PUCRS

ESCOLA POLITÉCNICA
PROGRAMA DE PÓS-GRADUAÇÃO EM CIÊNCIA DA COMPUTAÇÃO
MESTRADO EM CIÊNCIA DA COMPUTAÇÃO

MATHEUS ZAMPIERI MARCON

GEOMETRIC DEEP LEARNING FOR FUNCTIONAL NEUROIMAGING ANALYSIS

Porto Alegre
2021

PÓS-GRADUAÇÃO - *STRICTO SENSU*

Pontifícia Universidade Católica
do Rio Grande do Sul

PONTIFICAL CATHOLIC UNIVERSITY OF RIO GRANDE DO SUL
SCHOOL OF TECHNOLOGY
COMPUTER SCIENCE GRADUATE PROGRAM

GEOMETRIC DEEP LEARNING FOR FUNCTIONAL NEUROIMAGING ANALYSIS

MATHEUS Z. MARCON

Thesis submitted to the Pontifical Catholic University of Rio Grande do Sul in partial fulfillment of the requirements for the degree of Master in Computer Science.

Advisor: Prof. Felipe Rech Meneguzzi

Porto Alegre
2021

Ficha Catalográfica

M321g Marcon, Matheus Zampieri

Geometric deep learning for functional neuroimaging analysis / Matheus Zampieri Marcon. – 2021.
90 p.
Dissertação (Mestrado) – Programa de Pós-Graduação em Ciência da Computação, PUCRS.

Orientador: Prof. Dr. Felipe Rech Meneguzzi.

1. Inteligência Artificial. 2. Neuroimagem. 3. fMRI. 4. Redes Neurais Profundas. I. Meneguzzi, Felipe Rech. II. Título.

Elaborada pelo Sistema de Geração Automática de Ficha Catalográfica da PUCRS
com os dados fornecidos pelo(a) autor(a).
Bibliotecária responsável: Clarissa Jesinska Selbach CRB-10/2051

Dedico este trabalho a meus pais e minha companheira Fabiana.

ACKNOWLEDGMENTS

I thank my advisor Felipe Rech Meneguzzi, professor Augusto Buchweitz and my colleagues Marcelo Duarte and Nathalia Esper for the support on the development of this work. This work was supported by the Conselho Nacional de Desenvolvimento Científico e Tecnológico (CNPq).

APRENDIZADO PROFUNDO GEOMÉTRICO PARA ANÁLISE DE NEUROIMAGENS FUNCIONAIS

RESUMO

O estudo do conectoma cerebral humano, um conjunto complexo de relações entre redes neurais cerebrais que associam estrutura cerebral e funcionalidade, têm recebido crescente interesse na área de neuroimagem ao longo da última década. Técnicas de aprendizado profundo constituem o estado da arte para tarefas de classificação de diferentes disordens neurológicas a partir de neuroimagens, proporcionando análises em profundidade acerca de características inerentes da atividade e conectividade cerebrais sem a necessidade prévia de seleção de *features*. No entanto, operações convolucionais de redes profundas tradicionais são aplicadas a regiões fixas de elementos durante o aprendizado, enquanto dados de conectoma cerebral são melhor representados na forma de grafos, com elementos espacialmente dispersos. Neste trabalho, fazemos uso de técnicas de aprendizado profundo geométrico para análise de dados de conectoma de imagens de ressonância magnética funcional (fMRI), buscando a identificação e extração de representações de características de alto nível das dinâmicas de redes cerebrais envolvidas na cognição humana. Nossas conclusões sugerem que as técnicas investigadas podem superar o estado da arte relativo a modelos de classificação de dados de fMRI além de possibilitar uma metodologia simples para análise de resultados.

Palavras-Chave: inteligência artificial, neuroimagem, fMRI, redes neurais profundas.

GEOMETRIC DEEP LEARNING FOR FUNCTIONAL NEUROIMAGING ANALYSIS

ABSTRACT

The study of the human brain connectome, a complex set of cerebral network relationships associating structure and functionality, has seen a growing interest in the field of neuroimaging over the last decade. Deep learning techniques constitute the state-of-the-art for neuroimaging classification tasks on different neurological disorders, providing in-depth analysis into the inherent characteristics of brain activation and connectivity without the need for prior feature selection. However, convolutional operations of traditional deep networks affect fixed regions of elements during learning, whereas connectome data is best represented in the form of graphs, with spatially dispersed elements. We make use of geometric deep learning (GDL) for the analysis of whole-brain functional magnetic resonance imaging (fMRI) connectome data to identify and extract high-level feature representations of the cerebral network dynamics involved in human cognition. Our findings suggest that GDL techniques can outperform state-of-the-art models for classification of fMRI data while providing a simple framework for result analysis.

Keywords: artificial intelligence, neuroimaging, fMRI, deep neural networks.

LIST OF FIGURES

LIST OF TABLES

CONTENTS

1. INTRODUCTION

In its founding years, and well over the course of its development, the field of neuroscience maintained the viewpoint that brain structure and functionality could be reduced to the workings of the neurons, cells acting as individual units. Modern neuroscience, however, recognizes that the analysis of single neurons is insufficient for providing a general theory of the brain capable of explaining behavior, cognition and mental pathologies [Yuste, 2015]. With the advancement over the last decades of functional neuroimaging techniques, consensus has shifted towards the assumption that connectivity between neuron clusters, and not units, are the source of brain function. Functional neuroimaging comprises a set of techniques that aim at investigating human brain functionality *in vivo*. Functional magnetic resonance imaging (fMRI) is a noninvasive method that has established itself as the most used neuroimaging modality for this investigative purpose, particularly concerning research applications. Neural activity is indirectly quantified with fMRI techniques through the emission and sensing of magnetic fields [Huettel et al., 2004, ch. 1], in an attempt to identify the mental processes associated with different brain regions. Distinct experiment protocols are employed in fMRI scan sessions for evaluating a broad range of phenomena of interest, from the default brain behavior during resting periods to brain activity resulting from the presentation of stimuli to the subjects.

Recent MRI research, particularly with the advent of computational neuroscience, indicates that individual brain regions may possess more than a single function, and that brain functionality emerges from the interaction patterns of distributed cerebral neural networks [Bullmore and Sporns, 2009][1]. The comprehensive mapping of these networks is known as the *human connectome*. The analysis of the connectome and its constitution is of major interest for both academic and clinical purposes, given recent findings relating connectome structure to mental disorders [Sporns et al., 2005, Essen et al., 2013]. Obtaining a more thorough understanding of the connectome, however, is still an ongoing endeavor. The development of methods to aid in the analysis of multimodal fMRI experiments could be of great value to for this field of research.

In this work, we employ geometric deep learning to generate a data-driven artificial neural network model to investigate the brain dynamics involved in the dyslexia disorder and language related cognitive processes. We use the ACERTA dataset, provided by the Brain Institute of Rio Grande do Sul, which contains functional fMRI scans of over 80 school-aged subjects diagnosed with dyslexia, in addition to as many healthy controls. We model fMRI scans data into graph structures that are used as input to Graph Convolutional Network (GCN) models [Defferrard et al., 2016]. We identify and analyze the cerebral network biomarkers relevant for performing different classification tasks, generating visualizations for

[1] This work concerns both biological and artificial neural networks. As such, unless obvious from context, we will use the terms *cerebral* or *brain* when referring to biological networks in order to avoid misinterpretation.

each task. To benchmark the performance of our models, we perform simple binary classification tasks in the Human Connectome Project (HCP) dataset, consisting of high-resolution fMRI scans for 1200 healthy young adults.

Our experiments show that GCNs can outperform baseline methods and produce results on par with the state-of-the-art methods for fMRI classification tasks while operating directly in connectome data and improving explainability. As such, our work demonstrates promising new prospects for the application of deep learning in neuroscience. To the best of our knowledge, this is the first work to apply spatial-temporal GCN models to task fMRI data, and the second to apply GCNs in general for this purpose.

1.1 Contributions

The main contribution of our work is to assess the ability of GCNs to identify biomarkers related to cognitive and neurodevelopmental traits in multimodal fMRI scans. We investigate the performance of GCN models in discriminating between dyslexic and healthy control subjects and between good and bad readers using multimodal connectome data, while focusing on the extraction and analysis of the most relevant features used for classification. Thus, we propose the following research question:

- Can geometric deep learning models improve classification performance and explainability in the analysis of cognitive and neurodevelopmental traits in fMRI data?

In answering this research question, we generate data-driven GCN models that surpass state-of-the-art techniques in data classification and provide a straightforward framework for biomarker analysis in connectome data. Furthermore, we show that such techniques are equally useful in the analysis of both large and small datasets. Our results show that GCNs constitute a powerful tool in the investigation of the high-level connectivity patterns of the human brain.

This dissertation provides five contributions:

(1) The application and comparison of GCN and baseline models in classification tasks using resting-state and task fMRI connectome data.

(2) The first application of spatial-temporal GCNs to task fMRI data.

(3) Analysis and validation of the biomarkers identified by spatial-temporal GCN models regarding their relation to dyslexia and neural development.

(4) A demonstration of the applicability of geometric deep learning in the study of multimodal brain connectomics in both small and large datasets.

1.2 Publications

During the master's program, we worked on the following publications:

- Marcon, Matheus; Meneguzzi, Felipe. (2021) "Applications of graph convolutional networks for neuroimaging: A systematic review." Unpublished Manuscript.
- Marcon, Matheus; Duarte, Marcelo; Esper, Nathalia B.; Buchweitz, Augusto; Meneguzzi, Felipe. (2021) "Identifying task-fMRI biomarkers in dyslexia via graph convolutional networks." Unpublished Manuscript.
- Ballester, Pedro L.; Da Silva, Laura T.; Marcon, Matheus; Esper, Nathalia B.; Frey, Benicio N.; Buchweitz, Augusto; Meneguzzi, Felipe. (2020) "Predicting brain age at slice level: convolutional neural networks and consequences for interpretability". Frontiers in Psychiatry. Accepted Manuscript.
- Amado, Leonardo R.; Licks, Gabriel P.; Marcon, Matheus; Pereira, Ramon F.; and Meneguzzi, Felipe. (2020) "Using Self-Attention LSTMs to Enhance Observations in Goal Recognition." In Proceedings of the 33rd International Joint Conference on Neural Networks (IJCNN).

2. NEUROIMAGING BACKGROUND

In this chapter we provide a brief background on the field of neuroimaging and the technique of Magnetic Resonance Imaging (MRI). Section 2.1 provides an introduction to Magnetic Resonance Imaging (MRI). Section 2.2 describes Structural MRI scans and the resulting data, while Section 2.3 provides further detail into functional MRI (fMRI), the main type of data we aim to use for our research. Section 2.4 discusses the different noise sources of MRI scans, and the preprocessing techniques applied to MR images. Section 2.5 briefly introduces the field of graph theory, and its applications in the study of human brain connectivity.

2.1 Magnetic Resonance Imaging

Magnetic Resonance Imaging (MRI) is a technique that uses strong magnetic fields, sequences of magnetic field gradients and radiofrequency signals for generating images of biological tissues [Lauterbur, 1973]. MRI consists of a non-invasive technique capable of generating images with different *contrasts*, both in spatial and temporal settings, thus being appropriate for a wide variety of experiments. Over the last decades, MRI usage has increased in prevalence both in clinical and research settings, being employed for investigating cardiac [Nandalur et al., 2007, Hoffmann et al., 2003, Shimada et al., 2001], joint [Küseler et al., 1998], spinal cord [Bondurant et al., 1990], bone [Zimmer et al., 1985], musculoskeletal soft tissues [Siegel, 2001], and neurological [Gong and He, 2014] conditions, the latter being the focus of this work.

MRI scanners work by emitting a series of varying electromagnetic fields and magnetic gradients, known as a *pulse sequence*, which are tuned to the frequency of the hydrogen atom [Huettel et al., 2004, Ch. 4]. The pulse sequences causes the hydrogen protons to be spatially aligned to the external magnetic field. This alignment is then perturbed by bursts of radiofrequency (RF) signals, which bring them to an excited state. As the nuclei return to a relaxed state, they emit different MR signals that reflect intrinsic properties of the tissues they compose. These signals are captured by an antenna that measures net magnetization changes in the tissue realized by these processes. The successive application of perturbations and measurements on carefully selected planes along the patient's body, known as *slices*, makes possible the construction of a virtual 3D volume of the subject's tissues. Each slice is composed of a 2D matrix of MR signals stored in voxels, which are the basic building blocks of the acquired image. Although represented in 2D, voxels are 3D, reflecting the thickness of each acquired slice.

Depending on the employed pulse sequence, different *contrasts* can be achieved. Contrasts refer to image acquisitions that differentiate between varying proton densities, gray and white matter, or fluid versus tissue [Huettel et al., 2004, Ch. 1]. Different contrasts depend on a set of *intrinsic* and *extrinsic* parameters. Extrinsic parameters are set by the technologist, and consist of slice thickness, resolution, echo time (TE) and repetition time (TR). TR is the time between successive excitation pulses in the same slice, and TE the time between emission of the RF pulse and detection of the echo signal. Intrinsic parameters depend on individual tissue characteristics, and consist of spin-lattice relaxation time (T_1), spin-spin relaxation time (T_2), and proton density. These atomic properties are manipulated through the setting of TE and TR times, in order to control tissue magnetization. This allows the conditioning of the MR signal for particular tissue types [Huettel et al., 2004, Ch. 5]. During a scan, multiple pulse sequences might be employed, constituting the scan's MRI protocol. In what follows, we elaborate on the commonly obtained images for brain MRI scans.

2.2 Structural MRI

In neuroimaging, Structural or Anatomical MRI refers to images that provide highly detailed spatial information on the tissues forming the brain volume. Their usage has been prevalent for study and diagnosis of disorders such as Alzheimer's disease, Parkinsonian dementias, and Fronto-temporal lobe degeneration [Wattjes, 2011], since the extensive tissue loss usually observed in these pathologies can be visualized *in vivo* using this technique.

The main types of image contrast for structural scans are T_1-weighted, and T_2-weighted and Fluid Attenuated Inversion Recovery (FLAIR). T_1-weighted images represent tissues such as fat and fluids like the cerebrospinal fluid (CSF) as dark, whereas white and gray matter can be distinguished as light and dark gray shades. T_2-weighted contrast causes CSF and other fluids to appear bright, while air appears dark. Gray matter appears as lighter gray, and white matter as darker gray. FLAIR images focus on attenuating fluid signals, causing them to appear dark, which is useful for the clinical diagnostic of a number of neural disorders. Most MRI scan protocols include the acquisition of images with different types of contrasts. Given their distinct characteristics of highlighting specific tissues in opposition to others, the combination of contrast images allows for a broader analysis of a subject's brain structure. They are also essential for evaluating brain functionality, providing the underlying structure for the observed neural activity.

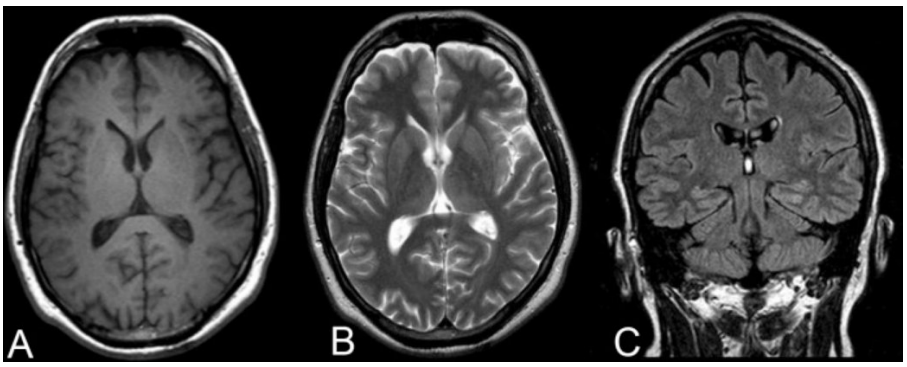


Figure 2.1: T1-weighted (a), T2-weighted (b) and FLAIR (c) MR images [Suoranta et al., 2013].

2.3 Functional MRI

Functional MRI (fMRI) is an imaging technique that generates images of the brain's activity over a period of time. fMRI scans are based on biological mechanisms other than the ones used for structural imaging. Neural signaling processes require energy expenditure in the form of adenosine triphosphate (ATP) [Glover, 2011]. ATP is produced mainly by the glycolytic oxigenation of glucose, a process which generates carbon dioxide as a byproduct. When a brain region is activated, the increase in neural firings and other local processes increase the region's energy requirements, resulting in higher consumption of oxygen. As local oxygen deposits are consumed, vasoactive substances are released, causing blood vessels to dilate in order to restore Oxygen (O2) levels. The increased blood flow delivers even more oxygen than needed to offset its consumption, raising its levels above baseline. The activation process can thus be described in terms of varying levels in oxygenated hemoglobin (HbO2) and deoxygenated hemoglobin (Hb) concentrations. The outset of activation results in a build-up in Hb and a decrease in HbO2 levels [Huettel et al., 2004, Ch. 7]. Within a few seconds of the subsequent vasodilation, an increase in HbO2 and decrease in Hb levels can be observed relative to resting condition. The variation in these concentration signals is described by the Hemodynamic Response Function (HRF), as illustrated in Figure 2.2.

Functional MRI capitalizes on these metabolic processes by using a new method of contrast, called Blood-Oxygen-Level-Dependent (BOLD) contrast. BOLD contrast generates a signal that varies with changes in the magnetic field surrounding red blood cells. While HbO2 is diamagnetic (repulsed by magnetic fields), Hb is paramagnetic (attracted to magnetic fields), since it contains 4 unpaired electrons. Thus, the magnitude of the BOLD signal is modulated by the concentration of Hb molecules in a tissue [Thulborn et al., 1982]. The expected BOLD response signal during activation can be defined based on the HRF. After stimuli is presented, the BOLD signal shows a short initial dip relative to baseline, followed by a peak that lasts for 4 to 7 seconds. Its level then falls below baseline for a few seconds, before returning to baseline levels.

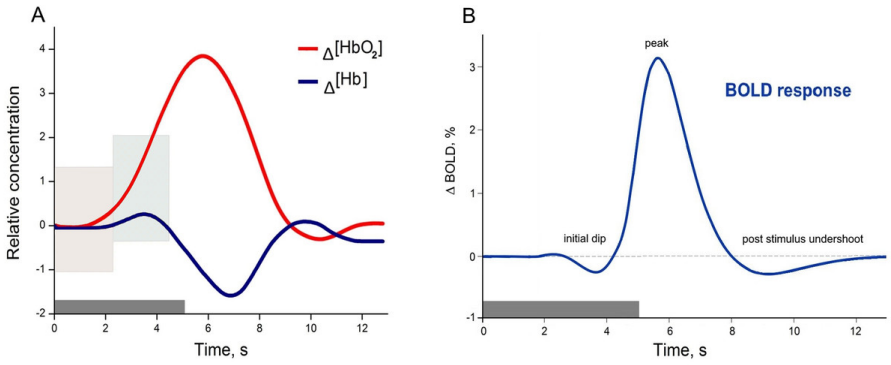


Figure 2.2: Canonical Hemodynamic Response Function (a) and corresponding BOLD response signal [Cinciute, 2019] (b). Stimuli presentation is represented as a grey bar from zero to five seconds.

There are two main categories, or paradigms, of fMRI scanning methods: task fMRI and resting-state fMRI. Task fMRI studies use different *experimental designs* for presenting stimuli to subjects during scans, who can be expected to react passively or to respond with actions such as button pressing, for example. Stimuli may consist of sounds, words, sentences, pictures, images and videos. Experimental designs are assembled to investigate different research hypotheses. On cognitive neuroscience, usual tests seek to evaluate skills such as memory, language processing, decision-making, or to assess emotional responses. By identifying the BOLD response signal throughout the time series, it is possible to infer the brain regions activated by each stimulus, mapping the underlying process of brain activation.

While identifying the BOLD signal in isolation within a region of the brain might sound technically simple, the task of identifying cerebral processes underlying this activation signals is not trivial. The brain never ceases its activity, creating the necessity for designs that maximize contrast between stimuli. On *blocked designs*, stimuli for each condition are assembled in blocks [Huettel et al., 2004, Ch. 11]. Stimuli presentation can alternate between blocks, with the usual inclusion of a rest period between two consecutive trials. Rest periods allow for the acquisition of the brain's baseline signal, which is used as contrast for activation events. In *event-related* designs, different conditions are typically presented in random order, and intervals between stimuli vary in duration.

