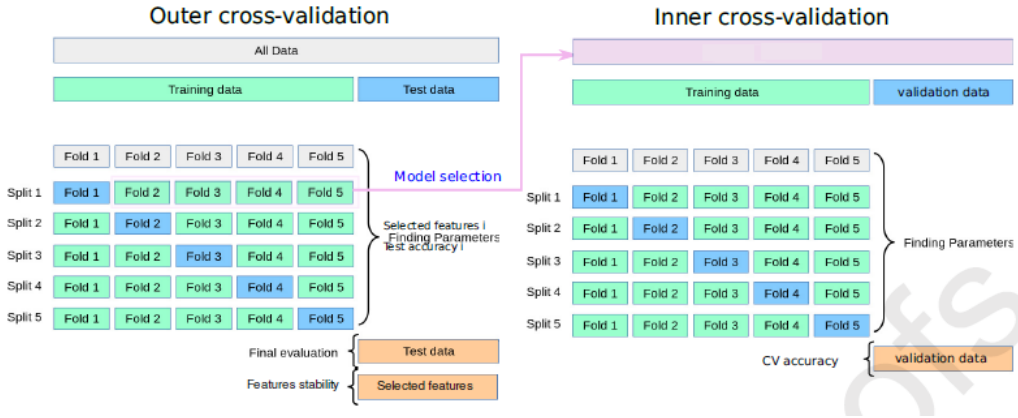

Figure 2.13: Structure of Stable Biomarkers Identification model

They observed that functional modality gives better performance than structural one, but is less stable. The better performance is given by the multi-modality input, and gives also a good trade-off between good performance and stable biomarkers. They even proceeded with the identification of brain areas involved in the classification of patients and controls.

Multi-modality disease modelling via collective deep matrix factorization

This model of Q. Wang et al [26] is based on a framework to fuse multiple data modalities for predictive modeling, using deep matrix factorization. In particular, they study three modalities together, two kinds of MRI and genetic data. The first type of MRI is **T1 MRI**, that capture structural information of grey matter in the brain. The other is the diffusion-weighted MRI (**dMRI**), that is sensitive to microscopic properties of brain's white

matter. So, T1 MRI captures areas composed of neurons while dMRI estimated connections between those area. The genotype impacts the disease in a way that is not directly related to brain structure and function. All these modalities interact in a complicated manner, this suggests that directly combining feature spaces may not lead to effective integration.

To reduce the feature dimensionality while maintaining most information they use **matrix factorization technique**. Traditional matrix factorizations assume linear interactions between data. This cannot be the case, there are non-linear interactions. They propose a deep matrix factorization framework to fuse information from multiple modalities and transfer predictive knowledge to differentiate patients with mild cognitive impairment (MCI), early stage of Alzheimer, from cognitive normal subjects. They build a non-linear hierarchical deep matrix factorization framework which decomposes each modality into a modality invariant component and a modality specific component, guided by supervision information.

To fuse multiple data modalities through deep matrix factorization they use a deep neural network to factorize each modality. The structure is illustrated in Figure 2.14.