Resting-state fMRI scans are images of the brain obtained over time where the brain's functionality is analyzed without the presentation of stimuli. The patient is instructed to attempt not to focus on particular thoughts, allowing them to stray freely. Meanwhile, a standard fixed image is shown on screen, consisting of a black background with a white fixation cross on its center, as seen on Figure 2.3.

The analysis of brain connectivity measures during rest led to the discovery of the Default Mode Network (DMN) [Raichle et al., 2001, Greicius et al., 2003], a large-scale connectivity network between certain brain regions which is most active during passive rest and mind-wandering. Later studies have found connectivity abnormalities on the DMN caused by

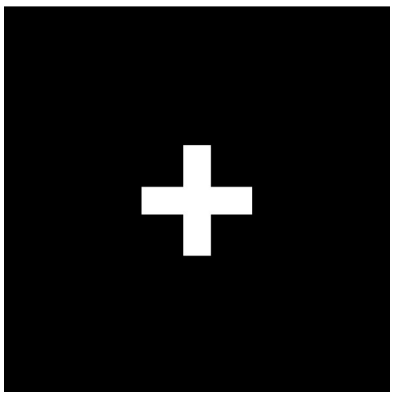


Figure 2.3: Example of the fixation cross presented on screen during resting-state experiments.

conditions such as Alzheimer's [Greicius et al., 2004] and autism [Lynch et al., 2013, Heinsfeld et al., 2018], showing the importance of studying the DMN for improving our understanding on these and many other mental disorders.

2.4 Image Preprocessing

Preprocessing is an essential step for the analysis of MRI images, given the variety and complexity of noise sources that can interfere in scans. These sources consist of thermal and system noise, motion and other physiological noise, and task unrelated activations [Huettel et al., 2004, Ch. 9]. Preprocessing aims at increasing signal-to-noise ratio (SNR) and also preparing data for statistical analysis.

Noise sources

Noise concerns the introduction of uninteresting variability on data. Given the complexity of the systems involved in MRI acquisitions, noise sources are varied. Thermal noise is caused by heating of electrons within the subject and within the scanner, possessing a Gaussian distribution of magnitude over time, and no specific spatial location. System noise can consist either of fluctuations in the generated gradient fields, or of transmitter's or receiver's radiofrequency. The *scanner drift* is an important form of system noise, which causes the MR signal to decrease linearly in magnitude over time due to drifts in the main magnet's static field.

Motion noise is a very common issue that can completely invalidate a scan session. Although more serious noise is caused by voluntary actions such as head movement, natural body processes also generate motion noise, such as breathing and heartbeats. When patient motion is not excessive or erratic, motion correction algorithms can be employed to digitally adjust head position and maintain voxel signals in the same coordinates through the

time series. Particularly for functional scans, other physiological processes can cause interference, such as blood flow and blood volume fluctuations, and oxygen metabolism [Huettel et al., 2004, Chapter 9].

Preprocessing pipeline

Preprocessing steps are applied in order to minimize the above mentioned problems before statistical analysis can be performed [Huettel et al., 2004, Ch. 10]. There is to date no definitive preprocessing pipeline, although the core steps are usually shared across studies, even if applied by different algorithms, software, or in different order. The first core step is applying slice time correction to the data. Most MRI protocols use interleaved slice acquisitions, where the scan collects all the odd numbered slices before moving to even slices. Since slices are acquired sequentially in time, brain regions spanning more than one slice will be collected at different times. These temporal fluctuations are corrected through the interpolation of voxel values. Skull-stripping is then applied, a technique that removes undesired scan artifacts, such as the skull. Since structural and functional scans are acquired independently, they must be spatially aligned to confer robustness to activation interpretability. This step is known as *coregistration*, which is fundamental for associating brain activations with its underlying structure, while also serving as a motion correction technique. Since human brains vary greatly in form, group analysis requires brain images to be normalized to a common *template*, in a process called *spatial normalization*. A commonly used template is MNI-152, constructed using averaged brain MRI images from 152 healthy subjects. Following these steps are temporal and spatial digital filtering, also known as *smoothing*. Filters are methods used to remove undesirable frequency components from a signal [Huettel et al., 2004, Ch. 10]. Since most bodily processes, such as breathing and heartbeats, occur in known frequency ranges, band-pass filters can be used to remove them. Band-pass filters remove certain frequencies from a signal, while leaving the remaining frequencies intact. Low-pass filters, which remove high-end frequencies, can be used for reducing thermal and scanner noises, while also making the signal *smoother*.

When analyzing task fMRI data, interfering activations caused by the brain's perception of stimuli unrelated to the task at hand can occur. Responses to sounds emitted from the scanner, visual stimulation or motor activities all influence BOLD signal contrasts and are captured by the scanning procedures [Huettel et al., 2004, Chapter 8]. Brain stimuli have a predictable effect on the BOLD signal, so that a matched-filter can be used to correlate the signal of interest with observed data, considerably increasing SNR. One form of representing the BOLD signal is the percent signal change (PSC) signal, where the BOLD value for each element of the time-series is represented by its percent variation relative to the signal baseline computed individually for each voxel. Representing BOLD signal through PSC or other statistic metric is fundamental for posterior analysis, since it confers the raw signal with a direct form of unitary comparison between time-points. This use of the indi-

vidual baseline for each voxel is also essential, since BOLD values suffer attenuation the deeper their source is located within the brain. However, PSC signals are highly susceptible to noise sources, such as movement or thermal variability. Thermal variability and other fluctuations from the MRI scanner tend to be random and independent from task stimuli, so that effects can be minimized when analyzing a large sample of scans [Huettel et al., 2004, Ch. 9]. Head movement can occur less randomly, and sometimes show correlations to task experiments. As BOLD related signal changes are very small in magnitude in comparison to movement [Huettel et al., 2004, Ch. 9], it is essential that subjects showing correlations between head movement and task stimuli presentation be excluded from analysis. There is evidence in the literature that PSC signals are reliable in scan-rescan examinations for sensorimotor, cognitive and affective tasks [Schuyler et al., 2010].

The separation of activation signals from the time series is possible through a deconvolution operation between the time series and the presented stimuli onset times. Regarding the analysis of event-related experimental designs, Beta Series Correlation (BSC) analysis can be used model functional connectivity during the different stages of a cognitive task [Rissman et al., 2004]. The method draws on the premise that brain regions in interaction during a given stimulus will exhibit correlation in activity across other stimuli of the same condition. The method uses separate covariates for each trial for constructing a general linear model (GLM), modelling brain activity evoked during each task stimuli. This results in the computation of parameter estimates known as *beta values* attributed to each voxel, which quantify how related a voxel's activity is to each stimulus presentation. Beta values can be correlated for estimating a measure of functional connectivity.

2.5 Graph Theory

Graphs are mathematical structures that model pairwise relationships between elements, and whose field of study is called graph theory. Introduced in 1736 by the famous mathematician Euler [Euler, 1741], graph theory has since become prevalent in most diverse areas of study given its ability to provide efficient data representations and analysis. Let a graph $\mathcal{G}$ be a pair $\mathcal{G} = (\mathcal{N}, \mathcal{E})$, where $\mathcal{N}$ are the *nodes* or *vertices* of $\mathcal{G}$ and $\mathcal{E}$ the *edges*, or *links*, of $\mathcal{G}$. The number of nodes of $\mathcal{G}$ is denoted $N = |\mathcal{N}_\mathcal{G}|$, known as the *order* of $\mathcal{G}$. The number of edges $E = |\mathcal{E}_\mathcal{G}|$ is the *size* of $\mathcal{G}$. Two nodes u and v are neighbors if there exists an edge $e = uv \in \mathcal{G}$. If two edges $e_1 = uv \in \mathcal{E}$ and $e_2 = uw \in \mathcal{E}$ share a common end, they are *adjacent* to each other. A graph is called a *multigraph* if it allows loops $e = uu \in \mathcal{E}$ and parallel or multiple edges $e_1 = uv, e_2 = vu \in \mathcal{E}$. A *directed* graph is one where edges have direction, that is, $e = uv \neq vu \in \mathcal{E}$.

Graphs can be computationally represented through *adjacency matrices* (Figure 2.4), *nxn* integer matrices A with elements A_{ij} valued 1 or 0 depending on whether nodes n_i

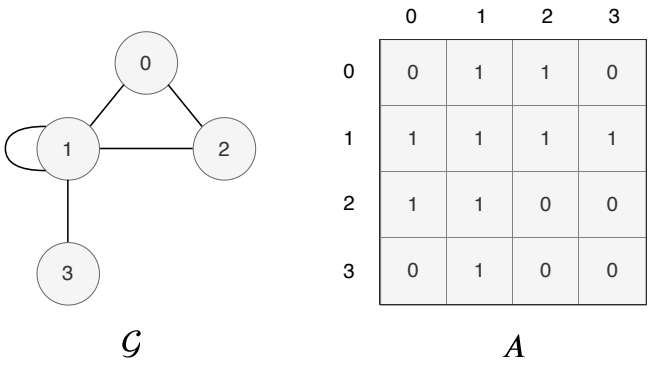


Figure 2.4: A graph $\mathcal{G}$ and its corresponding adjacency matrix A.

and n_j are connected. An adjacency matrix can be weighted, wherein edges are assigned non-binary values. A graph may contain a set of node attributes represented by a matrix $X \in \mathbb{R}^{N \times d}$, where $x_v \in \mathbb{R}^d$ is the d-size feature vector of node v. Likewise, a set of edge attributes can be represented by a matrix $X^e \in \mathbb{R}^{E \times d_e}$, where E is the number of edges and d_e the size of the edge feature vectors. Thus, $x^e_{v,u} \in \mathbb{R}^d_e$ represents the feature vector for edge $e = vu$. A number of metrics can be taken from the analysis of graph topologies:

- Node degree: the degree of a node is the number of edges connected to it. Nodes with high degree are referred to as *hubs*, which are crucial to efficient communication [Freeman, 1977].
- Path length: path length consists of the minimum number of edges connecting two nodes. Shorter mean path lengths result in more efficient networks, providing faster and more robust exchange of information.
- Clustering coefficient: clusters are groups of nodes connected among each other. The clustering coefficient is computed as the proportion between the number of connections among a node's neighbors and the maximum possible number. Complex networks have high clustering, which provides higher local efficiency.
- Motifs: motifs are smaller network building blocks which build up complexity in more evolved networks [Sporns and Kötter, 2004].
- Robustness: the level to which the removal of nodes or connections affects the overall graph topology [Bullmore and Sporns, 2012].

In graphs with random topologies, connections follow a Gaussian distribution [Cohen and Havlin, 2010, Ch. 4]. Its opposite, lattice topologies, consists of nodes whose connections form a regular tiling, resulting in short path lengths and very low global information passing efficiency. The modelling of real phenomena into graphs usually results in *complex networks*, which tend to present high clustering coefficients, modularity, and hierarchical structures[Cohen and Havlin, 2010, Ch. 3]. In particular, the study of complex networks focuses on scale-free and small-world topologies.

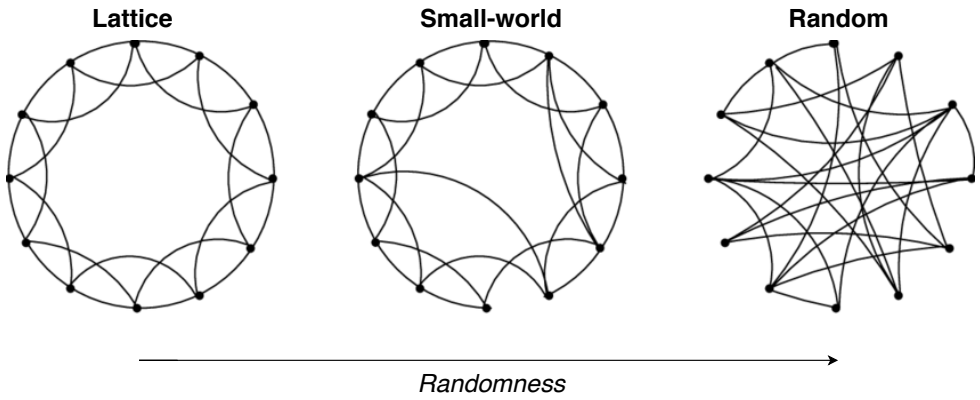


Figure 2.5: Comparison of different graph topologies in relation to a randomness coefficient.

Scale-free networks have power-law degree distributions, lacking a characteristic scale[Cohen and Havlin, 2010, Sec. 4.2]. Power-law distributions are functions where one variable varies as a power of another. Scale-free networks arise when the addition of new nodes is made to high order existing nodes, concentrating a higher number of edges on a few highly connected nodes. Small-world networks combine high local clustering and short path lengths linking these clusters, comprising an intermediate topology between lattice and random networks.

2.6 Human Brain Connectome

Throughout the last decades, much insight was gained on the relationship between the human brain's structure and functionality, with graph theory being one of the tools used to represent such relations [Bullmore and Sporns, 2009]. Brain functionality stems from the capacity of neurons to exchange information through electrochemical signals, known as synapses. When observed from a macroscopic scale, synaptic communications have been shown to constitute complex networking behavior spanning different brain regions. The mapping of these networks is known as the human *connectome*. Neurons transmit information through long nerve fibers called axons or tracts, which constitute the brain's white matter, and physically connect different brain regions. As such, initial attempts to compute the connectome focused on these structural connection, analyzing patterns from correlations in cortical thickness and volume of individual brains, which could signalize the presence of neural pathways. These analyses resulted in the construction of brain network graphs, wherein regions of interest (ROIs) were represented as nodes and their connections as edges.

Clearer mapping was achieved with the advent of *diffusion tensor imaging* [Le Bihan et al., 2001], a technique that generates contrast MR images from the diffusion of water molecules through the axons. The analysis of DTI images allowed the generation of

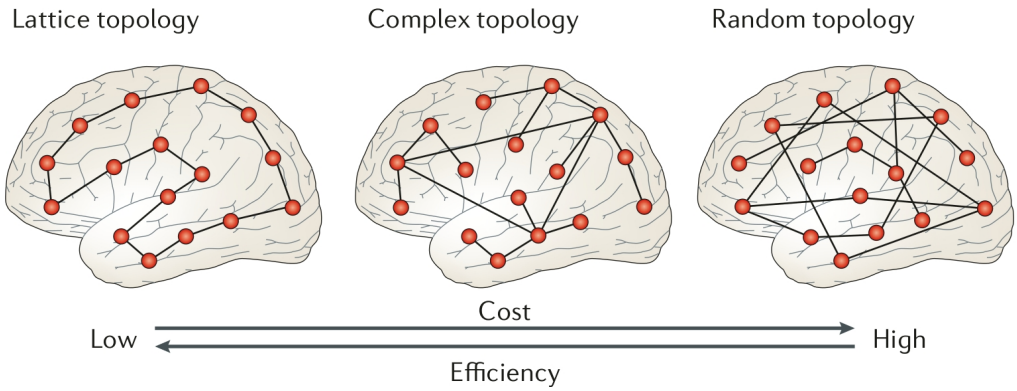


Figure 2.6: Effects of different brain wirings on cost and efficiency of information processing [Bullmore and Sporns, 2012].

structural connectomes taking into account the number of nerve tracts connecting different regions. More recently, fMRI scans have been used for composing *functional connectomes*, which carry information on the synchronicity of different brain regions via brain signal cross-correlations [Varoquaux and Craddock, 2013].

Analyses of fMRI connectomes have found that, as with most real world systems, the human brain has complex network characteristics, which present a middle ground concerning efficiency to cost trade-offs, as seen in Figure 2.6. The brain has small-world properties, a topology that allows for both modular local processing and global distributed processing [Achard et al., 2006], increasing efficiency in information exchange. Maintenance of these properties is essential for cognition, as small-world networks have been found to be disrupted by neurological disorders, such as schizophrenia [Bassett et al., 2008, Liu et al., 2008]. However, a more complete understanding of the mechanisms responsible for the emergence of brain functionality, and of the dynamic interactions between brain regions that modulate different mental processes and disorders, still constitutes a challenge (see Chapter 6). As such, the proposal of new tools and techniques that can provide reliable insights into these matters is of great importance to the field.

3. MACHINE LEARNING BACKGROUND

In this chapter, we provide a background on machine learning key concepts and techniques. We introduce basic concepts in Section 3.2, following with an overview on artificial neural networks in Section 3.3. We discuss performance evaluation methods in Section 3.4, regularization in Section 3.5, and learning and optimization processes in Section 3.6. We present an overview on deep learning in Section 3.7, and detail Graph convolutional networks in Section 3.8.

3.1 Machine Learning

Machine Learning (ML) is a branch of Artificial Intelligence (AI) that aims to develop computer programs that automatically improve with experience [Mitchell, 1997, Preface]. It consists of a set of algorithms used for learning mathematical models that describe observed data patterns, being applied primarily to problems in which solutions are too complex to be achieved with regular programming. Machine learning algorithms capitalize on the availability of vast amounts of data, a resource which has been generated in exponential fashion over the last decades due to the ever increasing prevalence of technology in human life [Makridakis, 2017].

ML-based applications have increased in number in recent years, being used in tasks such as image classification and generation [Krizhevsky et al., 2012, Goodfellow et al., 2014], natural language processing [Vaswani et al., 2017], speech recognition [Graves et al., 2013] and content recommendation [Zhang et al., 2019a]. Apart from the extensive commercial interests of big tech companies and startups alike, ML has also been increasingly employed in a variety of scientific research fields, including neuroscience [Richards et al., 2019].

3.2 Basic Concepts

Machine Learning can be split into three major learning paradigms: supervised learning, unsupervised learning and reinforcement learning. In this work, we will focus on the former two We also discuss semi-supervised learning, a method in between the supervised and unsupervised paradigms.

Supervised learning consists of algorithms that rely on datasets of I labeled example pairs $(\vec{x}_I, y_I)$, where $\vec{x}_i \in X$ refers to the feature vector of the i-th element of the example set X, and $y_i \in Y$ to its respective label from label set Y. The observation of these exam-

ples allows for the learning models to predict the relationship between pairs (i.e. $P(Y \mid X)$, that is, being able to infer the value of Y given X. Unsupervised learning algorithms are applied to data with unlabeled examples in order to learn information about the data's distribution. Semi-supervised learning consists of using both labeled and unlabeled examples for jointly optimizing predictions. Reinforcement learning is a technique based on a trial-and-error learning method, where an agent interacts with an environment by taking actions and observing rewards [Sutton and Barto, 2014, Ch. 1].

3.3 Artificial Neural Networks

Artificial Neural Networks (ANNs) [Tan et al., 2005, Sec. 5.4] machine learning models created as an attempt to simulate the workings of biological neural systems. The building block of ANNs is the artificial *neuron*, also known as the network nodes. The simplest model, consisting of a single neuron, is called the *perceptron*. The perceptron, illustrated in Figure 3.1 consists of input nodes representing input example features, and a layer of output nodes, representing the model output. Each input node has a weighted connection to the output node, whose optimal value is learned during training. A perceptron computes its predicted output $\hat{y}$ for a given set of input features x_j, where $j = 1, ..., d$ represents the j-th position of the feature vector, by performing their weighted sum with the addition of a bias value b, as seen on Equation 3.1:

$$v(x) = \sum_{j=1}^{d} w_j x_j + b \tag{3.1}$$

$$\hat{y} = o(x) = \varphi(v(x)) \tag{3.2}$$

The w_j variable refers to the weight values associated with each feature. A bias value is applied globally to the output, being modeled as the weight coming from an extra node. The φ function in Equation 3.2 is an activation function, which is predefined in order to adjust the model behavior.

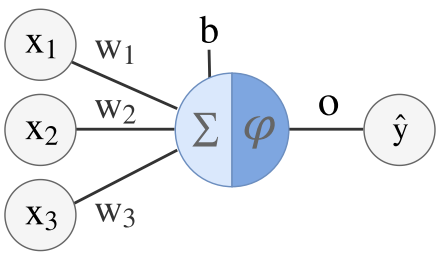


Figure 3.1: A perceptron network with 3 inputs.

Neural networks with only a single layer of neurons between input and output can only approximate linear functions, in which case they always converge to an optimal solution

for linearly separable classification problems. However, in the case of non-linear problems, such as the XOR gate classification, more complexity must be introduced to the model.

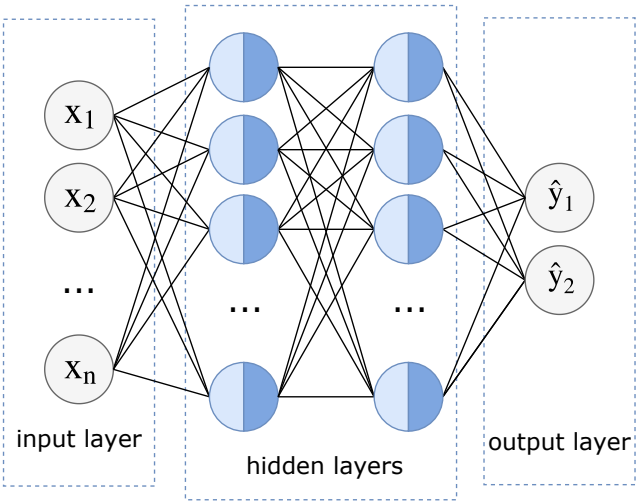


Figure 3.2: A multilayer perceptron (MLP) network.

This can be achieved with Multilayer Perceptrons (MLPs), networks with one or more intermediate *hidden layers* between input and output, each containing an arbitrary number of *hidden nodes* as illustrated in Figure 3.2. MLPs are *feed-forward*, meaning outputs from one layer become the input to the following one. The network is called *fully connected* (FC) if each neuron from layer i is connected to every neuron of layer $i+1$.

Non-linear activation functions are an essential component of modern ANNs. These functions introduce non-linear relationships between inputs and outputs, allowing the modeling of more complex functions. Examples of such functions are the sigmoid, hyperbolic tangent and the Rectified Linear Unit (RELU). In practice, these functions map inputs with values ranging from $\{-\infty, +\infty\}$ to the interval $\{0, 1\}$ or $\{-1, 1\}$, where a non-linear decision boundary is set up.

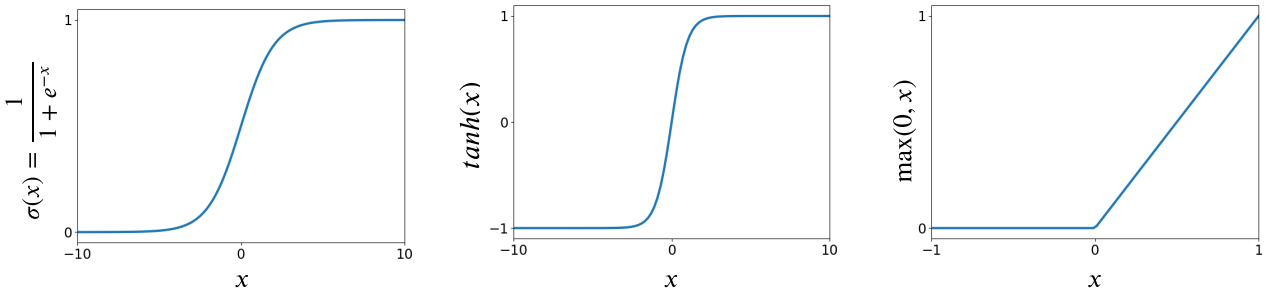


Figure 3.3: Sigmoid, hyperbolic tangent and RELU activation functions.

3.4 Performance Evaluation

Performance for regression tasks is usually measured with RMSE (root-mean-square error), which is the square root of the average squared error between $\hat{y}_i$ and y_i, where $\hat{y}_i$ is the predicted label and y_i is the true label for example $\vec{x}_i$. RMSE informs the magnitude of the error presented by the model's predictions. When considering classification tasks, however, different measures must be employed in order to account for the decision process made between discrete values, or classes, instead of continuous ones.

For a binary classification task, two types of errors can be identified: false positive (*FP*), or type I errors, where a negative class object is predicted as positive; and false negatives (*FN*), type II errors, where a positive class object is predicted as negative [Tan et al., 2005, Sec. 4.2]. Accuracy, the most commonly used measure, measures the percentage of examples correctly classified, that is, the number of correct predictions (True Positives (*TP*) + True Negatives (*TN*)) divided by the total number of examples. When training models using datasets with unbalanced classes, where the incidence of one class dominates the others, measures such as precision, recall, or F1-score are preferred over accuracy. Precision, defined as $p = TP/(TP + FP)$, serves as an indication of the model's exactness. The higher the precision, the lower the number of false positives accused. Recall, defined as $r = TP/(TP + FN)$ measures the number of correctly classified positive examples. High recall means few false negative predictions. F1-score represents the harmonic mean between recall and precision [Tan et al., 2005, Section 5.7.1], so that $F_1 = (2 \times Precision \times Recall)/(Precision + Recall)$.

The Receiver Operating Characteristic (ROC) Curve is a graphical tool for visualizing the trade-off between true positive rate (TPR) and false negative rate (FNR). Models with that perform well in discriminating between classes have curves closer to the upper left corner of the diagram, while models classifying examples by chance generate a curve residing in the main diagonal [Tan et al., 2005, Section 5.7.2]. The Area Under the Curve (AUC) score can be computed as metric of a models performance, as is a useful tool for comparing different models. A perfect model has $AUC = 1$, while a model performing random guesses has $AUC = 0.5$.

The task of classifying medical conditions is an example of the usefulness of these measures. If a certain condition occurs at a rate of 1%, a model which predicts all examples as false will accurately predict 99% of the examples, even if failing to classify all true positive cases. In such a case, measuring recall would indicate that true positives are not being correctly identified, and thus the poor performance of the model.

3.5 Regularization

A successful model is capable not only of fitting a function to the training data, but also of generalizing its predictions to data it has never seen. That is, both training error and generalization errors must be low [Tan et al., 2005, Sec. 4.4]. If a model performs too well on the training set while performing poorly in the test set, its generalization capacity is affected. Such a situation is called *overfitting*. Overfitting happens when a model becomes too complex, since any given dataset with n points can be perfectly approximated by a polynomial of n-degree. Such a close adjustment to the training data leads to a reduction in generalization power, affecting the classification or predictions over unseen data made by the model.

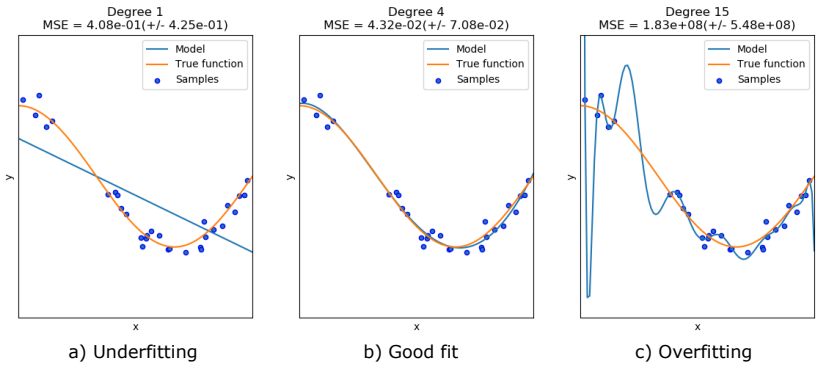


Figure 3.4: Underfitted (a) and overfitted (c) models contrasted to an ideal one (b). Above each figure is the correspondent polynomial degrees and mean square errors.

On the other hand, excessively simple models lead to *underfitting*. In this case, there is a lack in parameters, causing the model to be unable to learn enough information about the observed data and consequently to fail in capturing the underlying distribution of the data as to reliably perform predictions. Therefore, controlling model complexity is fundamental for achieving good performance. Regularization techniques consist of pruning the model's complexity in order to reduce overfitting. One often used approach is the addition of a penalty coefficient to the cost function, so as to penalize parameters that reach too high values [Bishop, 2006, Section 1.1]. When concerning artificial neural networks, this approach is known as weight decay. Regularization in ANNs is usually complemented with *dropout*, a computationally inexpensive technique that consists of randomly ignoring non-output neurons during training [Srivastava et al., 2014]. This process forces neurons to become less specialized to specific features of the input, improving the network's generalization power.

3.6 Learning and Optimization

Supervised machine learning models are composed of parameters or weights **w** that process an input $\vec{x_i}$ in order to output a prediction for the input's label. Thus, the goal of training a machine learning model is to determine the weights **w** which minimize the prediction error E observed for the set of training inputs X, that is:

$$\mathbf{w}^* = arg\min_{\mathbf{w}} \mathrm{E}(\mathbf{w}|X) \tag{3.3}$$

The error E is a measure of a model's predictive inaccuracy for a given input $\vec{x_i} \in X$. Error magnitudes are computed by an *error function* or *loss function*.

For binary classification, although prediction targets are boolean, the target function to be learned can be modeled as a probability that a given input instance belongs to a class. The *binary cross entropy* (BCE) loss function can be used to process such probability, where the error E(**w**) is defined as:

$$\mathrm{E}(\mathbf{w}) = -\frac{1}{N}\sum_{i=1}^{N} y_i\, log(\,\hat{y}_i\,) + (1 - y_i)\, log(1 - \hat{y}_i), \tag{3.4}$$

where y_i is the true label for example $\vec{x_i}$ and $\hat{y}_i$ is the predicted label. After computing the error, an optimization algorithm is used to minimize it. Gradient descent (GD) is the iterative optimization algorithm most commonly employed to that end. GD computes the gradient vector pointing to the direction of greatest increase to the error. The negative of the gradient leads to the direction of minimization of E(**w**), informing the model whether to increase or decrease each weight [Alpaydin, 2010, Chapter 10, Section 6]. If E(**w**) is a differentiable function, the gradient vector is composed of the partial derivatives of the error with regard to each weight parameter:

$$\nabla_w E = \left[\frac{\partial E}{\partial w_1}, \frac{\partial E}{\partial w_2}, ..., \frac{\partial E}{\partial w_d}, \right]^T \tag{3.5}$$

The minimization procedure begins with a random initialization of weights, which are updated at each step in the opposite direction of the computed gradients:

$$\Delta w_i = -\alpha \frac{\partial E}{\partial w_i}, \tag{3.6}$$

$$w_i = w_i + \Delta w_i, \tag{3.7}$$

where α is the *learning rate*, which determines the *stepsize*, or the magnitude of the movement to be made in that direction. The algorithm terminates when the derivative is zero,

meaning a *critical* or *stationary* point has been reached. A *local minimum* is a stationary point where the function E(**w**) has a lower value than its neighboring points and thus can not be decreased with infinitesimal steps [Goodfellow et al., 2016, Section 4.3]. Its opposite, the *local maximum*, is a point where the E(**w**) is the highest among its neighbors, so it can not be increased with infinitesimal steps. There are also *saddle points*, which are neither minima nor maxima. Figure 3.5 illustrates each type.

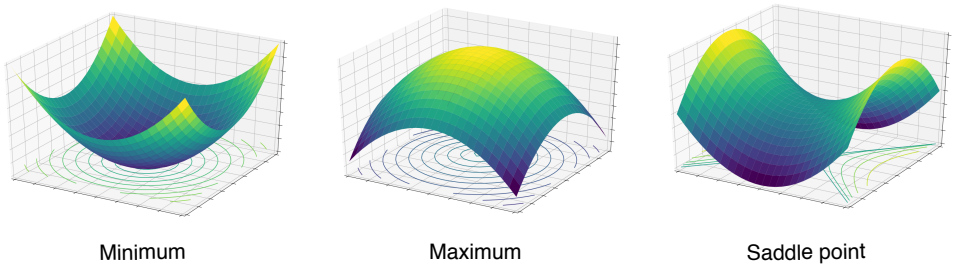


Figure 3.5: Visualization of different critical points.

The lowest possible value for E(**w**) is the *global minimum*, which could exist at a single or multiple points. The task of finding global minima is difficult, since the function could get trapped in local minima or saddle points. Thus, optimization usually settles for finding a value that approximates a global minimum as much as possible. The correct setting of the learning rate is essential to ensure that the optimizer can find a minimum. A larger value means faster optimization, but the chances of missing a minimum increase, which might make the algorithm unable to converge. Thus, the learning rate should be fine-tuned to values low enough to guarantee precision, but high enough to provide adequate optimization times. As it is a value defined before the training process, the learning rate is referred to as a *hyperparameter*.

The GD computation for hidden nodes is, however, nontrivial. In order to assess their partial derivatives for the error term, their output values must be known. *Backpropagation* is an algorithm devised to address this issue for neural network optimization. The algorithm works in two phases: the forward pass and the backward pass. In the first iteration, weights are initialized randomly, and a forward pass propagates the input through the hidden layers to generate an output $\hat{y}$. During the backward pass, the algorithm computes the partial derivatives of the error, and updates the weights by applying equations 3.6 and 3.7. Through this procedure, weights from layer $n+1$ are updated prior to those of layer n.

To compute the gradients of each neuron regarding its weights, the chain rule is applied:

$$\frac{\partial z}{\partial x} = \frac{\partial z}{\partial y}\frac{\partial y}{\partial x} \tag{3.8}$$

The chain rule states that if a variable z depends on the variable y, which depends on the variable x, then the relationship between z and x can be found by breaking down the intermediary relationships. Thus, the effect of a specific weight w_{hj} w.r.t. to the error E is:

$$\frac{\partial E}{\partial w_{hj}} = \frac{\partial E}{\partial y_i} \frac{\partial y_i}{\partial x_h} \frac{\partial x_h}{\partial w_{hj}}, \tag{3.9}$$

where y_i refers to an output neuron, and x_h an input feature. An iteration of the backpropagation algorithm is called an *epoch*. The optimization for each epoch can be computed over all the training data, in process known as batch gradient descent (BGD) [Wilson and Martinez, 2003]. When dealing with large datasets, however, taking all examples into consideration might be computationally expensive. Stochastic gradient descent (SGD) is an alternative method, which approximates the gradients using small sets of examples, known as a *mini-batch* [Goodfellow et al., 2016, Sec. 5.9]. SGD allows training on very large datasets within reasonable time. Since it evaluates a smaller number of examples, it captures more fluctuations in the gradients, forming a zigzag pattern of descent towards the minimum. As an approximation, SGD and is still prone to getting stuck at local minima or plateaus, and may not converge or be slow at doing so.

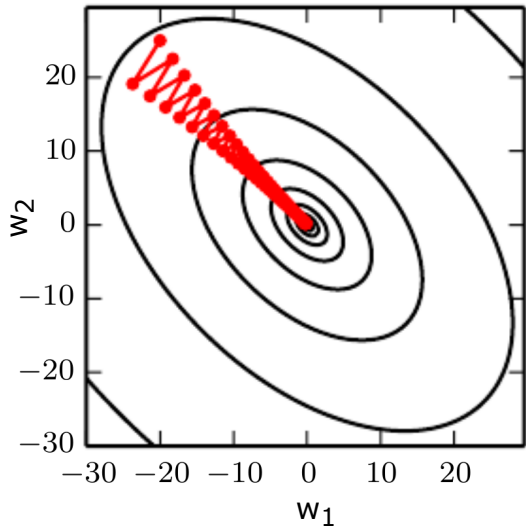


Figure 3.6: Optimization patterns formed by GD without momentum [Goodfellow et al., 2016].

Other methods can be employed to improve optimization, such as *momentum* and *learning rate adaptation*. Momentum [Sutskever et al., 2013] applies a *velocity* factor to the weight update formula 3.7, which aids gradients computed in the same direction as the previous one, and hinders gradients in the opposite direction. Apart from making convergence faster, it helps to prevent the function from getting stuck at local minima and plateaus. Learning rate adaptation works by defining per-parameter learning rates, which are updated as learning progresses. *Adam* (Adaptive Moment Estimation) [Kingma and Ba, 2014] is an optimizer which combines per-parameter learning rates with momentum-like parameters,

and has become the standard optimizer for *deep learning* algorithms, which are covered in the following section 3.7.

In spite of its usefulness, back propagation suffers from a few issues, such as *exploding* and *vanishing* gradients [Pascanu et al., 2013]. *Exploding gradients* refer to the exponential increase in the error derivative of some terms, resulting in large weight updates which make the model unstable. *Vanishing gradients* is the opposite problem, where the computed gradients slowly converge to 0 in value, causing updates to ignore weight values from the initial layers. This problem is found in deeper models with many hidden layers, where activation functions such as sigmoid can further contribute to the exponential gradient decrease. Other functions, such as RELU, are more appropriate in these case. The derivative of the RELU function is always equal to 1 for positive values, helping to better sustain the gradient throughout the model. This is one of the major reasons why RELU has become the standard activation function for deep learning models.

3.7 Deep Learning

Machine learning algorithms face difficulties when dealing with data constituted of many dimensions, a problem known as *the curse of dimensionality*. Thus, complex problems such as speech or object recognition have been historically difficult for traditional AI algorithms. *Deep Learning* is a subfield of machine learning algorithms consisting of models with a large number of layers for learning high-level feature representations. As described in section 3.3, MLP models can be comprised of numerous layers, and therefore considered deep models. Nevertheless, such a model would suffer great limitations when applied to high-dimensional data, computing $O(N^2)$ parameters for data of dimension N. For example, consider the task of classifying images of size 150x150 pixels with an MLP consisting of a single hidden layer with 100 neurons. In order to feed the network the image must be flattened into a 1-D vector, which when multiplied by the neurons of the hidden layer would result in more than 2 million parameters. As stated in section 3.4, too many parameters make networks prone to overfitting, which can lead the model to be computationally expensive, as well as poor in performance.

3.7.1 Convolutional Neural Networks

Convolutional Neural Networks (CNNs) are models that overcome the issues presented above [Lecun, 1989]. CNNs became popular in 2012 with AlexNet [Krizhevsky et al., 2012], a network famous for beating that year's ImageNet LSVRC contest with an incredible margin towards other participants, consequently obtaining state-of-the-art status for image

classification tasks. CNNs are neural networks specialized in processing data with grid-like structure and presenting translational invariance regarding this grid. Images (2D or 3D matrices) and time-series (1D vectors) are examples of data with such characteristics [Goodfellow et al., 2016, Ch. 9]. If a signal is defined as something that conveys information, the Convolution Theorem states [Hayes, 1998] that convolutions are linear mathematical operations that combine two input signals into an output signal [Hayes, 1998]. The basic block of CNNs is the *convolutional layer*.

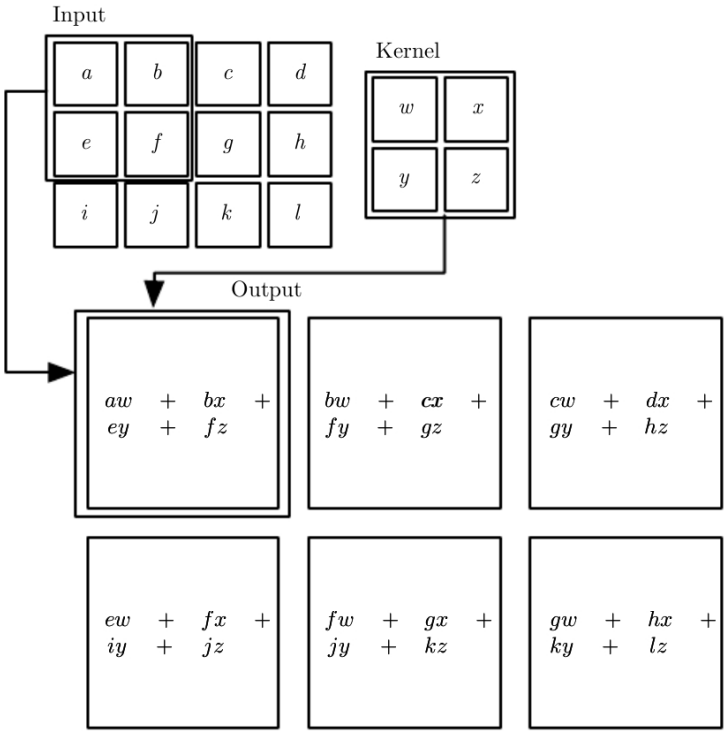


Figure 3.7: Visualization of a convolution operation [Goodfellow et al., 2016].