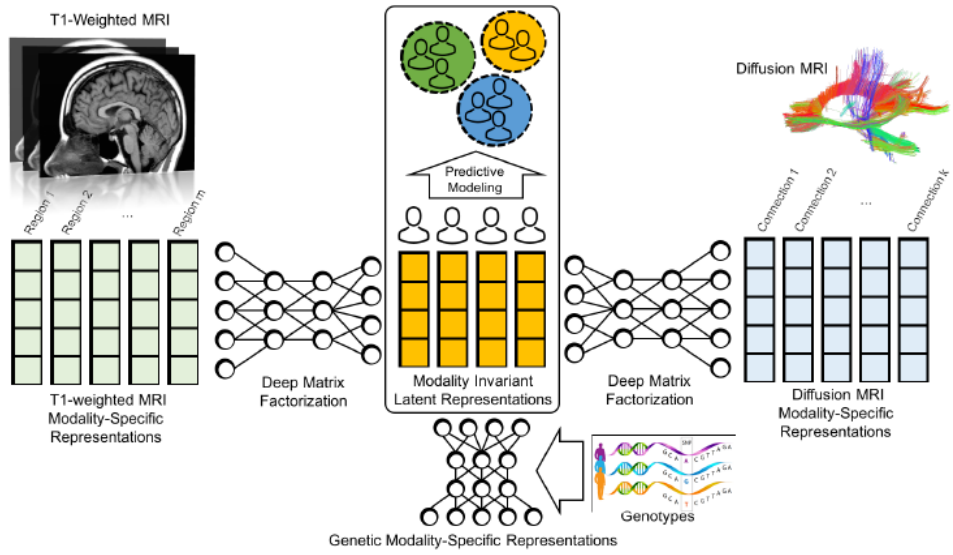

Figure 2.14: Structure of the deep neural network of Deep Matrix Factorization

The deep neural network serves as highly non linear mapping between the input matrix and the factorized matrix, and projects the latent represen-

tations non-linearly to the same latent space. They saw that with the fusion of all the three modalities the performance are higher than when they use only one or two modalities, and also outperforms other matrix factorization methods.

Chapter 3

Experiments

3.1 Experimental setup

3.1.1 Methods studied

So far, we have seen some methods for brain network classification. Now we are going to describe the experiments done with some of them, to evaluate their performance in a more general setting. This means that the experiments are done with other brain networks datasets. In particular, the methods we are going to experiment are four:

- Explainable Classification of Brain Networks via Contrast Subgraphs 2.1.2;
- Unsupervised Network Embedding for Graph Visualization, Clustering and Classification 2.1.2;
- Network Classification with Application to Brain Connectomics 2.1.3;
- Stable Biomarker Identification for Predicting Schizophrenia in the Human Connectome 2.1.3.

To make a recap and have in mind what we are studying, we will briefly summarize what they do. The first two methods are in the family of statistical fingerprints, the second two in Machine learning.

The first in the list aims to extract a constrast-subgraph, a set of vertices whose induced subgraph is dense in the summary graph of the condition group, and sparse in the summary graph of the control group. To make classification they calculated the number of edges of the subgraph induced by the contrast subgraphs (ASD-TD and TD-ASD) for each patient. They made classification with these two features and an SVM.

In Unsupervised network embedding, they train an autoecoder neural network that construct a feature space for each graph in input. These features, composed of embedding vectors, will be the input of a classification function.

Network Classification is given us like a library of the programming language R. Their aim is to find nodes or subnetworks with good discriminative power, meaning that they want to select only the most informative nodes and edges. To capture structural assumptions on these informative edges, they focus on convex structured sparsity penalties, with convex optimization.

In Stable Biomarkers identification they perform an automatic feature selection procedure to identify biomarkers that will be used to classify brain networks. In this case classify people affected by schizophrenia. To make classification they also design a RFE-SVM classifier.

3.1.2 Datasets used

The principle datasets are four big ones, from which are extracted other ones. The main four are:

- **ABIDE** (Autism Brain Imaging Data Exchange) [9], is a collaboration of 16 international imaging sites that have aggregated and are openly sharing neuroimaging data from 539 individuals suffering from ASD (autism) and 573 typical controls. These 1112 datasets are composed of structural and resting state functional MRI data along with an extensive array of phenotypic information. All data are preprocessed and different types of preprocessing are described in the web site.

- **MTA** (Multimodal Treatment of Attention Deficit Hyperactivity Disorder) [1], datasets from a subset of MTA participants at 6 sites, both with and without childhood ADHD, who were studied as part of a follow-up multimodal brain imaging examination. The principal aim of the effort was to investigate the effect of cannabis use among adults with a childhood diagnosis of ADHD. The study was a 2x2 design of those with/without ADHD and those who did/did not regularly use cannabis.

- **HCP** (Human Connectome Project) [27], includes high-resolution scans from young healthy adult twins and non-twin siblings (ages 22-35) using four imaging modalities: structural images, resting-state fMRI (rfMRI), task-fMRI (tfMRI), and high angular resolution diffusion imaging (dMRI). Behavioural and other individual subject measure data are included on all subjects.

- **Schizophrenia** [2], is a dataset acquired from approximately 100 patients with schizophrenia and 100 age-matched controls during rest as well as several task activation paradigms targeting a hierarchy of cognitive constructs. Neuroimaging data include structural MRI, functional MRI, diffusion MRI, MR spectroscopic imaging, and magnetoencephalography.