A convolutional layer can be described as possessing three stages. In the *convolutional stage*, inputs are convolved with filters, also known as *kernels*. Kernels are matrices or vectors initialized with random parameters, which are optimized during training to provide the extraction of relevant features. The output formed by these features is called a *feature map*. By using a kernel much smaller than the input, we can achieve an extensive reduction in the number of parameters, allowing for more efficient generalization. The kernel moves through the input from left to right, top to bottom, and each of its elements is multiplied by the input element occupying the same position. Thus, the same kernel parameters are applied to the whole input. Zero-padding can be added to the input edges to prevent loss of information in those regions. Since kernels are applied to small local regions one at a time, they introduce a *locality relational bias* to the model [Battaglia et al., 2018]. This is desirable for tasks involving image data, which present high local covariance (elements in proximity contain similar information).

In the *detector stage*, a nonlinear activation function is applied point-wise to the feature maps resulting of the first stage. The third stage is the *pooling stage*, where a pooling function is applied. Pooling functions apply statistical operations to the output in order to reduce its variability and parameter number. It also switches spatial resolution to feature resolution, which encode higher-level information as data representations get deeper. The output of the pooling stage, and thus of the convolutional layer is usually flattened and fed to a fully-connected layer, responsible for the classification of the pooled features. The use of the fully-connected layer enables each input element to interact with every possible output, as well as allowing end-to-end learning.

3.7.2 Siamese Networks

Siamese networks are models specialized in computing similarity metrics for different data modalities [Bromley et al., 1993]. These networks consist of two twin sub-networks with shared weights, as illustrated in Figure 3.8. Each sub-network is fed a different input, and their outputs are concatenated and compared. Because of the shared weights, similar inputs are guaranteed to have similar outputs. During training, the label values for each class are not used, as the model only checks if labels are the same or not. As such, these networks learn what feature-embeddings differentiate distinct classes, instead of trying to classify new examples.

In order to train such a model, a *constrastive loss function* [Hadsell et al., 2006] can be employed, computing error as:

$$E = E_s + E_d = (1 - Y)\frac{1}{2}(D_W)^2 + (Y)\frac{1}{2}\{\max(0, m - D_W)\}^2, \tag{3.10}$$

where D_W is the Euclidian distance between outputs, and m is a margin value greater than 0. The value of Y is equal to 0 if the inputs belong to the same class, and 1 if otherwise. As such, the first partial term, E_s acts by minimizing D_W between *similar* examples. The second partial term, E_d acts by maximizing D_W between *dissimilar* examples. This is achieved through the *max* function and the margin m, which acts by turning distance values predicted as small by the network into larger values, which can be optimized. When $D_W \geq m$, the resulting gradient will be 0, meaning the distinct examples have received a high enough value of dissimilarity.

Siamese Convolutional Networks (SCNs) have been successfully applied in tasks such as face recognition [Chopra et al., 2005] and one-shot image classification [Koch, 2015], where an SCN was able to differentiate between 40 different classes after training with a single example per class. The ability to achieve state-of-the-art results with few training examples allows many applications to benefit from the use of SCNs, even though their training is slower and more irregular than that of CNNs. In comparison with softmax-based

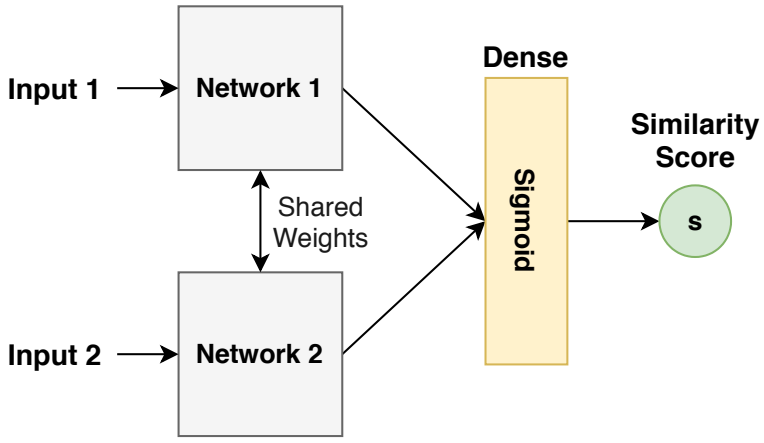


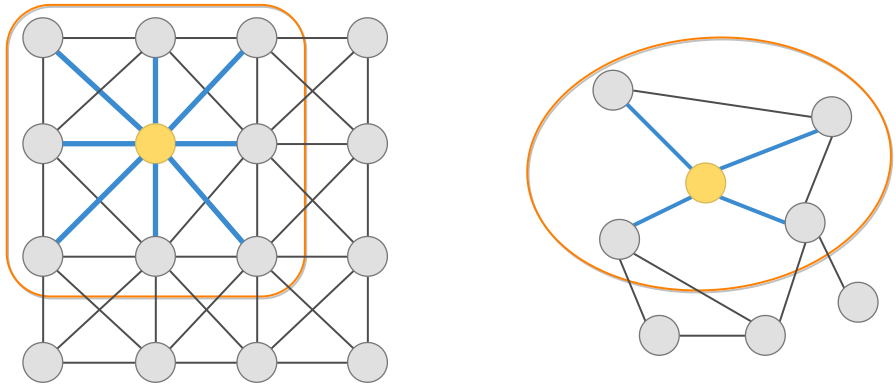
Figure 3.8: Siamese network architecture.

classification models, siamese networks show better or competitive performance in small datasets, or when the number of classes is very large [Horiguchi et al., 2020].

3.8 Graph Convolutional Networks

The convolution operations of CNNs are applied to grid structures, lacking the ability to distinguish non-euclidean geometrical relations, such as graphs. In order to deal with graphs, CNNs require preprocessing steps that transform graph data to simpler representations, which means valuable information contained in the graph structure may be lost.

Graph Convolutional Networks (GCNs) are networks that aim to generalize grid convolutional layers for graph structured data, as illustrated in Figure 3.9. There are two main branches of graph convolutions: spatial graph convolutions and spectral graph convolutions [Wu et al., 2020]. Spatial approaches apply convolutions based on each node's spatial configuration, employing different forms of message passing mechanisms to propagate node values to their neighbors along their edges. Spectral approaches draw on *spectral graph theory* to represent graph information in terms of *eigenvectors* and *eigenvalues* related to its corresponding *adjacency matrix* and *Laplacian matrix* [Bruna et al., 2013]. The eigendecomposition of a matrix, that is, its decomposition into a set of eigenvectors and eigenvalues, allows for analysis of certain matricial properties that are not immediately apparent [Goodfellow et al., 2016, Sec. 2.7]. The Laplacian matrix is a symmetric, positive semi-definite matrix (all values are non-negative), and its eigenvalues are closely related to the graph's topology. Kernels are based on a K-th order polynomial function of the eigenvalues, and convolutions are applied to the K-th level neighborhood of each node.



(a) Grid convolution (b) Graph convolution

Figure 3.9: Comparison between convolution operations on a grid (a) and on a graph (b).

3.8.1 Spectral Graph Convolutions

In order to learn the filters on the spectral graph domain, the graph topological information is decomposed into a Fourier basis, in a process closely related to the decomposition of audio or video signals into its basic frequency components. Instead of the usual decomposition of signals into sine functions, spectral filters are decomposed in relation to their set of eigenvectors. This allows the definition of a graph Fourier Transform, which transforms the graph to the spectral domain where convolutions can be applied.

Let a graph be defined as $\mathcal{G} = (\mathcal{N}, \mathcal{E})$, where $\mathcal{N}$ is the set of nodes and $\mathcal{E}$ the set of edges, which can be represented by the adjacency matrix A encoding the connections between nodes. The elements of A can be either binary, representing presence or absence of connections, or continuous, representing connection weights. The Laplacian of $\mathcal{G}$ is defined as $L = D - A \in \mathbb{R}^{N \times N}$, where N is the total number of nodes and $D \in \mathbb{R}^{N \times N}$ is a diagonal degree matrix with entries $D_{ii} = \sum_j A_{i,j}$. The normalized Laplacian is $L = I_n - D^{-1/2} A D^{-1/2}$, where I_n is an identity matrix, and contains a complete set of orthonormal eigenvectors $U = \{u\}_{l=0}^{n-1} \in \mathbb{R}^N$, and its corresponding set of eigenvalues $\Lambda = \{\lambda\}_{l=0}^{n-1} \in \mathbb{R}^N$. The Laplacian is described in the Fourier basis of U so that $L = U \Lambda U^\top$, allowing for the graph Fourier Transform of a signal $x \in \mathbb{R}^N$ to be defined as $\hat{x} = U^\top x \in \mathbb{R}^N$. This definition is first used in the ChebNet model [Defferrard et al., 2016], which has become the standard spectral GCN approach, and has been employed to a great extent in fMRI studies (see Chapter 6. ChebNet generalizes the filtering operation on Euclidean spaces to the spectral domain, where the filtering of a signal x by a kernel g_θ is defined as:

$$y = g_\theta(L)x = g_\theta(U \Lambda U^T)x = U g_\theta(\Lambda) U^\top x, \tag{3.11}$$

where $g_\theta(\Lambda)$ can be understood as a function of the eigenvalues of L, and $\theta \in \mathbb{R}^N$ is a vector of Fourier coefficients. Kernels can be localized in space through the use of polynomial kernels of the form:

$$g_\theta(\Lambda) = \sum_{k=0}^{K-1} \theta_k \Lambda^k, \tag{3.12}$$

where $\theta \in \mathbb{R}^K$ is a vector of polynomial coefficients of order K. In this formulation, the value at node j of kernel g_θ centered at node i is:

$$(g_\theta(L))_{i,j} = \sum_k \theta_k (L^k)_{i,j} \tag{3.13}$$

If $d_G(i,j)$ is the shortest path distance between two nodes, $d_G(i,j) > K \implies (L^k)_{i,j} = 0$. Thus, a kernel composed of K-th order polynomials is K-localized on its neighborhood, and its learning complexity is $\mathcal{O}(K)$. Nevertheless, the complexity of applying such a kernel to a signal is still $\mathcal{O}(n^2)$. To circumvent the high complexity issue, the use of Chebyshev polynomials has been proposed, allowing for the recursive computation of the parameters of kernel g_θ [Hammond et al., 2009]. A Chebyshev polynomial $T_k(x)$ of order k can be computed by the recurrence $T_k = 2xT_{k-1}(x) - T_{k-2}(x)$, with $T_0 = 1$ and $T_1 = x$. The convolution operation then becomes:

$$y = g_\theta * x = \sum_{k=0}^{K-1} \theta_k T_k(\tilde{L}) x, \tag{3.14}$$

where $T_k(\tilde{L}) \in \mathbb{R}^{N \times N}$ is the k-th order Chebyshev polynomial computed at the scaled Laplacian $\tilde{L} = 2L\lambda_{\max} - I_n$, which consists of eingenvalues in the $[-1, 1]$ range. By defining $\bar{x}_k = T_k(\tilde{L})x \in \mathbb{R}^N$, the recurrence can be computed as $\bar{x}_k = 2\tilde{L}\bar{x}_{k-1} - \bar{x}_{k-2}$. Thus, the filtering operation is reduced to cost $\mathcal{O}(K|\mathcal{E}|)$. The resulting feature map for the sample s of a mini-batch of S examples is defined as:

$$y_{s,j} = \sum_{i=1}^{F_{in}} g_{\theta_{i,j}}(L) x_{s,i} \in \mathbb{R}^n, \tag{3.15}$$

where $x_{s,i}$ are the input features, and F_{in} refers to vectors $F_{in} \times F_{out}$ of Chebyshev coefficients, which are the learnable parameters. Training is done with backpropagation, computing two gradients:

$$\frac{\partial E}{\partial \theta_{i,j}} = \sum_{s=1}^{S} [\bar{x}_{s,i,0}, ..., \bar{x}_{s,i,K-1}]^T \frac{\partial E}{\partial y_{s,j}} \quad \text{and} \quad \frac{\partial E}{\partial x_{s,i}} = \sum_{j=1}^{F_{out}} g_{\theta_{i,j}}(L) \frac{\partial E}{\partial y_{s,j}}, \tag{3.16}$$

where E is the error computed for mini-batch S. The resulting set of operations consist of K sparse matrix-vector multiplications and one dense matrix-vector multiplication. To prevent overfitting and further reduce computational complexity, a first-order approximation of these operations is introduced in [Kipf and Welling, 2016] using limited kernel size of $K = 1$ and

maximum eingenvalues approximated to $\lambda_{max} \approx 2$. These changes allow equation 3.14 to be rewritten as:

$$y = g_{\theta'} * x = \theta'_0 x + \theta'_1 (L - I_N) x = \theta'_0 x + \theta'_1 D^{-\frac{1}{2}} A D^{-\frac{1}{2}} x, \tag{3.17}$$

with θ'_0 and θ'_1 as free parameters. Computations can be further simplified by assuming a single parameter $\theta = \theta'_0 = -\theta'_1$, resulting in a convolution operation for a signal $X \in \mathbb{R}^{N \times C}$ for a graph of N nodes $\mathcal{N}$ and C input channels defined as:

$$y = \tilde{D}^{-\frac{1}{2}} \tilde{A} \tilde{D}^{-\frac{1}{2}} X \Theta, \tag{3.18}$$

where $\tilde{D}$ is the diagonal degree matrix for the adjacency matrix with added self loops $\tilde{A} = A + I_N$. The resulting filtering operation has complexity $\mathcal{O}(|\varepsilon| FC)$ for F output feature maps [Kipf and Welling, 2016]. The computation of equation 3.18 can be performed through the product of matrices $\tilde{A}$ and X, which could be interpreted as bridging spectral and spatial graph convolutions.

3.8.2 Spatial-Temporal Graph Convolutions

Spatial-Temporal Graph Convolutional Networks (ST-GCNs) are models that analyze a graph's spatial node relationships along with temporal information. The addition of time information in the convolution operations allows for more accurate processing of graphs with dynamic behavior, a very useful characteristic for a variety of domains such as traffic forecasting [Yu et al., 2018, Li et al., 2018] and body motion detection [Yan et al., 2018], where the time component cannot be efficiently represented by statistical methods. A spatial-temporal graph can be described as a graph where each node is connected to itself across T time points. Let a stationary spatial graph $\mathcal{G}$ be defined as $\mathcal{G} = (\mathcal{N}, \mathcal{E})$. A spatial-temporal graph $\mathcal{G}_{ST}$ can be defined as $\mathcal{G}_{ST} = (\mathcal{N}, \mathcal{E}, \mathcal{T})$, where $\mathcal{T}$ is the set of temporal elements associated with each node. Thus, the node set $\mathcal{N} = \{n_{it} | i = 1, ..., N; t = 1, ..., T\}$ includes both N nodes composing the graph in space and each of their corresponding T neighbors in time, so that node n_{it} represents the i-th node on time-point t.

ST-GCNs approaches can follow architectures based on Recurrent Neural Networks (RNNs) or CNN models [Wu et al., 2020]. RNN-based approaches combine different types of recurrent units, such as LTSMs [Seo et al., 2016] or diffusion layers [Li et al., 2018] to process time information along with graph convolution layers. CNN-based approaches apply 1D convolutions to learn temporal information after graph convolution layers have computed spatial information. The CGCN model [Yu et al., 2018] uses 1D convolutional layers along with ChebNet [Defferrard et al., 2016]. In the ST-GCN model [Yan et al., 2018], the authors use the simplified spectral graph convolution of equation 3.18 along with a Partition Graph

Convolution (PGC) layer to sample each node's neighbors into groups with different labels based on relevant criteria, and a 1D convolutional layer that computes the temporal data. Besides the already mentioned spatial kernel K, which modulates the spatial reach of the convolution operations, a temporal kernel Γ is introduced whose size represents the window size that samples nodes in the time domain. The ST-GCN model introduces changes in the convolution operation defined in 3.18 with the addition of an *edge importance matrix* $M \in \mathbb{R}^{N \times N}$, a learnable matrix of parameters, to each convolution layer:

$$y = \tilde{D}^{-\frac{1}{2}} (\tilde{A} \circ M) \tilde{D}^{-\frac{1}{2}} X \Theta \tag{3.19}$$

The M matrix is initialized as an all-ones matrix, in order not to interfere with the correlation weights of A. The matrix resulting from the training phase is composed of non-negative edge values weighted according to the relevance of each edge for the given task. The addition of the edge importance matrix provides a simple method for result interpretation, since the resulting weights of each element constitute a direct measure of the relevance of each edge in the learning process.

3.8.3 Graph classification tasks

Approaches to classification tasks on graph data can be divided into two categories: *node-focused* and *graph-focused* [Manessi et al., 2020]. Node-focused approaches consist of a graph $\mathcal{G}$ with fixed structure, composed of a set of labeled and unlabeled nodes. The goal is to learn from the graph's features and topology in order to classify the unlabeled nodes, in a semi-supervised learning approach. Examples of this approach are the classification of documents in citation networks [Kipf and Welling, 2016], the classification of atoms in molecular structures [Scarselli et al., 2009], or the classification of subjects in a population [Parisot et al., 2018a]. On the other hand, graph-focused classification concerns the task of predicting the class of individual graphs, based on their sets of nodes, edges and features, usually in supervised fashion. This approach has been applied for image classification on the MNIST [Defferrard et al., 2016] and ImageNet [Henaff et al., 2015] datasets, where graphs have a fixed 2D grid structure. The selection between node-focused and graph-focused approaches depends on the data's structure, as well as the goals for the given task.

Although both node-focused and graph-focused methods have been applied to fMRI analysis (see Chapter 6), our work performs graph-focused tasks, since our goal is to analyse the intrinsic properties of cerebral network connectivity. Graph-focused approaches, where brain ROIs are represented as nodes and their connectivity as edges, allow GCN models to learn directly from cerebral dynamics, identifying the most relevant ROIs and connections for each classification task.

4. GEOMETRIC DEEP LEARNING FOR NEUROIMAGING ANALYSIS

In this chapter, we detail our approach to applying geometric deep learning for neuroimaging classification. We introduce the two used datasets, following with a description of the classification tasks we have performed. Key to our approaches is how we structure fMRI data as graphs and how we model the neural network architectures to learn from such graphs. The human connectome comprises a series of complex cerebral networks that associate brain structure and functionality. Recent research points to the fact that variability in cerebral connectivity among subjects may be explained by differences in cognitive and behavioral networking patterns [Barch et al., 2013].

Most deep learning applications to neuroimaging data focus on large open access datasets (see Chapter 6), which are essential both for their size, due to the large data requirements of deep learning, and for their open-source nature, facilitating reproducibility and performance benchmarking of different model architectures and analysis techniques. However, the reality of neuroimaging research is that projects usually evaluate a reduced number of subjects, being more susceptible to the effects of artifacts and selection bias [Neuhaus and Popescu, 2018]. This is especially true for task fMRI experiments, which are more complex and expensive to develop and acquire.

The use of novel tools such as GCNs for the investigation of brain connectivity during fMRI experiments, from both task and resting-state scans, could provide neural network models with more integral representations of such cerebral dynamics. We achieve this by constructing fMRI graph structures that encode spatial information of functional connectivity across brain ROIs, as well as temporal BOLD time-series data for each ROI. This procedure, detailed in Section 4.3, takes advantage of the fact that GCN architectures are capable of learning directly from such complex data representations.

In Section 4.4, we introduce two GCN models: ChebNet [Defferrard et al., 2016], and ST-GCN [Yan et al., 2018]. ChebNet is a model which defines spectral graph convolutions in the for of Chebyshev polynomial expansions, making spectral convolutions less costly to compute while retaining their performance. This model constitutes one of the fundamental works on GCNs [Yan et al., 2018], and as such has been widely used particularly among fMRI studies [Ktena et al., 2017, Arslan et al., 2018, Parisot et al., 2018a]. ST-GCN is a GCN architecture that computes both spatial and temporal information encoded into a single graph structure. The integration of temporal information is a key feature for fMRI data analysis, given the strictly temporal character of the functional information represented in the BOLD signal. An ST-GCN model has been recently applied for the first time to resting-state fMRI data on the HCP dataset [Gadgil et al., 2020]. We reproduce this initial study and apply ST-GCNs to a private task fMRI dataset, comparing its results to ChebNet. We assess the ability of GCN models to achieve state-of-the-art performance in small datasets, capitalizing

on graph representations to perform data augmentation techniques that do not synthetically alter or interfere with the data in any form (see Section 4.3).

State-of-the-art models for fMRI classification consist of CNN architectures using both 2D and 3D convolutions, the latter being more computationally and data expensive [Hu et al., 2019]. Previous studies with GCNs on fMRI data achieved state-of-the-art results with relatively simple models [Ktena et al., 2017, Parisot et al., 2018a]. We compare our models to baseline CNN architectures on performance, preprocessing requirements, and result analysis, showing that GNCs provide reasonable advantages over their Euclidean counterparts.

We investigate the assumption that brain networks identified by a GCN as relevant for a classification task may encode higher-level connectivity representations, comprising valuable data for connectome analysis between conditions or tasks. Classification metrics such as accuracy are indicators of whether the networks the classification model highlights as important are indeed significant for analysis from a neuroscientific perspective. However, our main objective is the analysis of the underlying brain organizations that differentiate between classes, which is explored in Chapter 5.

4.1 Datasets and Preprocessing

In this section, we describe the datasets we used for this research regarding the scanning equipment, acquisition parameters, experiment paradigms, and the preprocessing procedures scanned data was subjected to.

4.1.1 ACERTA Dataset

Our work is performed in collaboration with the Brain Institute of Rio Grande do Sul (BraIns) and the Research Group in Multimodal Neuroimaging, which is associated with the institute. The ACERTA dataset — which stands for “evaluation of children at risk of learning disorder” — is part of a BraIns research project which evaluated over 700 children in order to investigate the neural basis of learning disorders [Buchweitz et al., 2019]. Of these children, 100 were diagnosed with dyslexia, of which more than 80 were scanned. These scans were complemented by a similar number of healthy control subjects scans. Dyslexia is a neurobiological disorder that causes learning difficulties in children, estimated to affect 5 to 10 percent of global population [Buchweitz et al., 2019]. Dyslexic children have problems identifying speech sounds and relating them to letters, syllables and words. These issues translate to reading problems, causing the children to read slowly and with low accuracy. To investigate the disorder, dyslexic and healthy control subjects were scanned for structural

and functional MR images. Functional scans consisted of both resting state and task related scans. All healthy controls were rescanned after 1.5 years following the same protocols.

The task experiment was conducted using a mixed event-related reading paradigm validated for Brazilian children. The test stimuli, also called trials, consist of 60 words that appear on-screen for seven seconds each. Words are split into three categories: regular, irregular and pseudowords. In Portuguese, a regular word is a word which is pronounced using the standard phonetic mapping of syllables in a given language. Irregular words show variations between written form and pronunciation, that is, the sound of some syllables must be memorized from experience. Pseudowords are combinations of letters that resemble an actual word, but contain no real meaning. Subjects are presented 20 words of each category, along with a question as to whether that word exists or not. Participants select "Yes" or "No" answers by pressing response buttons placed on both hands, with "Yes" on the left hand and "No" on the right, to match on-screen positions of those words. The answers provided by each subject are stored, and a score value is computed. This score allows control subjects to be classified as good, regular, or bad readers.

The presentation of each stimulus is offset by randomly placed intervals ranging from one to three seconds in duration. After every 10 words, a seven-second rest period ensues, with the on-screen presentation of a centered crosshair (Figure 2.3). Two 30-second baseline rest periods were also inserted in each scan. To ensure equilibrium in T1 magnetization, a six-second dummy scan was presented at the beginning of each 30-word set. An additional 10-second rest period was presented at the end of each scan session. Data was collected on a GE HDxT 3.0 T MRI scanner with an 8-channel head coil. The task and the resting-state EPI sequences used the following parameters: TR = 2000 ms, TE = 30 ms, 29 interleaved slices, slice thickness = 3.5 mm; slice gap = 0.1 mm; matrix size = 64 x 64, FOV = 220 x 220 mm, voxel size = 3.44 x 3.44 x 3.60 mm.

All preprocessing steps were carried out using the AFNI (Analysis of Functional NeuroImage) software [Cox, 1996]. Scans were slice-time and motion corrected, and coregistered with their individual structural T1 scans. Structural images were segmented into gray matter, and cerobrospinal fluid (CSF) and spatially normalized to MNI152 template space. Functional scans are also normalized to MNI152 template and then smoothed using a Gaussian filter. Time points presenting motion of more than 0.3 mm were removed. Nuisance regression was performed using the average time-sequence signal of the white matter and cerebrospinal fluid. All scans are parcellated into ROIs according to the Shen268 atlas [Shen et al., 2013], resulting in 268 distinct ROIs. For task-fMRI, we use Percent Signal Change (PSC) data as input. PSC time-series 2.4 is composed of the percent variation in BOLD signal for each time-point across the whole task scan duration. The mean PSC time-series across the voxels pertaining to each ROI is computed so that each ROI contains a single time-series. After prepocessing procedures, each time-series consists of 640 time-points for each ROI, corresponding to $TR = 1.0s$.

The acquisition of fMRI scans of children have a number of disadvantages when compared to the scanning of adults, the most salient of which is excessive movement. Subjects presenting high average movement rates are removed from our experiments. The resulting dataset consists of 71 examples, 34 DIS and 37 HC. For the reading performance task, we use data obtained for each subject in two distinct scan sessions, acquired 1.5 years apart. This results in a dataset with 54 examples, out of which 20 correspond to good readers and 34 to bad readers.

4.1.2 Human Connectome Project

The Human Connectome Project (HCP) is a consortium led by Washington University, University of Minnesota and Oxford University whose objective is to gather and make freely available MRI data from 1200 healthy young adult subjects (ages 22-35) [Essen et al., 2013]. The field of fMRI research has been recently facing a reproducibility crisis, causing the use of open access datasets to be extensively promoted in order to facilitate replication and consequently reliability [Nickerson, 2018]. The objective of the HCP is to serve as such a database, providing high resolution images on varied MR modalities, such as structural MRI, resting-state fMRI, task fMRI and diffusion MRI. All 1200 subjects were scanned on these four modalities with a 3T Siemens scanner, while 200 of these subjects were also scanned using a 7T scanner. Due to limitations in data storage and processing capabilities, we use resting-state fMRI data from 140 subjects published in releases Q1 and Q2, which allows us to reproduce previous studies related to our work [Finn et al., 2015]. We also reproduce results obtained in a larger sample of all 1200 subjects used by Gadgil et al., although the preprocessing steps applied to their data is not clearly documented [Gadgil et al., 2020]. The resting-state fMRI data acquisition procedures for HCP consisted of two sessions performed in different days, where each session is divided in two scans, where either left-to-right (LR) or right-to-left (RL) phase-encodings are used. The HCP minimal preprocessing pipeline was used [Glasser et al., 2013], which includes motion correction, coregistration and noise removal. The pipeline does not include slice timing correction, since acquisition is made with Fast TR (TR=720ms).

In addition to the HCP minimal processing pipeline [Glasser et al., 2013], we perform further preprocessing steps following the work of [Finn et al., 2015]. These steps consist on the removal of linear components from 12 motion parameters and linear trend, regression of mean time-series of white matter and CSF signals, and band-pass filtering with lower cut-off frequency of 0.01 Hz and higher cut-off frequency of 0.1 Hz. These steps were applied using the AFNI software. Scans are parcellated into 268 distinct ROIs according to the Shen268 atlas [Shen et al., 2013]. This parcellation differs from the parcellation used by Gadgil et al. where 22 macro regions are used.

4.2 Cognitive disorder and Neurodevelopment Classification

We test our approach on different classification tasks using fMRI data, namely three binary classification and one multi-class subject recognition or fingerprinting task. On the ACERTA dataset, we perform classification between dyslexic (DIS) and healthy control subjects (HC), as well as a reading performance classification task between groups of good and bad readers. The goal of these tasks is to identify biomarkers related to dyslexia and learning difficulties among children. Good and bad groups are defined through the scores achieved by each subject during the "word existence" task, which serve as a measure of their text interpretation abilities. Subjects are classified as good, regular or bad readers according to pre-defined score thresholds. We exclude dyslexic subjects from the reading classification task in order to remove likely confounding factors in the evaluation of cognitive development. To benchmark our models performance in the private ACERTA dataset, we perform a sex classification task in the HCP dataset, comparing our findings to the literature.

The subject fingerprinting task is a multi-class classification problem where the goal is to identify brain scans taken from the same subject in different scan sessions as belonging to that subject [Finn et al., 2015]. Previous studies have found that resting-state connectivity is highly individualized, allowing the establishment of links between the cerebral connectome and individual-level characteristics[Jalbrzikowski et al., 2020]. The HCP dataset provides an opportunity for investigating whether deep learning can be used to improve fingerprinting performance, since the dataset has been used successfully on this task by other methods. However, previous studies report poor results when performing fingerprinting with task fMRI data in comparison to resting-state data, especially when using whole-brain scans as input [Kaufmann et al., 2017, Finn et al., 2015]. The ACERTA dataset, with its scan acquisitions made 1.5 years apart for HC children, presents an opportunity to invetisgate fingerprinting biomarkers related to learning processes on school-aged children. Our goal is to first reproduce the results reported by previous studies on the HCP resting-state data before applying our model to the task fMRI data on ACERTA.

4.3 Graph Modeling

We propose a graph-focused approach, where graphs represent the whole brain volume of a single subject with nodes and edge attributes composed of neuroimaging data. We investigate graphs built using task and resting-state fMRI data as node features and edge attributes. Although graphs can be multimodal, using both fMRI modalities, we build graphs of either task data or resting-state data. We work with two distinct graph convolutional networks, the ChebNet spectral GCN model and a spatial-temporal GCN (ST-GCN)

model. For the ChebNet model, a graph is defined as $\mathcal{G} = (\mathcal{N}, \mathcal{E})$, where the set of nodes $\mathcal{N}$ corresponds to the set of 268 regions of interest (ROIs) in the shen-268 atlas, and the set of edges $\mathcal{E}$ represents their connections. The nodes contain a set of attributes represented by a matrix $X \in \mathbb{R}^{n,S}$, where n is the number of nodes and S the size of the window feature vectors extracted from the original task PSC time-series. The set of edges also contains attributes represented by matrix $X^e \in \mathbb{R}^m$, where m is the number of edges. Thus, attribute $x^e_{u,v}$ consists of the Pearson's correlation coefficient computed between nodes u and v that constitute edge e. A weighted adjacency matrix is derived from the average connectivity matrix computed across subjects selected for training. Connectivity matrices are computed using the Nilearn[1], an open-source Python package for neuroimaging analysis providing statistical and machine learning tools [Abraham et al., 2014].

For the spatial-temporal GCN (ST-GCN) model, graphs are defined as $\mathcal{G} = (\mathcal{N}, \mathcal{E}, \mathcal{T})$, where $\mathcal{T}$ is the set of temporal edges. The temporal connections attach each node to itself in the next time-point. In this notation, the attributes of each node can be represented by a node feature matrix $X \in \mathbb{R}^{n,T}$, where T is the number of time-points for a given example graph. We perform simple window slicing data augmentation on each scan, a subsampling method where windows of size S are extracted from the original time-series of size S_0, generating S_0/S examples per subject. The value for S is a hyperparameter defined empirically for each individual classification task and deep learning model combination. We allow no overlay across windows. Due to the window slicing data augmentation, we can define number of time-points $T = S$, which is the window size hyperparameter.

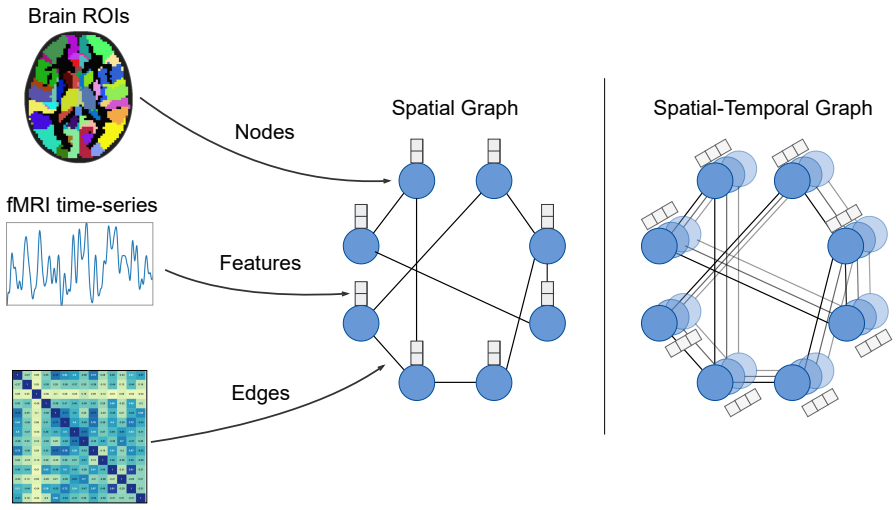


Figure 4.1: Graph modeling procedure.

Figure 4.1 illustrates the spatial and spatial-temporal graph modeling procedures. The spatial graph is used in the ChebNet GCN model, and the spatial-temporal graph is used in the ST-GCN model. Instead of attributing edges to all node pairs, which for a 268-node graph would result in over 70 thousand edges, we attribute edges to pairs of regions that

[1] https://nilearn.github.io/

show a correlation value above a defined adjacency threshold value, which effectively serves as an hyperparameter, resulting in sparse graph representations. This hyperparemeter, henceforth referred to as *adjacency threshold*, acts as a direct mechanism to reduce model complexity. Threshold values are defined for positive and negative correlations, since both are relevant from a neurological perspective. The values for the upheld connections are encoded into the weighted adjacency matrix W.

Although we report only the results for graphs constructed using PSC data, we also experimented with Beta values on early stages of our work, and with multi-modal graphs using combinations of resting-state and task fMRI to compose node and edge attributes. Our early results showed that classifier performance for graphs using Beta values was poorer in comparison to PSC graphs. This is probably attributed to the fact that the Beta series generated by the ACERTA preprocessing pipeline outputs a single Beta value for each voxel per stimulus presentation. For the ACERTA scan paradigm, this results in a set of 60 values per voxel, which in contrast to the 640 time-points for the PSC time-series greatly limits our possibilities of data augmentation and thus the total number of examples. Additionally, Beta values are a statistic metric derived from the time-series that constitute an indirect measure of the time component, hindering the performance of the spatial-temporal model which benefits from larger examples in the time domain. The multi-modal graphs presented no increase in performance, and we believe the result interpretation for such models would be unclear based on the combination of connectivity and activation data.

4.4 Architectures

Based on recent fMRI research using GCNs [Ktena et al., 2017, Parisot et al., 2018a], we use the spectral graph convolution layer with Chebyshev polynomials proposed by [Defferrard et al., 2016] and the ST-GCN model [Gadgil et al., 2020] on the dyslexia and reading performance classification tasks.

4.4.1 ChebNet

For the ChebNet model, illustrated in Figure 4.2, we use a shallow architecture with only two spectral graph convolution (GC) layers, since shallow GCN models have been shown to consistently outperform deeper models, with best results achieved between 2-3 layers [Magner et al., 2019]. The input to the network are feature matrices $X \in \mathbb{R}^{N \times S}$, where S is the size of the windows extracted from the PSC time-series at each of N ROIs. We use windows of size 7 for the PSC time-series, which are mapped to each of the stimuli and rest

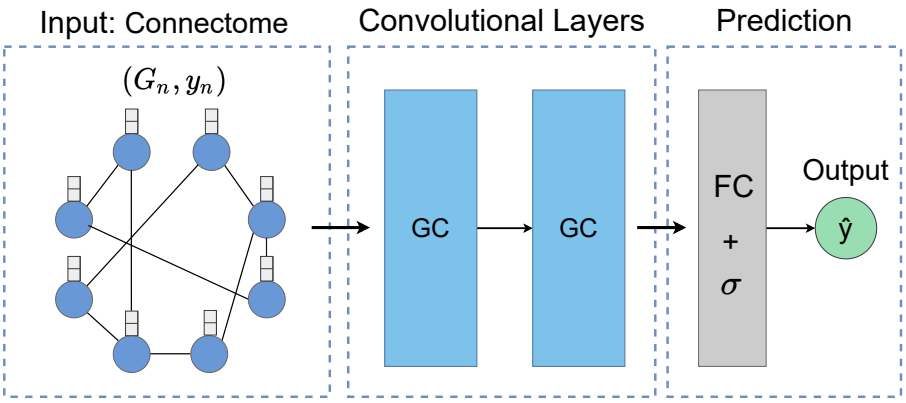


Figure 4.2: ChebNet model.

Layer	Dimension
GC	(1,16)
ReLU	
GC	(16,1)
ReLU	
Fully-connected	(268,100)
Fully-connected	(100,1)

Table 4.1: ChebNet architecture.

periods presented during the fMRI scan. Windows are positioned 3 seconds after stimulus presentation, due to the expected delay in BOLD signal response.

The model is trained using kernel size $K = 3$ and Chebyshev convolutions of shape (F_{in}, F_{out}), where F_{in} is the number of input channels, which corresponds to the window size S in the first layer, and $F_{out} = 16$ is the number of output channels. A batch normalization layer and RELU activation follow the first layer. The output of these operations is a graph in the same format as the input graph and containing a single value updated from the convolution operations. The resulting graph is flattened and connected to a fully-connected (FC) network for classification. The output is activated with a sigmoid activation function and used to compute the binary cross entropy loss.

4.4.2 ST-GCN

The ST-GCN model, illustrated in Figure 4.3, is built with 4 spatial-temporal (ST-GC) layers. Similarly to ChebNet, the network inputs are 1-channel feature matrices $X \in \mathbb{R}^{N \times T}$, where T is the size of the windows extracted from the PSC time-series at each of N ROIs. To capitalize on the temporal aspect of the model, we use larger windows of 300 time-points per dataset example. We use the edge-importance ST-GCN model [Gadgil et al., 2020], which

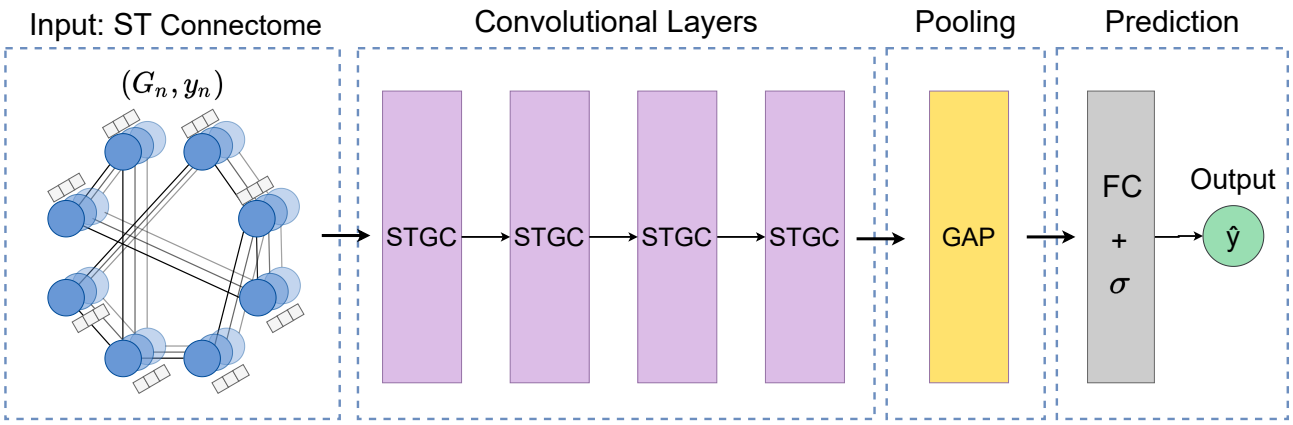


Figure 4.3: ST-GCN model.