The datasets ABIDE, MTA and Schizophrenia are also used with all the patients because are already divided in control group and condition group, while HCP dataset must be divided.

The datasets have been extracted from ABIDE are four, *Children*, that includes patients which are at most 9 years, *Adolescents*, individuals between 15 and 20 years old, *EyesClosed*, with people that performed their fMRI with their eyes closed, and *Male*, considering only male subjects. These had been already extracted by T. Lanciano et al [17], and kindly given to me to make experiments.

From HCP dataset we though that could be interesting to divide male and female patients, dataset that we will call hcp-gender, to see if they are in some way different in their brain connections, like in some other papers described in 2.

At the end we have eight datasets: ABIDE, Children, Adolescents, EyesClosed, Male, MTA, Schizophrenia and hcp-gender.

It is important to specify that each patient is represented by an adjacency matrix contained in a .csv file.

3.1.3 Code

All the experiments were launched with the program Anaconda. We start with an environment of Python 3.5, but some of the methods were written in Python 2.7, so we switched to an Anaconda environment with Python 2.7 when requested from the method.

To make the experiments easier to perform, was implemented a python script at which can be specified, as input arguments, which method we want to evaluate, which dataset to use and even if the data must be weighted or binary, meaning that one can choose to have weighted edges of the network, or binary ones.

After a method has been runned there is the classification part. The classification is done for only two of the methods we take in consideration.

These are *Explainable Classification of Brain Networks via Contrast Subgraphs* [17] and *Unsupervised Network Embedding for Graph Visualization, Clustering and Classification* [12], because, as said in their summary explanation, what we extract form them are some features of the graphs in input, and for this reason we have to design a classification for them. The other two methods have their own classification in the code, and we though it was not reasonable to use a different one.

Our classifier is designed with the grid search function of the Python library sklearn, to find the best parameters for each method. We used a cross-validation of 5/10 folds, repeated 5 times. To split the dataset we choose to have 80% of train set and 20% of test set. Then from the best parameters found, we make predictions and evaluate the performance with **accuracy**. The accuracy is calculated directly from the grid search function for each loop of the cross-validation, so we take the mean of all the values.

All datasets have their own preprocess, made from who released the data. We also added some data modification before to give the datasets to the algorithms. First of all is checked if the values of the diagonal of each matrix were zeros, if not they are corrected, because we do not want to consider each ROI connected to itself, so like an edge. Then, for each adjacency matrix we calculated the 0.2 quantile and 0.8 quantile, to leave out all the values of the edges lower than the 0.2 quantile and bigger than the 0.8 quantile. This is done for all datasets, except for *Adolescents*, *EyesClosed* and *Male*, because they were in binary form, so all values in the matrices were zeros and ones, and are left in the original form, after checking the diagonal.

Now, as we said, for the other datasets we could choose if transform them in binary edged or leave them with weighted edges. If it is chosen to transform them in binary data, after leaving out the values below and above the quantiles, all the values different from zero are transformed in ones, even the negative values.

3.2 Results

Now we are going to show the results of each method.

	b	w
children	0.51 ± 0.04	0.49 ± 0.08
adolescents	0.53 ± 0.03	0.53 ± 0.03
eyesclosed	0.54 ± 0.09	0.54 ± 0.09
male	0.51 ± 0.07	0.51 ± 0.07
ABIDE	0.52 ± 0.03	0.52 ± 0.06
mta	**0.69 ± 0.03**	**0.69 ± 0.02**
hcp_gender	0.53 ± 0.11	0.53 ± 0.02
schizophrenia	0.54 ± 0.05	0.51 ± 0.06
mta_cannabis	0.47 ± 0.04	0.55 ± 0.10

Table 3.1: Results of Contrast-subgraph method with each dataset.

3.2.1 Results of Explainable Classification of Brain Networks via Contrast Subgraphs

With the code taken from the work of Lanciano et al [17] we extract the two contrast-subgraphs ASD-TD and TD-ASD. Then we calculate the subgraphs induced by the two contrast-subgraphs on each patient. For each subgraph obtained we count the number of edges. In this way we have two features (columns), one is the number of edges induced by ASD-TD for each patient, and the other is the number of edges induced by TD-ASD for each patient. Each patient is represented by a row. We give these features to the classifier described in the previous section, designed by us. This procedure is done for each dataset, and for each dataset is done for weighted data and binary data. Also, we tried with different values of α (0.01, 0.02, 0.03).