Layer	Dimension
ST-GC	(1,64)
BatchNorm + ReLU	
ST-GC	(64,64)
BatchNorm + ReLU	
ST-GC	(64,64)
BatchNorm + ReLU	
ST-GC	(64,64)
BatchNorm + ReLU	
Global Avg Pooling	(268,300)
Fully-connected	(64,1)

Table 4.2: ST-GCN architecture.

applies the simplified chebyshev convolution of equation 3.18 on the spatial information, along with 2D convolutions on the time-series of each node.

The model architecture is described in Table 4.2. Spatial convolutions of 64 layers are applied with spatial kernels of size $K = 1$. Following each spatial GC layer, temporal 2D convolutions are applied with temporal kernel size $(\Gamma, 1)$ where $\Gamma = 11$. Each spatial and temporal layer is followed by 2D batch normalization and RELU activation. We apply global average pooling and use a fully-connected (FC) network followed by sigmoid activation function for classification.

4.4.3 Siamese ST-GCN

For the subject fingerprinting task, we generate a siamese ST-GCN model composed of a pair of 4 to 7-layer ST-GCNs. Our choice for the use of a siamese model is motivated by the success achieved by studies employing siamese architectures for face recognition tasks [Chopra et al., 2005]. Siamese models learn by discriminating pairs of examples

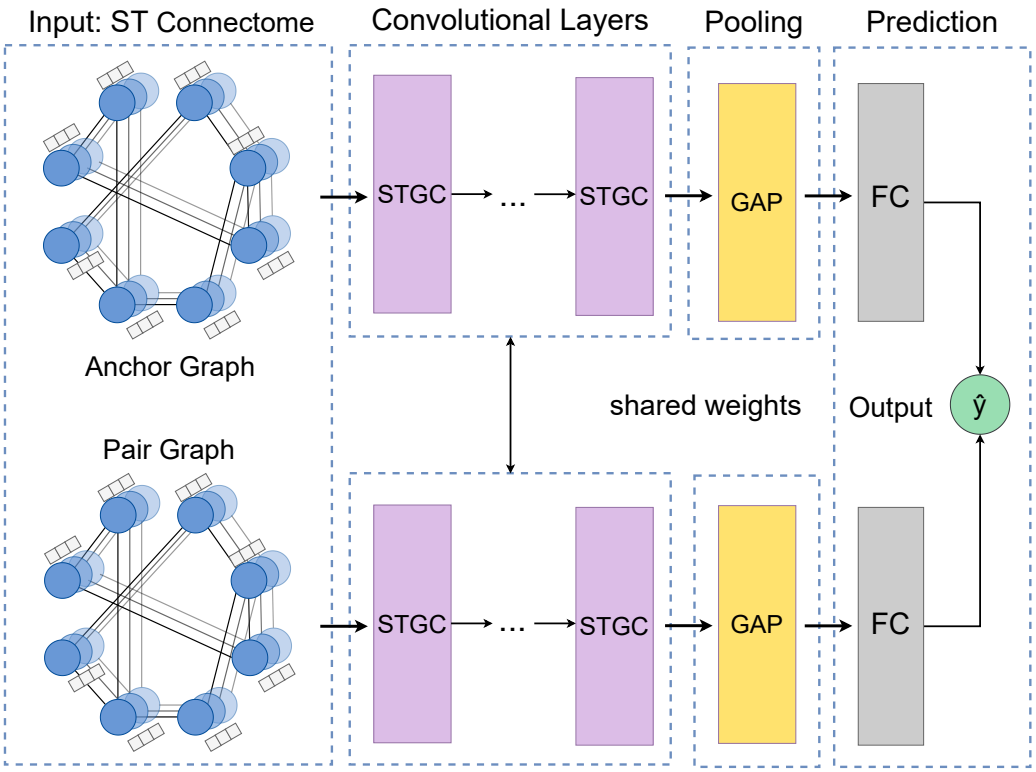


Figure 4.4: Siamese ST-GCN model.

as to whether they belong or not to the same class. As such, the goal of a siamese model is to learn low-dimensional data embeddings capable of representing the most relevant features for distinguishing between subjects across the entire dataset. Since the objective of the fingerprinting task is to identify subjects across two fMRI sessions acquired with a spacing of 1.5 years, we force the network to classify pairs of scans from different sessions during training, in an attempt to guide the network to learn the features most related to the neurological changes expected for children in school age. Although the fingerprinting of subjects is a multi-class classification task, our model performs binary classification between pairs, which can later be fitted to the multi-class prediction by a voting procedure.

The architecture for each branch of the siamese pair is the same previously defined for the single ST-GCN model in Table 4.2. The model inputs are an *anchor* example and a corresponding *pair*, each passing through a branch of the network. The pairs are either examples of the same subject (positive) or not (negative). We test alternatives of loss function: contrastive loss (see Section 3.7.2) and an adapted NT-Xent [Chen et al., 2020]. For contrastive loss, The L2 norm between the resulting anchor and pair vectors is used to compute the loss, for which we opt for a margin value $C = 0.5$. The NT-Xent loss is computed taking a mini-batch of N_B examples as input, and computing the cosine similarity $sim(u, v) = u^\top v / \|u\| \|v\|$ between pairs of examples (u, v). The loss or error $E_{i,j}$ for a positive pair (i, j) is defined as:

$$E_{i,j} = -\log \frac{exp(sim(x_i, x_j)/\tau)}{\sum_{k=1}^{N_B} \Upsilon_{(k,i)} exp(sim(x_i, x_k)/\tau)}, \tag{4.1}$$

where $\Upsilon_{(k,i)}$ is a function that evaluates to 1 if k and i form a negative pair and 0 otherwise, and τ is a temperature parameter which scales the range of the outputs. We use $\tau = 1.0$. We introduce the Υ function in NT-Xent in since our mini-batches contains a randomized proportion of positive and negative examples, so that each pair must be checked individually. Details on the pair formation procedures are presented in Section 5.5.

4.4.4 Baseline CNN

We use a 2D Convolutional Neural Network (CNN) as baseline to both GCN models. Although 3D CNNs have shown better performance than 2D models for fMRI classification [Hu et al., 2019], 3D models are more expensive to train and easier to overfit due to their larger number of parameters, a concern for small datasets like ACERTA. We select a VGG-based architecture as it constitutes a powerful model for image recognition tasks [Simonyan and Zisserman, 2015]. VGG architectures are composed of groups of CNN layers followed by RELU activation, and max pooling layers separating each CNN group. We also include batch normalization layers following each convolution. The final max pooling layer is followed by fully-connected layers used for classification.

Layer	Dimension
Conv 2D	(61,64)
BatchNorm + ReLU	
Conv 2D	(64,64)
BatchNorm + ReLU	
Max Pooling	(4)
Conv 2D	(64,32)
BatchNorm + ReLU	
Conv 2D	(32,32)
BatchNorm + ReLU	
Max Pooling	(8)
Fully-connected	(64,1)

Table 4.3: VGG architecture.

We use VGG models with two sizes in different tasks, with 4 or 7 CNN layers in order to control overfitting. Since VGG 2D convolutional layers can not be applied to the multidimensional data used in the GCN models, we use the F-statistics extracted from the fMRI time-series as input. F-statistics are a measure of signal change relative to baseline level in a fMRI scan attributed to each voxel, which can be represented as a 3D image. In order to apply 2D convolutions to such images, we generate multi-channel examples

where the z-axis is represented as channels, so that inputs have shape (X, Y, C). We use 64-channel outputs in the first group CNNs, and 32-channel outputs for the second group CNNs.

5. EXPERIMENTS AND RESULTS

In this chapter, we describe the experiments we conducted on fMRI data for the ACERTA and Human Connectome Project datasets, detail training and evaluation procedures and discuss our results and their implications. We detail the hyperparameters used for each model on each classification task, which are selected via grid search, with the exception of the adjacency threshold. The searched hyperparameters are chosen based on literature recommendations. We carried out all experiments in a quad-core Intel Core i7-8565 CPU @ 1.80GHz, 8 GB of RAM, and 2GB NVIDIA GeForce MX110 graphics card running Ubuntu Linux 18.04. We implement our models in the Pytorch framework using the Pytorch Geometric library's graph convolutional layer implementations [Fey and Lenssen, 2019].

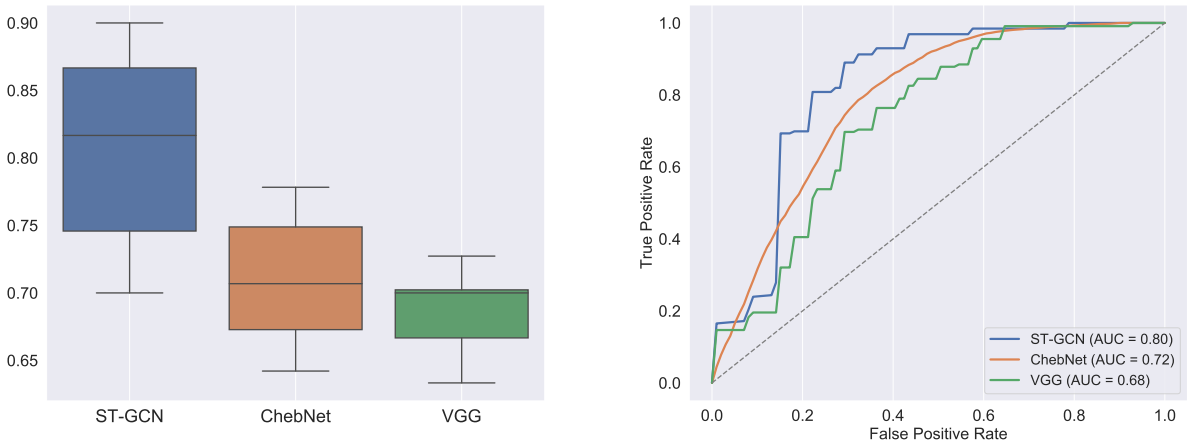
5.1 Dyslexia Classification

We perform the dyslexia classification task using ST-GCN, ChebNet, and VGG models. The task consists of binary classification between groups of dyslexic and healthy control subjects. We split the dataset into training and test sets using holdout with 70% of examples in the training set. For the GCN models, the connectivity matrices for each subject are averaged across all training set subjects and used as the graph edge attributes, so that each graph in the dataset has the same topology. The graph topology used for examples of the training set is also used for the examples of the test set. Training is performed using the hyperparameters of Table 5.1, which also presents the total number of examples for each model after data augmentation is applied to the models that support it, ST-GCN and ChebNet. We use binary cross entropy loss, Adaptive Moment Estimation (ADAM) as optimizer and plateau learning rate scheduling.

Hyperparemeter	**ST-GCN**	**ChebNet**	**VGG**
Learning rate	1×10^{-3}	1×10^{-4}	1×10^{-4}
Weight decay	1×10^{-3}	5×10^{-3}	5×10^{-2}
Dropout	5×10^{-1}	5×10^{-1}	5×10^{-1}
Adj threshold	5×10^{-1}	5×10^{-1}	-
Window size	300	7	-
Nº of examples	142	4260	71

Table 5.1: Training hyperparameters for dyslexia classification.

Figure 5.1 shows the results achieved for the three models for a total of 10 executions of 100 epochs each. The ST-GCN shows the best performance in discriminating between classes, achieving a mean accuracy of 82% and mean AUC score of 0.80. ChebNet presented mean accuracy of 71% and mean AUC score of 0.72, while VGG presented



(a) Accuracy.

(b) ROC curves and AUC scores.

Figure 5.1: Dyslexia classification results for each model.

mean accuracy of 70% and mean AUC of 0.68. Both GCN approaches have the advantage of having more available data due to the time-series data augmentation. ChebNet has the most examples available for training, resulting in a smoother ROC curve. The ST-GCN model performed best when working with examples with windows of size $S = 300$, resulting in twice as much examples as the VGG model, but 30 times less examples than ChebNet. Its improved performance can thus be mostly attributed to its capacity of processing temporal data within the temporal convolutional layers, given it benefits from larger time-series instead of larger number of examples.

5.1.1 Effects of adjacency thresholding

The results for ST-GCN and ChebNet models presented in Figure 5.1 correspond to experiments using adjacency threshold of 0.5. We select this value after empirical testing. Figure 5.2 shows classification accuracy for the ST-GCN using distinct adjacency threshold values. The higher the threshold, the fewer the number of selected edges, and the higher the correlation represented by each selected edge.

Higher threshold values, that is, input graphs using only the most correlated edges across the whole brain, yield better performance. The total number of selected edges is approximately 70 thousand edges when no threshold is used, 5 thousand for threshold of 0.5 and 700 for threshold of 0.7. This values vary slightly per execution, since the mean connectivity is computed from the examples in the training set. We present our results for dyslexia classification and the remaining experiments using the adjacency threshold of 0.5 instead of 0.7, given that the higher number of available edges provide more flexibility in

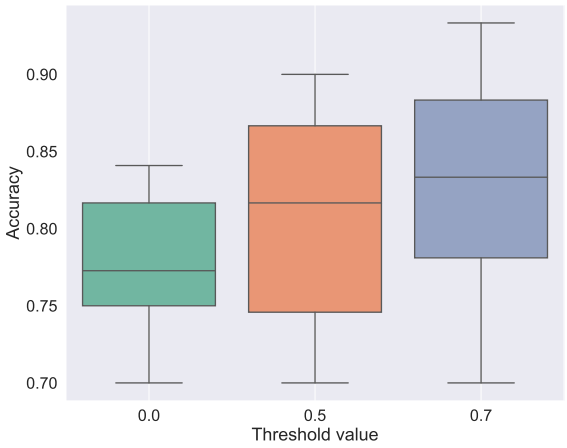


Figure 5.2: Effect of different adjacency thresholds on accuracy.

the learning process, allowing the investigation of more edges across the brain volume and enriching result interpretability.

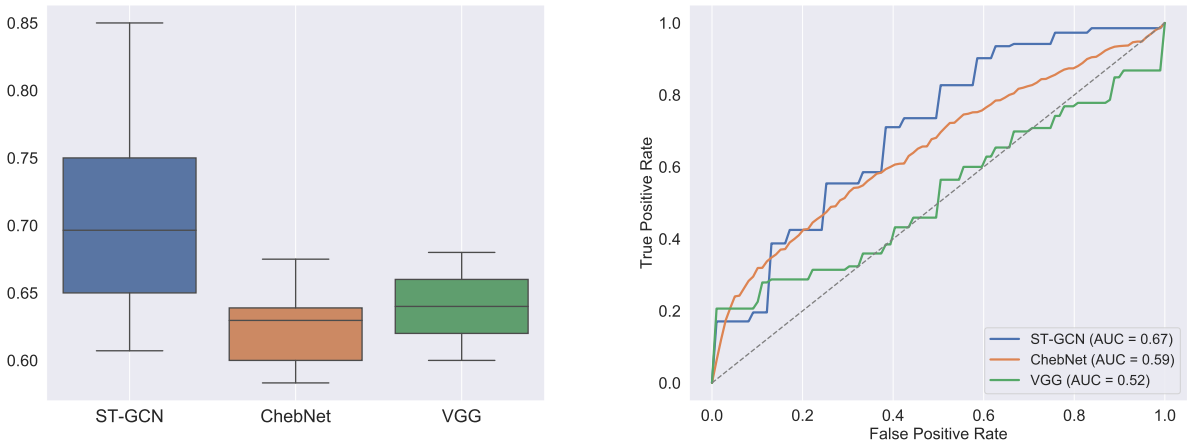
5.2 Reading Performance Classification

We repeat the procedures used in the dyslexia classification for the task of reading performance classification. This task consists of binary classification between groups of Good and Bad readers. We train ST-GCN, ChebNet and VGG models in the dataset consisting of only healthy-control subjects, using holdout to select 70% of examples for training and the remaining 30% for testing. Training is performed with the hyperparameters detailed in Table 5.2, binary cross entropy loss, and optimized using Adaptive Moment Estimation (ADAM) as optimizer and plateau learning rate scheduling.

Hyperparemeter	**ST-GCN**	**ChebNet**	**VGG**
Learning rate	1×10^{-3}	5×10^{-5}	1×10^{-4}
Weight decay	1×10^{-2}	1×10^{-2}	5×10^{-2}
Dropout	5×10^{-1}	5×10^{-1}	5×10^{-1}
Adj threshold	5×10^{-1}	5×10^{-1}	-
Window size	300	7	-
Nº of examples	108	3240	54

Table 5.2: Training hyperparameters for reading performance classification.

Figure 5.3 shows the results obtained after 10 executions of each model. The ST-GCN model provides the best results, with a mean accuracy of 70% and mean AUC score of 0.67. ChebNet presents a mean accuracy of 63% and mean AUC score of 0.59. We note that although accuracy for the VGG model was on par with the other models, with mean value of 64%, its AUC score is close to a random classifier. This discrepancy may be explained



(a) Accuracy.

(b) ROC curves and AUC scores.

Figure 5.3: Reading performance classification results for each model.

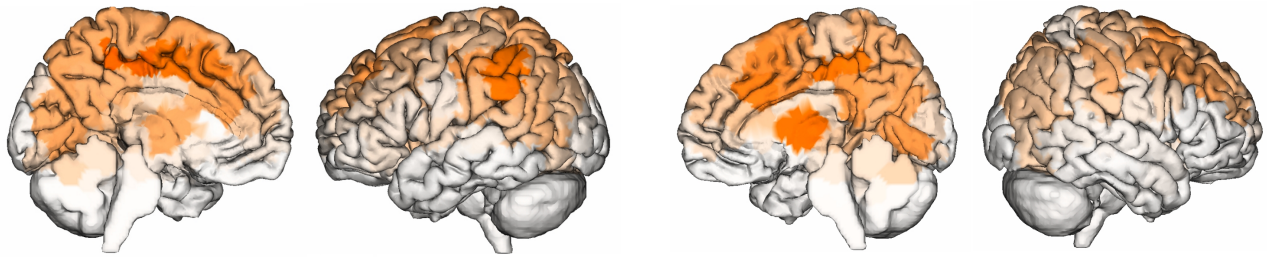
by the smaller number of available examples for this model, since no data augmentation is applied. The small number of samples means slight model biases in the prediction of either class could inflate accuracy numbers, since a single example can have a noticeable impact in results.

Results are beneath those obtained for the dyslexia classification task, which could be expected since sample size is slightly smaller and the dataset for this task consists of healthy controls only, forming a more homogeneous distribution across the groups to be classified. However, the performance achieved by the ST-GCN indicates that the model was able to identify distinctions between the data distributions of both groups, even if its ROC curve shows that predictions are not optimal.

5.3 ACERTA Biomarker Analysis

To investigate the most relevant features for the classifications task, we analyze the edge importance matrices $M \in \mathbb{R}^{N \times N}$ extracted from the ST-GCN models. The edge importance matrices are matrices of parameters which are optimized during training. During optimization, weights are attributed to each edge composing the input graph's adjacency matrix according to their relevance for classification. As such, edge importance weights are not a direct representation of the brain functionality of the analyzed subjects, but a representation of the edges and, by extension, the brain ROIs identified as relevant by the classifier. The analysis of this representation allows for an indirect indication of the cerebral networks involved in the conditions of interest, in this case dyslexia and reading ability. Edge importance weights are non-negative values attributed to edges representing both positive and negative correlations. During the learning process, the edges deemed as least relevant are

attributed an equal low value baseline weight. We analyze all edges weighted above this baseline.



(a) Left side brain. (b) Right side brain.

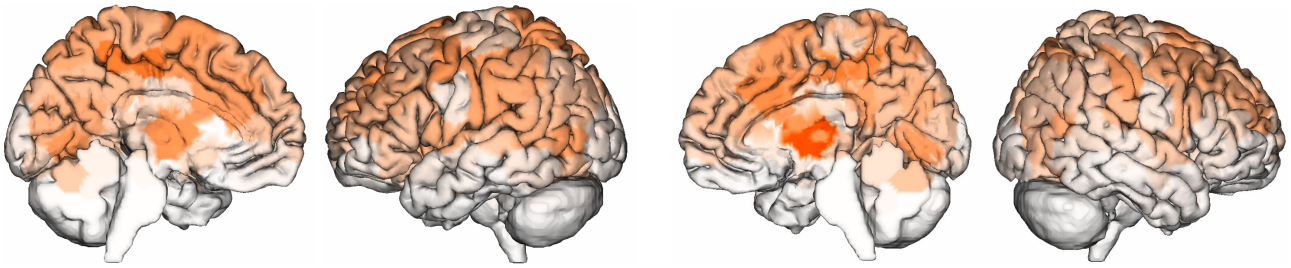
Figure 5.4: Sagittal view of edge distribution for the dyslexia classification. Darker areas represent higher number of connections. Images generated using the BioImage Suite web application[1].

A representation of the edge distribution across brain regions is shown in Figure 5.4. Darker shades of orange represent regions with nodes of higher degree (higher number of connections). The nodes of highest degree and (D=98) thus the most relevant in distinguishing dyslexic and controls are found in the left limbic (L-limbic) lobe, in the dorsal posterior cingulate cortex (PCC), Broadmann area (BA=31), and premotor and supplementary motor cortex (BA=6). The L-limbic and R-limbic also show high degress in the right PCC (D=91), ventricular posterior and anterior cingulate cortices (D=74/BA=24,D=67/BA=23), and the dorsal anterior cingular cortex (ACC) (D=64/BA=32). The R-subcortical lobe shows high connectivity in the Thalamus (D=92), and the L-parietal lobe in the angular gyrus (D=89/BA=39) and the visual motor cortex (D=70/BA=7). The R-prefrontal and L-prefrontal lobes are highlighted in the dorsolateral prefrontal cortex (dlPFC) (D=78,81/BA=9) and frontal eye field (D=81/BA=9). The R-occipital lobe presents high connectivity in the primary (D=62/BA=17) and secondary (D=68/BA=18) visual cortices.

Most of the regions detected by our method are listed in the literature as playing a role in dyslexia and general language processes. The PCC has been linked to dyslexia in previous studies [Stoitsis et al., 2008, Buchweitz et al., 2019] in its relation to pre-attentive processes. The thalamus has been associated with high-level cognitive functions such as attention and working-memory, with studies pointing to its importance in mnemonic attention [de Bourbon-Teles et al., 2014] and language-related abilities [Radanovic et al., 2003]. The dorsal ACC has shown more signal activation in healthy controls relative to dyslexics [Buchweitz et al., 2019], while the angular gyrus has exhibited deactivation in dyslexic men [Pugh et al., 2000].

The edge distribution for the reading performance classification comprehends most of the same nodes described for the dyslexia classification, although with different attribution of degrees. As seen in Figure 5.5, there is a high number of connections located in R-

[1] Available at: https://bioimagesuiteweb.github.io/webapp (last accessed: March 2020)



(a) Left side brain. (b) Right side brain.

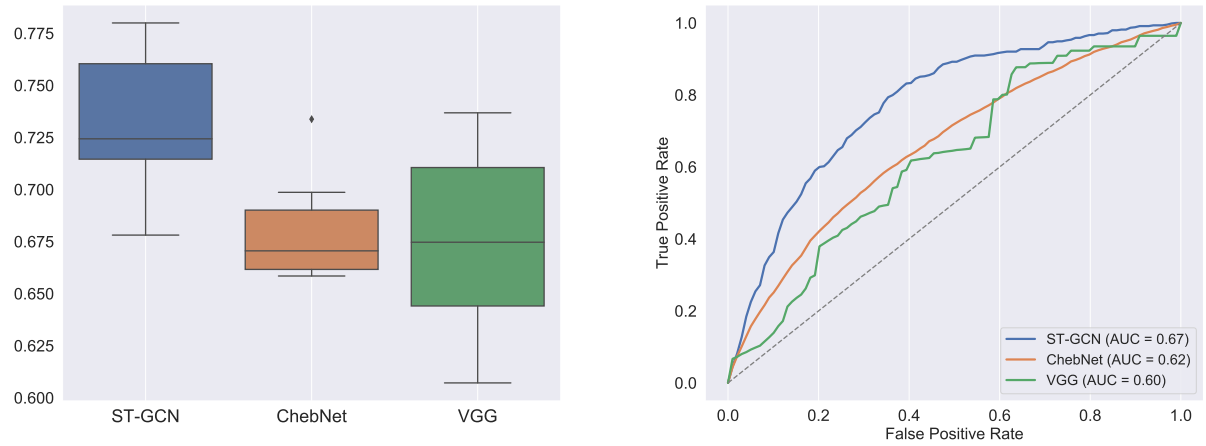
Figure 5.5: Sagittal view of edge distribution for the reading performance classification. Darker areas represent higher number of connections.

thalamus (D=123) and L-thalamus (D=104). The L-dorsal PCC (D=125/BA=31) and R-dorsal PCC (D=108/BA=31) show similarly high degrees. We observe high connectivity in the prefrontal lobe, such as the R-frontal eye fields (D=113/BA=8) and the L-dlPFC (D=101/BA=9). The secondary visual cortices on the L-occipital lobe (D=99/BA=18) and R-occipital lobe (D=98/BA=18), and the L-insula (D=94/BA=13) are also noticed. Among these, the insula appears only in the reading performance task. Its functionality has been linked to salience processing [Uddin, 2014], decision making [Ibrahim et al., 2019] and speech [Uddin et al., 2017].

5.4 Sex Classification

We perform a subject classification task on resting-state images from two releases of the HCP dataset, Q1 and Q2, resulting in 140 subjects. The task consists of binary classification of subjects between male and female groups. We maintain the hyperparameter values used for ST-GCN in the ACERTA tasks, so that window size used for data augmentation is $S = 300$ and the adjacency threshold of 0.5 for the GCN models. Since we analyze resting-state for this task, and the number of samples is larger, we increase the window-size for ChebNet to $S = 100$. The augmentation procedure generates a total of 560 examples for the ST-GCN, and 1680 for ChebNet. Given the larger size of this dataset in comparison to ACERTA, we increase the number of layers for ST-GCN from 4 to 8, and for ChebNet from 2 to 7.

Since we use resting-state data for this task, we are unable to exactly reproduce the CNN approach used for the ACERTA dataset, which uses F-statistic data as input. Different approaches are described in the literature for the application of CNNs to resting-state data, such as computing mean and standard deviation over sliding windows across 4D fMRI data to generate 2-channel 3D examples used as input for 3D CNNs [Li et al., 2018], or transforming 4D images into three multi-channel 2D images used as input to a 2D CNN ensemble [Hu et al., 2019]. In order to use the same baseline CNN model as in the ACERTA task, we



(a) Accuracy.

(b) ROC curves and AUC scores.

Figure 5.6: Sex classification task results.

generate multi-channel 2D images from the mean time-series of each scan. We use holdout with 70% of examples in the training set to split the dataset for the learning phase. We train the model using binary cross entropy loss and Adaptive Moment Estimation (ADAM).

The obtained results are shown in Figure 5.6. Best results are reached by the ST-GCN model with mean accuracy of 72.5% and mean AUC of 0.67. ChebNet achieves mean accuracy of 67% and mean AUC score of 0.60, and VGG presents mean accuracy of 68% and mean AUC of 0.62. The results obtained by the ST-GCN model are consistent with the state-of-the-art for this task when using the same number of subjects [Hu et al., 2019]. However, we are unable to reproduce the results reported for ST-GCN on the Gadgil et al. dataset, consisting of 1200 subjects. We achieve mean accuracy of 76%, which is below the reported accuracy of 83.6% and the state-of-the-art for a similar number of subject samples.

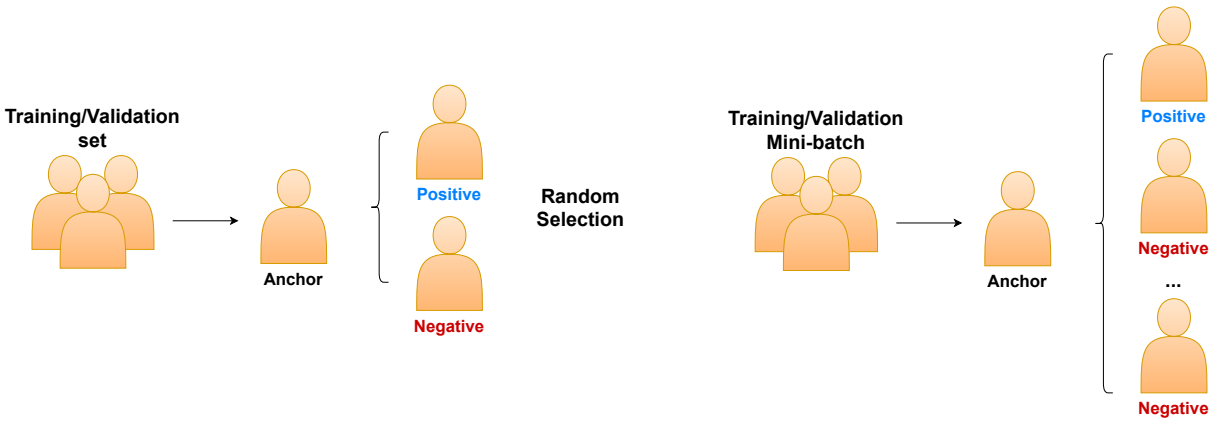
Hyperparemeter	**ST-GCN**	**ChebNet**	**VGG**
Learning rate	1×10^{-3}	1×10^{-3}	1×10^{-4}
Weight decay	1×10^{-3}	1×10^{-2}	1×10^{-2}
Dropout	5×10^{-1}	5×10^{-1}	5×10^{-1}
Adj threshold	5×10^{-1}	5×10^{-1}	-
Window size	300	100	-
Nº of examples	560	1680	140

Table 5.3: Training hyperparameters for sex classification.

5.5 Subject Fingerprinting

We perform the fingerprinting task on both the ACERTA and HCP datasets, with only healthy controls used for ACERTA. For this task, we use a Siamese Network that receives a pair of examples and attempts do predict whether they belong to the same subject. To this purpose, the dataset is split so that for each subject both visit 1 (V_1) and visit 2 (V_2), which correspond to fMRI scan acquisitions made 1.5 years apart, are kept in the same set (training or test). We form pairs of input data by attributing an unique id value for each subject and randomly selecting a pair of subjects for each example in the training and validation sets. The original examples of each split are called *anchors*, and their corresponding example *pairs*. Labels with value 1 are attributed to positive examples, where anchor and pair have the same id (same class), and labels with value 0 to negative examples, where anchor and pair have different ids. For each anchor example, we randomly select one positive and one negative example pair from the dataset. We employ a learning rate scheduler that updates on training loss plateaus with patience of 10 epochs. We split subjects in training and test sets using holdout with 70% of examples in the training set. For training and validation, we perform binary classification in order to better assess the model's predictive power before performing actual fingerprinting across all subjects in the dataset (see Section 4.4.3).

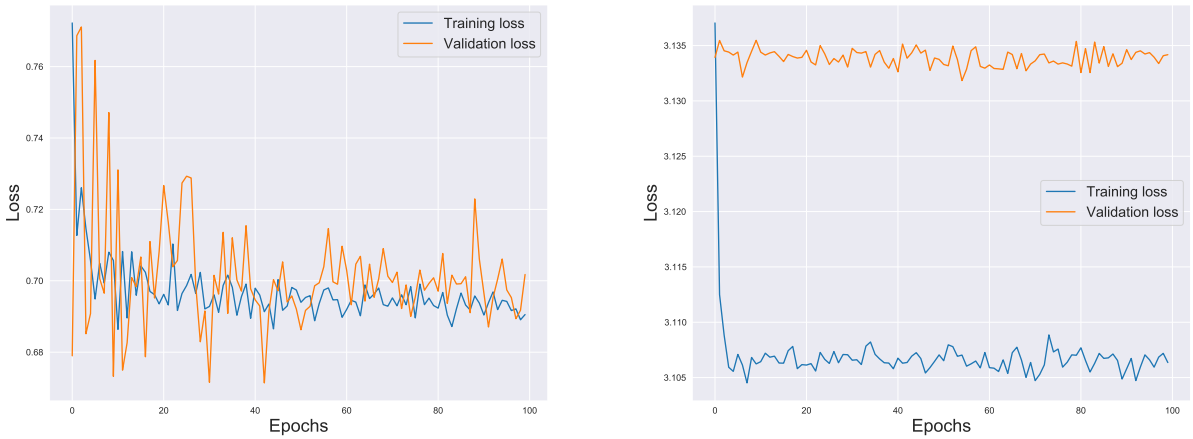
Our results for this task are unsatisfactory, with accuracy remaining within the chance range for both the contrastive loss and NT-Xent loss models. Figure 5.8 shows the training and validation losses computed across 5 executions for 100 epochs for the HCP dataset. Neither of the loss functions employed are able to correctly inform the model in the learning process, and we see no improvement in accuracy across epochs for both HCP and ACERTA datasets, with models showing signs of underfitting. We increase the number



(a) Pair formation and computing of Contrastive Loss.

(b) Computing NT-Xent for each mini-batch example.

Figure 5.7: Pair formation and loss computation procedures for each function. Each mini-batch example in (b) is a pair formed as shown in (a).



(a) Contrastive loss.

(b) NT-Xent loss.

Figure 5.8: Loss curves for the HCP dataset fingerprinting.

of layers and channels for the graph convolutional layers, but these changes are unable to improve model performance. We believe these results can be partly attributed to the fixed graph topology required for our models to operate, where the edge attributes are the ROI connectivity values averaged across the subjects in the training set. Although our model finds no pattern to identify individuals, the literature show that simple correlation between resting-state connectivity matrices and a voting procedure for predictions can achieve an accuracy of up to 99%. Our models seem unable to learn these individual connectivity patterns from the raw time-series alone, without the individual connectivity data.

6. RELATED WORK

We have conducted a systematic literature review on state-of-the-art applications of GCNs on neuroimaging data, following the framework proposed in [Kitchenham and Charters, 2007]. In this chapter, we present an overview of our review. The goal of the review is to assess the performance of GCNs when compared to the state-of-the-art, the various approaches to graph modeling, and the most used graph convolution implementations. We pinpoint the most relevant reviewed studies concerning these aspects and relate them to the present work. Our search was conducted using the search strings detailed in Table 6.1, returning a total of 146 studies. A large number used either GNNs without the inclusion of convolutions, or CNNs. In addition, many results were related to MRIs on organs other than the brain or didn't make use of medical imaging data. Another set of studies were published by the same research group and described identical approaches. Papers presenting such characteristics were excluded from our review, which assessed 26 relevant studies.

String	'**P**' AND '**Q**' AND '**R**' NOT '**S**'
P	'Graph Convolutional Networks' OR 'Graph Neural Networks' OR 'GCN' OR 'GCNN' OR 'Geometric Deep Learning'
Q	'fMRI' OR 'MRI' OR 'DTI' OR 'DWI'
R	'Brain' OR 'Neuroimage' OR 'Neuroimaging'
S	'EEG' AND 'MEG'

Table 6.1: Detailed search strings used on the systematic review.

A major trend among the reviewed papers is the use of spectral-based GCNs, with only a single paper employing spatial-based models [Kawahara et al., 2017]. The spectral-based studies used either the ChebNet model [Defferrard et al., 2016] or its first-order approximation, usually referred to as GCN [Kipf and Welling, 2016]. The choice for spectral-based convolutions might be attributed to its straightforward implementation, whereas spatial-based methods offer a greater variety of options for implementing message passing mechanisms [Wu et al., 2020]. We follow these early works in our choice for a spectral-based approach due to its dissemination in fMRI studies, robust implementation and good performance on classification tasks.

Graph Type	Edges	Nodes	Citation
Connectivity	s-MRI	s-MRI	[Liu et al., 2020]
	rs-fMRI	rs-fMRI	[Ktena et al., 2017], [Arslan et al., 2018], [Kim and Ye, 2020], [Zhang and Huang, 2019] [Gadgil et al., 2020]
	t-fMRI	t-fMRI	[Qu et al., 2021]
	DWI	DWI	[Lee et al., 2019], [Hong et al., 2019], [Kawahara et al., 2017]
	DWI	rs-fMRI	[Yao et al., 2021], [Li et al., 2020]
	s-MRI	DWI	[Zhang et al., 2019c], [Liu et al., 2017]
	s-MRI	rs-fMRI	[Ktena et al., 2017], [Ma et al., 2018], [Zhang et al., 2019b]
Population	Phenotypic	rs-fMRI	[Parisot et al., 2018b], [Kazi et al., 2019], [Anirudh and Thiagarajan, 2017], [Valenchon and Coates, 2019], [Huang and Chung, 2020], [Jun et al., 2020]
	Morphological	s-MRI	[Liu et al., 2019]

Table 6.2: Summary of graph modeling approaches identified in the literature. Abbreviations refer to structural MRI (s-MRI), Diffusion Weighted Imaging (DWI), resting-state fMRI(rs-fMRI) and task fMRI (t-fMRI).

Most studies investigate GCN applications to large open-source datasets, which are commonly used to benchmark state-of-the-art methods. The ABIDE[1] dataset, which investigates autism spectrum disorder, is used in 6 studies. ADNI[2], a dataset containing scans from subjects afflicted by Alzheimer's disease, is used in 5 studies, and TADPOLE[3], a subset of the ADNI dataset, is used in 2 studies. The UK Biobank[4] (healthy adults) and PPMI[5] (Parkinson's disease) datasets are used in 2 studies each. Three studies report the use of the HCP dataset. One study uses data from the PNC[6] dataset, which is composed of fMRI scans of children. Use of private datasets are reported in 5 studies, with subject sample sizes ranging from 42 to 167 subjects. The present work differs from the target studies in our review in that we apply the same techniques to both private and open-source datasets. We choose HCP as our open-source dataset for its large number of high-resolution multi-modal images and its previous use for the fingerprinting task [Finn et al., 2015], allowing for direct comparison to our methods.

Classification tasks using GCNs are divided into node-focused and graph-focused (see Section 3.8.3). Table 6.2 summarizes the graph modelling approaches identified by our literature review. Connectivity graphs refer to graphs that represent the human connectome,

[1] http://preprocessed-connectomes-project.org/abide/ (last accessed: March 2020)
[2] http://adni.loni.usc.edu (last accessed: March 2020)
[3] https://tadpole.grand-challenge.org/ (last accessed: March 2020)
[4] https://www.ukbiobank.ac.uk/ (last accessed: March 2020)
[5] https://www.ppmi-info.org/ (last accessed: March 2020)
[6] https://www.med.upenn.edu/bbl/philadelphianeurodevelopmentalcohort.html (last accessed: March 2020)

with nodes representing brain ROIs and edges encoding their connectivity. The classification of connectivity graphs constitute graph-focused tasks, where the goal is to predict the label for whole graphs. Population graphs are assembled with subjects represented as nodes, and edges connecting subjects according to their common characteristics. Most studies use phenotypic variables (sex, age, genome) for that end, while one study reports use of brain ROI morphology features such as such as grey matter volume, cortical thickness and surface area. The classification of population graphs constitute node-focused tasks and are usually performed in semi-supervised manner, with the goal of predicting the label of individual nodes.

Both approaches are capable of achieving state-of-the-art performance for prediction tasks, as reported by studies on the same dataset [Ktena et al., 2017, Parisot et al., 2018b]. We opt for a graph-focused connectivity approach given our objective of generating models that learn directly from cerebral network dynamics. Works comparing graphs constructed using different data modalities, such as resting-state fMRI only and graphs combining structural MRI as edge attributes and resting-state fMRI as node attributes [Ktena et al., 2017] report no significant differences in predictive performance between both configurations. As mentioned in Section 4.3, although we test different graph configurations, we focus our efforts on graphs with a single modality due to their clearer interpretability and simplicity of implementation.

The use of Spatio-Temporal GCNs for fMRI is first proposed for a graph-focused sex classification task on the HCP and NCANDA [7] datasets [Gadgil et al., 2020]. The authors use an architecture based on the original ST-GCN work [Yan et al., 2018] with a minor alteration on the edge importance mechanism. Instead of computing edge importance for each convolutional layers, a single edge importance matrix is attached to the adjacency matrix and its weights are updated during the training phase. This model outperforms the state-of-the-art with an accuracy of 83.6%. The visualizations generated from the edge importance matrix highlight cerebral networks pertaining mostly to the visual cortex. The ability to include temporal information in the graph convolution operation is a key feature in the analysis of BOLD time-series. We reproduce the author's approach on the HCP dataset and apply an ST-GCN model to the ACERTA dataset, motivated by the reported model performance and the result visualization potential it entails.