Each experiment is repeated ten times, in order to take the mean of all the accuracies, and the standard deviation. The results obtained with each dataset is shown in Table 3.1. We can see that the best result with this method is given with the dataset *mta* both in binary and weighted form. The running time to extract the contrast-subgraphs is very low, only few seconds, and the running time to make classification is at most 16 minutes.

It is interesting to see how change the number of edges induced by both the contrast-subgraphs, depending on the class of the patient. In the scatterplot of Figure 3.1 are reported these features for the best result obtained with this method, so with dataset mta and $\alpha = 0.02$. We can notice that the majority of tipically developed patients have more edges induced by contrast subgraph ASD-TD than patients affected by Autism,. Also, we can see the diagonal line that should represent the division between the two classes of patients. Our is not very precise, and we can see that even from the accuracy

result, that is 69%.

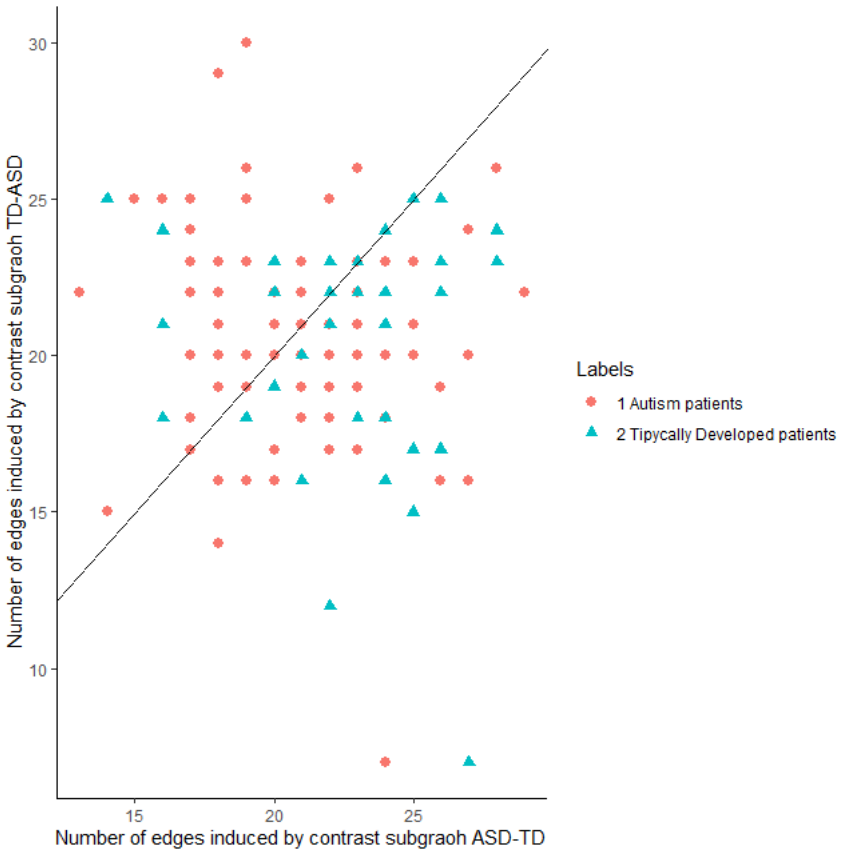

Figure 3.1: Scatterplot of number of edges induced by the contrast subgraphs to mta dataset. At x-axis there the number of edges induced by the contrast subgraph ASD-TD, while at y-axis the once induced by the contrast subgraph TD-ASD, on each patient.

3.2.2 Results of Unsupervised Network Embedding for Graph Visualization, Clustering and Classification

The output we have from *Unsupervised Network Embedding* is an embedded vector for each patient. The embedded vector contains 1024 values, so we will have a matrix in which each row corresponds to a patient, and the columns are the values of the embedded vector of each patient. Obviously, each dataset is studied separately and each one has different extracted features. We give the features to our classifier, and repeat the experiment ten times for each dataset, to take at the end the mean of all accuracies and the standard deviation.

	b	w
children	0.53 ± 0.05	0.50 ± 0.05
adolescents	0.55 ± 0.07	0.55 ± 0.07
eyesclosed	0.52 ± 0.04	0.52 ± 0.04
male	0.56 ± 0.02	0.55 ± 0.03
ABIDE	0.52 ± 0.01	0.52 ± 0.05
mta	0.62 ± 0.03	**0.69 ± 0.09**
hcp_gender	0.62 ± 0.08	0.56 ± 0.02
schizophrenia	0.63 ± 0.03	0.60 ± 0.01
mta_cannabis	0.56 ± 0.03	0.51 ± 0.09

Table 3.2: Results of Unsupervised network embedding method with each dataset.

The results for each dataset are shown in Table 3.2. From that we can see that this method worked better with the dataset mta in weighted form, so leaving the edges weighted. This algorithm has five parameters that can be changed: *number of epochs*, *batch size*, *embedding size*, *learning rate* and the *noise* to apply to data. We made many tests with different parameters to find the best combination for each dataset.

The best results is obtained again with dataset *mta* in weighted form. In Figure (...) we can see a plot implemented by L. Gutiérrez and adjusted for our output, that shows how our data, so the brain networks, are divided in their correspondence class.

The running time to extract the features depends most of all on the number of epochs and length of the data, anyway it is not very high, more and less 15 minutes. Even the classification part does not take long, the maximum time, with both extracting the features and classification, is 2 hours, but the mean of the total runtime between all the datasets is 30 minutes.

3.2.3 Results of Network Classification with Application to Brain Connectomics

In this work is implemented also the classification. It is written in R code, and, to compute the accuracy score, we were able to extract the predicted values and the true values. The main function is *graphclass()*, and has three parameters to try and tune, that are lamba λ, rho ρ and gamma γ. As in the previous experiments, we run the algorithm 10 times for each dataset, then compute the mean of all accuracies and the standard deviation. The results for all the datasets are in Table 3.3. The best score is with the

	b	w
children	0.59 ± 0.09	0.54 ± 0.07
adolescents	0.61 ± 0.05	0.61 ± 0.05
eyesclosed	0.55 ± 0.05	0.55 ± 0.05
male	0.61 ± 0.02	0.61 ± 0.02
ABIDE	0.61 ± 0.04	0.58 ± 0.03
mta	0.71 ± 0.03	0.68 ± 0.02
hcp_gender	**0.77 ± 0.01**	0.71 ± 0.03
schizophrenia	0.58 ± 0.09	0.59 ± 0.06
mta_cannabis	0.57 ± 0.07	0.59 ± 0.12

Table 3.3: Results of Network classification method with each dataset.

dataset hcp_gender, for which the best parameters are $\lambda = 10^{-4}$, $\rho = 1$ and $\gamma = 10^{-5}$.

The mean of the running time is 5 minutes, anyway it is not more than 11 minutes, depending on the length of the data.

3.2.4 Results of Stable Biomarker Identification for Predicting Schizophrenia in the Human Connectome

As the previous model, even *Stable Biomarkers Identification* has already in the code the classification part. The results that we have in output is a .mat file, where are stored all the accuracies computed at each step of the outer cross-validation (see 2.1.3). Each accuracy corresponds to the mean of all accuracies evaluated within the inner cross-validation, in which each step corresponds a certain percentage of features. Each accuracy has also the corrispondent standard deviation. We show the best accuracy computed, even if this means that each dataset could have the best result with a different number of features. Regarding the data, they want to study functional, structural and multimodal data. We only have functional networks, so we will make evaluation only on this kind of data modality. Also, at the end, they evaluate the robustness of the selected features, the trade-off between accuracy and stability, and identify the set of brain regions engaged in the discrimination of patients. In our work, we will only stop at classification evaluation. From Table 3.4 we can see al the results obtained with each dataset. As we can see, we have the best result with this method with the dataset *schizophrenia*, in binary form.

The running time of this implementation is very high, it depends on the dimention of the dataset, but it takes from six to eight ours to end.

	b	w
children	0.59 ± 0.09	0.54 ± 0.07
adolescents	0.61 ± 0.05	0.61 ± 0.05
eyesclosed	0.55 ± 0.05	0.55 ± 0.05
male	0.61 ± 0.02	0.61 ± 0.02
ABIDE	0.61 ± 0.04	0.58 ± 0.03
mta	0.71 ± 0.03	0.68 ± 0.02
hcp_gender	0.77 ± 0.01	0.71 ± 0.03
schizophrenia	0.58 ± 0.09	0.59 ± 0.06
mta_cannabis	0.57 ± 0.07	0.59 ± 0.12

Table 3.4: Results of Stable biomarkers identification method with each dataset.

We recall that they designed an RFE-SVM, meaning that they compute the accuracy for different percentage of features taken. For this reason it is interesting to see how the accuracy changes according to the features percentage. In their pubblication they already showed some plots to evaluate this attribute. We took the same plot, the one already written from them in a python script, and adapted it to our results. This is shown in Figure 3.2. The graph shows us that in this case the accuracy does not change much according to features percentage, still more features we consider, more the accuracy is high, even if sligthly.

3.3 Discussion

We can see the summary of all the results mentioned in this chapter in Table 3.5. According to our experiments, the methods that have best accuracy are *Stable Biomarkers identification for predicting schizophrenia in Human Connectome* and textitNetwork classification with Application to Brain Connectomics. This means that they are valid tools to make classification on brain networks.

The accuracy is not the only evaluation that is worth to take in consideration. Obviously it is really important to have right results, in order to use these methods for scientific purpuses. What we think also important to evaluate is the running time. All the algortihms that we evaluated have different running times, we can see them in Figure (...). From this plot we can see how much time the work of Gutiérrez et al. [13] takes to evaluate all the dataset, and this is not very good if we want to make a diagnosis for a patient.

Taking in mind the considerations made now, we can say that between

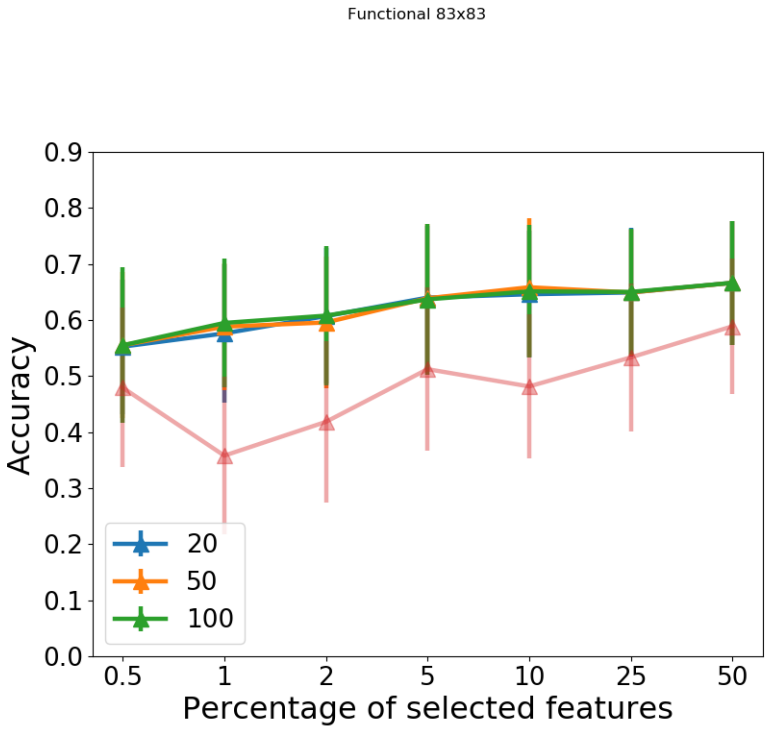

Figure 3.2: Graph representing the accuracy according to the percentage of features selected to make classification.

		Contrast-subgraph	GraphEmbs	Biomarkers	Graphclass
children	b	0.50 ± 0.02	0.50 ± 0.02		
	w				
adolescents	b				
	w				
eyesclosed	b				
	w				
male	b				
	w				
ABIDE	b				
	w				
mta	b				
	w				
hcp_gender	b				
	w				
schizophrenia	b				
	w				

Table 3.5: Accuracy results and standard deviation of each method experimented with each dataset.

the methods we choose to make brain network classification, the best one is *Network classification with Application to Brain Connectomics*, in fact it gives the best results of our experiments and do not take much time to evaluate out datasets.

From these results we can say that even the dataset is important, because it needs to be as good as possible for that method. Here we have several datasets. Most of them have been collected with different machines and technologies. Also, they have been trasformed and preprocessed in different ways. This is the reason why it is important to evaluate each method with the highest amount of datasets, because we want to see how the results change. We can see if some method is more sutable for some kind of mental disease, even if it written that could be used for all mental disorder.

Chapter 4

Conclusions

In this work we made an empirical summary of methods on *Brain Network Classification*. We have seen how it is important to choose the right one for each dataset, together with the right parameters. In particular we have:

- Described some basics arguments to go easier through the work;
- Made a summary of all methods we have found in the literature that make brain network classification, even if not all perfectly alligned with our experiments;
- Selected the methods to make experiments, so the ones that take as input wighted graphs in Adjacency matrices.
- Made experiments on those methods and comment on the results.

4.1 Future works

In future investigations we would like to:

- Get better results, in terms of accuracy, from the methods we made experiments, investigating further their parameters;
- Find other methods in the literature that implemented brain network classification;
- Make experiments even on different methods, that take as input not only matrices;
- Some of the methods we have taken in account, also take in account the nodes of the graphs as real parts of our brain. For each method we could explore how good they do that, and consequently which are the relevant areas that influence the classification of particular mental desease.

Bibliography

[1] ACPI. Acpi: Multimodal treatment of attention deficit hyperactivity disorder (mta) - preprocessed arra nida. http://fcon_1000.projects.nitrc.org/indi/ACPI/html/acpi_mta_1.html, 2015.

[2] C. Aine, H. J. Bockholt, J. Bustillo, J. Ca(ñ)ive, A. Caprihan, C. Gasparovic, F. Hanlon, J. Houck, R. Jung, J. Lauriello, J. Liu, A. Mayer, N. Perrone-Bizzozero, S. Posse, J. Stephen, J. Turner, V. Clark, and V. Calhoun. Multimodal neuroimaging in schizophrenia: Description and dissemination. *Neuroinformatics*, 15, 10 2017.

[3] J. D. Arroyo Relin, D. Kessler, E. Levina, and S. F. Taylor. Network classification with applications to brain connectomics. *The Annals of Applied Statistics*, 13(3), Sep 2019.

[4] S. Asiri. Machine learning classifiers. https://towardsdatascience.com/machine-learning-classifiers-a5cc4e1b0623, 2019.

[5] F. Bach, R. Jenatton, J. Mairal, and G. Obozinski. Structured sparsity through convex optimization, 2012.

[6] A. Beck and M. Teboulle. A fast iterative shrinkage-thresholding algorithm for linear inverse problems. *SIAM Journal on Imaging Sciences*, 2(1):183–202, 2009.

[7] X. Bi, X. Zhao, H. Huang, D. Chen, and Y. Ma. Functional brain network classification for alzheimers disease detection with deep features and extreme learning machine. *Cognitive Computation*, 12:513–527, 2019.

[8] T. brain preservation foundation. What is a connectome? https://www.brainpreservation.org/content-2/connectome, 2017.

[9] C. Cameron, B. Yassine, C. Carlton, C. Francois, E. Alan, J. András, K. Budhachandra, L. John, L. Qingyang, M. Michael, Y. Chaogan, and B. Pierre. The neuro bureau preprocessing initiative: open sharing of preprocessed neuroimaging data and derivatives. *Frontiers in Neuroinformatics*, 7, 2013.

[10] B. Chim, F. Crevecoeur, and J.-C. Delvenne. Supervised classification of structural brain networks reveals gender differences. In *2018 19th IEEE Mediterranean Electrotechnical Conference (MELECON)*, pages 269–274, 2018.

[11] D. K. Duvenaud, D. Maclaurin, J. Iparraguirre, R. Bombarell, T. Hirzel, A. Aspuru-Guzik, and R. P. Adams. Convolutional networks on graphs for learning molecular fingerprints. In C. Cortes, N. Lawrence, D. Lee, M. Sugiyama, and R. Garnett, editors, *Advances in Neural Information Processing Systems*, volume 28. Curran Associates, Inc., 2015.

[12] L. Gutiérrez-Gómez and J.-C. Delvenne. Unsupervised network embeddings with node identity awareness. *Applied Network Science*, 4(1), Oct. 2019.

[13] L. Gutirrez-Gmez, J. Vohryzek, B. Chim, P. S. Baumann, P. Conus, K. D. Cuenod, P. Hagmann, and J.-C. Delvenne. Stable biomarker identification for predicting schizophrenia in the human connectome. *NeuroImage: Clinical*, 27:102316, 2020.

[14] B. Jie, M. Liu, D. Zhang, and D. Shen. Sub-network kernels for measuring similarity of brain connectivity networks in disease diagnosis. *IEEE Transactions on Image Processing*, 27(5):2340–2353, May 2018.

[15] D. Koop, J. Freire, and C. T. Silva. Visual summaries for graph collections. In *2013 IEEE Pacific Visualization Symposium (PacificVis)*, pages 57–64, 2013.

[16] D. Koutra, J. Vogelstein, and C. Faloutsos. Deltacon: A principled massive-graph similarity function. 2013.

[17] T. Lanciano, F. Bonchi, and A. Gionis. Explainable classification of brain networks via contrast subgraphs. In *Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*. ACM, July 2020.

[18] Y. Lecun, L. Bottou, Y. Bengio, and P. Haffner. Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 86(11):2278–2324, 1998.

[19] J. B. Lee, R. Rossi, and X. Kong. Graph classification using structural attention. In *Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, KDD '18, page 16661674, New York, NY, USA, 2018. Association for Computing Machinery.

[20] D. Rogers and M. Hahn. Extended-connectivity fingerprints. *Journal of Chemical Information and Modeling*, 50(5):742–754, 2010. PMID: 20426451.

[21] S. Sarraf and G. Tofighi. Deep learning-based pipeline to recognize alzheimer's disease using fmri data. *bioRxiv*, 2016.

[22] N. Shervashidze, S. Vishwanathan, T. Petri, K. Mehlhorn, and K. Borgwardt. Efficient graphlet kernels for large graph comparison. In D. van Dyk and M. Welling, editors, *Proceedings of the Twelth International Conference on Artificial Intelligence and Statistics*, volume 5 of *Proceedings of Machine Learning Research*, pages 488–495, Hilton Clearwater Beach Resort, Clearwater Beach, Florida USA, 16–18 Apr 2009. PMLR.

[23] A. W. Toga, K. A. Clark, P. M. Thompson, D. W. Shattuck, and J. D. Van Horn. Mapping the Human Connectome. *Neurosurgery*, 71(1):1–5, 07 2012.

[24] K. Ugurbil and D. V. Essen. Hcp young adult. https://www.humanconnectome.org/study/hcp-young-adult, 2017.

[25] J. Wang, V. D. Calhoun, J. M. Stephen, T. W. Wilson, and Y.-p. Wang. Integration of network topological features and graph fourier transform for fmri data analysis. In *2018 IEEE 15th International Symposium on Biomedical Imaging (ISBI 2018)*, pages 92–96, 2018.

[26] Q. Wang, M. Sun, L. Zhan, P. Thompson, S. Ji, and J. Zhou. Multi-modality disease modeling via collective deep matrix factorization. In *Proceedings of the 23rd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '17, page 11551164, New York, NY, USA, 2017. Association for Computing Machinery.

[27] M. Woolrich, B. Ripley, M. Brady, and S. M. Smith. Temporal autocorrelation in univariate linear modeling of fmri data. *NeuroImage*, 14:1370–1386, 2001.

[28] Y. Yan, J. Zhu, M. Duda, E. Solarz, C. Sripada, and D. Koutra. Groupinn: Grouping-based interpretable neural network-based classification of limited, noisy brain data. In *Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining, KDD 2019, London, UK, August 4-8, 2019*, 2019.

[29] J. Zhou, G. Cui, S. Hu, Z. Zhang, C. Yang, Z. Liu, L. Wang, C. Li, and M. Sun. Graph neural networks: A review of methods and applications. *AI Open*, 1:57–81, 2020.