Siamese GCN models have been used for metric learning of binary classes on the ABIDE dataset, achieving state-of-the-art performance for classification between autistic subjects and healthy controls [Ktena et al., 2017] . The model uses binary positive and negative pair labels during training, and the authors propose a *constrained variance loss function*. This loss function operates similarly to contrastive loss (see Section 3.7.2, but instead of acting on the Euclidean distances, it aims to maximize the mean similarity between same-class examples and minimize it for examples of different classes, while constraining

[7] http://ncanda.org/data-analysis-core.php

the variance for each class within a given threshold. The good results achieved by this approach on fMRI data combined by the successful uses of siamese models to face recognition tasks [Chopra et al., 2005] serve as motivation for the application of a siamese model to the subject fingerprinting task (see Section 4.2).

Subject fingerprinting is performed with remarkable results on the HCP dataset using data from two resting-state sessions and four task sessions acquired in two following days [Finn et al., 2015]. The authors compare subject's connectivity matrices following the assumption that matrices from the same subject will show increased Pearson's correlation values. Connectivity matrices are computed using both whole-brain and selected networks data, and generating one matrix for each scan session, resulting in six examples per subject. Each example is compared to every other example in the dataset, and the example pair achieving highest Pearson's correlation value is selected as the model's prediction. The reported accuracy for whole-brain data ranges from 92.9%-94.4% for resting state data and from 54%-87.3% for combinations of task and resting-state data. When using only selected networks for identification, accuracy ranged from 98%-99% on resting-state data and from 80%-90% for combinations of task and resting-state data.

Another relevant study performs fingerprinting between scan sessions separated 1.5 years apart [Jalbrzikowski et al., 2020]. The authors use SVM and Elastic net regression classifiers to achieve accuracy of 89%-98% for both resting-state and task fMRI data. However, a feature selection method is applied that reproduces the approach of Finn et al. and includes only the 5% most relevant edges as input to the classifiers. These results indicate that novel fingerprinting approaches requiring less feature selection could provide relevant insights and comparisons to the findings already reported in the literature.

7. CONCLUSION

In this work, we applied GCN models on graph classification tasks using whole-brain resting-state and task fMRI data from two datasets. Our results show that ST-GCN, a spatial-temporal GCN model, consistently outperforms our baseline methods for binary classification, achieving state-of-the-art performance on the open-source HCP dataset. We employ ST-GCN in the investigation of reading disorders and cognition on the ACERTA dataset, and provide analysis on the biomarkers identified by the network as relevant to each classification tasks through the edge importance mechanism.

The obtained results show that GCN models capable of processing temporal information have improved performance over strictly spatial models regarding the analysis of BOLD signal data, to which the temporal component is key. We also show that time-series window slicing data augmentation can aid models to compensate for low data availability, improving classification performance. This improvement is shown by the comparisons made to baseline CNN models that allow no augmentation, particularly in the reading performance task, where the number of subjects is lowest. We validate our findings in the related literature, demonstrating that the proposed method constitutes a straightforward and effective option for fMRI analysis, including with regard to datasets of reduced size. However, we are not successful in our approach for subject fingerprinting, which performs worse than previous methods.

Our contributions are: (1) The application and comparison of GCN and baseline models in classification tasks using resting-state and task fMRI connectome data. (2) The first application of spatial-temporal GCNs to task fMRI data. (3) Analysis and validation of the biomarkers identified by the ST-GCN models regarding their relation to dyslexia and neural development. (4) A demonstration of the applicability of geometric deep learning in the study of multimodal brain connectomics in both small and large datasets.

Limitations

One practical limitation to the use of deep learning approaches in general by the larger neuroimaging community is their relative complexity of implementation and posterior analysis. Our approach only slightly alleviates the foremost concern regarding current state-of-the-art models, although analysis is made remarkably simple through the edge importance mechanism. An important limitation to our approach is the use of the same adjacency matrix for all examples, computed from the mean connectivity matrix across the subjects in the training set. We believe using the ROI connectivity for each individual subject would be essential for the fingerprinting task, where correlation between functional connectomes alone has been shown to produce remarkable results for resting-state data [Finn et al., 2015]. GCN models that deal with dynamic graph structures could be investigated for that end.

Regarding result reliability, although we remove ACERTA subjects presenting correlations between task stimuli and movement from our analysis, we did not test whether subjects could be classified based on their frame-to-frame motion rates, so that we can not exclude the possibility that such artifacts affect our results. It should also be noted that although we use data augmentation to synthetically increase our sample size and improve classification performance, the augmentation does not change underlying data distributions, so that our results are still susceptible to possible selection biases that can affect small datasets [Neuhaus and Popescu, 2018].

Future Work

Our graphs are constructed using fMRI data for both node and edge attributes. Although previous studies report no difference in classification performance when using graphs constructed using structural or diffusion MRI as edge attributes [Parisot et al., 2018a], such graph models could provide relevant insights from a neuroscientific perspective that lie outside the reach of the present work. To the best of our knowledge, no work in the literature has applied GCNs to multi-modal graphs using task fMRI data. The use of behavioral or socioeconomic data could also be explored through regression tasks or graph modeling methods allowing their inclusion. The original ST-GCN model employed a node partition mechanism detailed in Section 3.8.2 that was not used in this work. We believe this mechanism could be used to assimilate additional information, including non-neuroimaging data such as the aforementioned behavioral and socioeconomic data, but also hypothesis-driven information such as the presence of a node in a previously identified network.

As such, future work could investigate: (1) network architectures capable of processing graphs with dynamic topologies [Pareja et al., 2019, Xu et al., 2020, Sankar et al., 2019]; (2) graph construction using structural or diffusion MRI as edge attributes; (3) use node partition strategies, as described in the original ST-GCN publication [Yan et al., 2018]; (4) inclusion of behavioral data.

Our findings point to ST-GCN architectures as powerful alternatives to 2D and 3D CNNs for fMRI data analysis, providing state-of-the-art performance and explainable results. We believe further studies employing this and similar methods can contribute with relevant insights on the investigation of the varied aspects of brain structure and functionality emerging from the human connectome.

REFERENCES

[Abraham et al., 2014] Abraham, A. Pedregosa, F. Eickenberg, M. Gervais, P. Mueller, A. Kossaifi, J. Gramfort, A. Thirion, B. and Varoquaux, G. (Feb, 2014). Machine learning for neuroimaging with scikit-learn. *Frontiers in Neuroinformatics*, 8:14.

[Achard et al., 2006] Achard, S. Salvador, R. Whitcher, B. Suckling, J. and Bullmore, E. (Jan, 2006). A resilient, low-frequency, small-world human brain functional network with highly connected association cortical hubs. *Journal of Neuroscience*, 26(1):63–72.

[Alpaydin, 2010] Alpaydin, E. (2010). *Introduction to machine learning*. The MIT Press, Cambridge, USA, 2nd edition.

[Anirudh and Thiagarajan, 2017] Anirudh, R. and Thiagarajan, J. J. (Apr, 2017). Bootstrapping graph convolutional neural networks for autism spectrum disorder classification. arXiv:1704.07487.

[Arslan et al., 2018] Arslan, S. Ktena, S. I. Glocker, B. and Rueckert, D. (Jun, 2018). Graph saliency maps through spectral convolutional networks: application to sex classification with brain connectivity. arXiv:1806.01764.

[Barch et al., 2013] Barch, D. Burgess, G. Harms, M. Petersen, S. Schlaggar, B. Corbetta, M. Glasser, M. Curtiss, S. Dixit, S. Feldt, C. Nolan, D. Bryant, E. Hartley, T. Footer, O. Bjork, J. Poldrack, R. Smith, S. Johansen-Berg, H. Snyder, A. and Van Essen, D. (Oct, 2013). Function in the human connectome: task-fMRI and individual differences in behavior. *NeuroImage*, 80:169–189.

[Bassett et al., 2008] Bassett, D. S. Bullmore, E. Verchinski, B. A. Mattay, V. S. Weinberger, D. R. and Meyer-Lindenberg, A. (Sep, 2008). Hierarchical organization of human cortical networks in health and schizophrenia. *Journal of Neuroscience*, 28(37):9239–9248.

[Battaglia et al., 2018] Battaglia, P. W. Hamrick, J. B. Bapst, V. Sanchez-Gonzalez, A. Zambaldi, V. Malinowski, M. Tacchetti, A. Raposo, D. Santoro, A. Faulkner, R. Gulcehre, C. Song, F. Ballard, A. Gilmer, J. Dahl, G. Vaswani, A. Allen, K. Nash, C. Langston, V. Dyer, C. Heess, N. Wierstra, D. Kohli, P. Botvinick, M. Vinyals, O. Li, Y. and Pascanu, R. (Jun, 2018). Relational inductive biases, deep learning, and graph networks. arXiv:1806.01261.

[Bishop, 2006] Bishop, C. M. (2006). *Pattern recognition and machine learning information science and statistics)*, chapter 1.1, pages 10–11. Springer, Berlin, Germany, 1st edition.

[Bondurant et al., 1990] Bondurant, F. Cotler, H. Kulkarni, M. McArdle, C. and Harris, J. (Mar, 1990). Acute spinal cord injury. a study using physical examination and magnetic resonance imaging. *Spine*, 15(3):161–168.

[Bromley et al., 1993] Bromley, J. Bentz, J. W. Bottou, L. Guyon, I. LeCun, Y. Moore, C. Säckinger, E. and Shah, R. (Nov, 1993). Signature verification using a "siamese" time delay neural network. *International Journal of Pattern Recognition and Artificial Intelligence*, 7(4):669–688.

[Bruna et al., 2013] Bruna, J. Zaremba, W. Szlam, A. and LeCun, Y. (Dec, 2013). Spectral networks and locally connected networks on graphs. arXiv:1312.6203.

[Buchweitz et al., 2019] Buchweitz, A. Costa, A. C. Toazza, R. de Moraes, A. B. Cara, V. M. Esper, N. B. Aguzzoli, C. Gregolim, B. Dresch, L. F. Soldatelli, M. D. da Costa, J. C. Portuguez, M. W. and Franco, A. R. (Jan-Feb, 2019). Decoupling of the occipitotemporal cortex and the brain's default-mode network in dyslexia and a role for the cingulate cortex in good readers: a brain imaging study of brazilian children. *Developmental Neuropsychology*, 44(1):146–157. PMID: 29412010.

[Bullmore and Sporns, 2012] Bullmore, E. and Sporns, O. (Apr, 2012). The economy of brain network organization. *Nature Reviews Neuroscience*, 13:336–349.

[Bullmore and Sporns, 2009] Bullmore, E. and Sporns, O. (Feb, 2009). Complex brain networks: graph theoretical analysis of structural and functional systems. *Nature Reviews Neuroscience*, 10:186–198.

[Chen et al., 2020] Chen, T. Kornblith, S. Norouzi, M. and Hinton, G. (Feb, 2020). A simple framework for contrastive learning of visual representations. arXiv:2002.05709.

[Chopra et al., 2005] Chopra, S. Hadsell, R. and LeCun, Y. (2005). Learning a similarity metric discriminatively, with application to face verification. In: *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pages 539–546, San Diego, USA. IEEE.

[Cinciute, 2019] Cinciute, S. (Mar, 2019). Translating the hemodynamic response: why focused interdisciplinary integration should matter for the future of functional neuroimaging. *PeerJ*, 7:e6621.

[Cohen and Havlin, 2010] Cohen, R. and Havlin, S. (2010). *Complex networks: structure, robustness and function*. Cambridge University Press, Cambridge, UK, 1st edition.

[Cox, 1996] Cox, R. W. (Jun, 1996). Afni: software for analysis and visualization of functional magnetic resonance neuroimages. *Computers and Biomedical Research*, 29(3):162–73.

[de Bourbon-Teles et al., 2014] de Bourbon-Teles, J. Bentley, P. Koshino, S. Shah, K. Dutta, A. Malhotra, P. Egner, T. Husain, M. and Soto, D. (May, 2014). Thalamic control of human attention driven by memory and learning. *Current Biology*, 24(9):993–999.

[Defferrard et al., 2016] Defferrard, M. Bresson, X. and Vandergheynst, P. (Jun, 2016). Convolutional neural networks on graphs with fast localized spectral filtering. arXiv:1606.09375.

[Essen et al., 2013] Essen, D. C. V. Smith, S. M. Barch, D. M. Behrens, T. E. Yacoub, E. and Ugurbil, K. (Oct, 2013). The wu-minn human connectome project: an overview. *NeuroImage*, 80:62–79. Mapping the Connectome.

[Euler, 1741] Euler, L. (1741). Solutio problematis ad geometriam situs pertinentis. *Commentarii Academiae Scientiarum Imperialis Petropolitanae*, 8:128–140.

[Fey and Lenssen, 2019] Fey, M. and Lenssen, J. E. (Mar, 2019). Fast graph representation learning with pytorch geometric. arXiv:1903.02428.

[Finn et al., 2015] Finn, E. Shen, X. Scheinost, D. Rosenberg, M. Jessica, H. Chun, M. Papademetris, X. and Constable, R. (Oct, 2015). Functional connectome fingerprinting: identifying individuals using patterns of brain connectivity. *Nature Neuroscience*, 18:1664–1671.

[Freeman, 1977] Freeman, L. C. (Mar, 1977). A set of measures of centrality based on betweenness. *Sociometry*, 40(1):35–41.

[Gadgil et al., 2020] Gadgil, S. Zhao, Q. Pfefferbaum, A. Sullivan, E. V. Adeli, E. and Pohl, K. M. (2020). Spatio-temporal graph convolution for resting-state fmri analysis. In: *Proceedings of the 23rd International Conference on Medical Image Computing and Computer Assisted Intervention*, pages 528–538, New York, USA. Springer.

[Glasser et al., 2013] Glasser, M. Sotiropoulos, S. Wilson, J. Coalson, T. Fischl, B. Andersson, J. Xu, J. Jbabdi, S. Webster, M. Polimeni, J. DC, V. and Jenkinson, M. (Oct, 2013). The minimal preprocessing pipelines for the human connectome project. *NeuroImage*, 80:105.

[Glover, 2011] Glover, G. H. (Apr, 2011). Overview of functional magnetic resonance imaging. *Neurosurgery Clinics of North America*, 22(2):133–139.

[Gong and He, 2014] Gong, Q. and He, Y. (Aug, 2014). Depression, neuroimaging and connectomics: a selective overview. *Biological Psychiatry*, 77(3):223–235.

[Goodfellow et al., 2016] Goodfellow, I. Bengio, Y. and Courville, A. (2016). *Deep learning*. The MIT Press, Cambridge, USA, 1st edition.

[Goodfellow et al., 2014] Goodfellow, I. J. Pouget-Abadie, J. Mirza, M. Xu, B. Warde-Farley, D. Ozair, S. Courville, A. and Bengio, Y. (Jun, 2014). Generative adversarial networks. arXiv:1406.2661.

[Graves et al., 2013] Graves, A. rahman Mohamed, A. and Hinton, G. (Mar, 2013). Speech recognition with deep recurrent neural networks. arXiv:1303.5778.

[Greicius et al., 2003] Greicius, M. D. Krasnow, B. Reiss, A. L. and Menon, V. (Jan, 2003). Functional connectivity in the resting brain: a network analysis of the default mode hypothesis. In: *Proceedings of the National Academy of Sciences*, pages 253–258, New York, USA. National Academy of Sciences.

[Greicius et al., 2004] Greicius, M. D. Srivastava, G. Reiss, A. L. and Menon, V. (Mar, 2004). Default-mode network activity distinguishes alzheimer's disease from healthy aging: evidence from functional MRI. In: *Proceedings of the National Academy of Sciences*, pages 4637–4642, New York, USA. National Academy of Sciences.

[Hadsell et al., 2006] Hadsell, R. Chopra, S. and LeCun, Y. (2006). Dimensionality reduction by learning an invariant mapping. In: *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pages 1735–1742, New York, USA. IEEE.

[Hammond et al., 2009] Hammond, D. K. Vandergheynst, P. and Gribonval, R. (Dec, 2009). Wavelets on graphs via spectral graph theory. arXiv:0912.3848.

[Hayes, 1998] Hayes, M. H. (1998). *Schaum's outline of digital signal processing*. McGraw-Hill, Inc., New York, USA, 1st edition.

[Heinsfeld et al., 2018] Heinsfeld, A. S. Franco, A. R. Craddock, R. C. Buchweitz, A. and Meneguzzi, F. (Aug, 2018). Identification of autism spectrum disorder using deep learning and the abide dataset. *NeuroImage Clinical*, 17:16–23.

[Henaff et al., 2015] Henaff, M. Bruna, J. and LeCun, Y. (Jun, 2015). Deep convolutional networks on graph-structured data. arXiv:1506.05163.

[Hoffmann et al., 2003] Hoffmann, U. Globits, S. Schima, W. Loewe, C. Puig, S. Oberhuber, G. and Frank, H. (Oct, 2003). Usefulness of magnetic resonance imaging of cardiac and paracardiac masses. *The American Journal of Cardiology*, 92(7):890 – 895.

[Hong et al., 2019] Hong, Y. Kim, J. Chen, G. Lin, W. Yap, P.-T. and Shen, D. (Apr, 2019). Longitudinal prediction of infant diffusion mri data via graph convolutional adversarial networks. *IEEE Transactions on Medical Imaging*, PP:2717–2725.

[Horiguchi et al., 2020] Horiguchi, S. Ikami, D. and Aizawa, K. (Dec, 2020). Significance of softmax-based features in comparison to distance metric learning-based features. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 42(5):1279–1285.

[Hu et al., 2019] Hu, J. Kuang, Y. Liao, B. Cao, L. Dong, S. and Li, P. (Dec, 2019). A multichannel 2d convolutional neural network model for task-evoked fMRI data classification. *Computational Intelligence and Neuroscience*, 2019:1–9.

[Huang and Chung, 2020] Huang, Y. and Chung, A. C. S. (Sep, 2020). Edge-variational graph convolutional networks for uncertainty-aware disease prediction. arXiv:2009.02759.

[Huettel et al., 2004] Huettel, S. Song, A. and McCarthy, G. (2004). *Functional magnetic resonance imaging, Second Edition*. Sinauer Associates, Inc., Sunderland, USA, 1st edition.

[Ibrahim et al., 2019] Ibrahim, C. Rubin-Kahana, D. S. Pushparaj, A. Musiol, M. Blumberger, D. M. Daskalakis, Z. J. Zangen, A. and Le Foll, B. (Jul, 2019). The insula: a brain stimulation target for the treatment of addiction. *Frontiers in Pharmacology*, 10:720.

[Jalbrzikowski et al., 2020] Jalbrzikowski, M. Liu, F. Foran, W. Klei, L. Calabro, F. J. Roeder, K. Devlin, B. and Luna, B. (Jul, 2020). Functional connectome fingerprinting accuracy in youths and adults is similar when examined on the same day and 1.5-years apart. *Human Brain Mapping*, 41(15):4187–4199.

[Jun et al., 2020] Jun, E. Na, K.-S. Kang, W. Lee, J. Suk, H.-I. and Ham, B.-J. (Aug, 2020). Identifying resting-state effective connectivity abnormalities in drug-naïve major depressive disorder diagnosis via graph convolutional networks. *Human Brain Mapping*, 41(17):4997–5014.

[Kaufmann et al., 2017] Kaufmann, T. Alnæs, D. Doan, N. Brandt, C. Andreassen, O. and Westlye, L. (Feb, 2017). Delayed stabilization and individualization in connectome development are related to psychiatric disorders. *Nature neuroscience*, 20:503–504.

[Kawahara et al., 2017] Kawahara, J. Brown, C. P Miller, S. Booth, B. Chau, V. Grunau, R. Zwicker, J. and Hamarneh, G. (Sep, 2017). Brainnetcnn: convolutional neural networks for brain networks: towards predicting neurodevelopment. *NeuroImage*, 146:1038–1049.

[Kazi et al., 2019] Kazi, A. shekarforoush, S. krishna, S. A. Burwinkel, H. Vivar, G. Kortuem, K. Ahmadi, S.-A. Albarqouni, S. and Navab, N. (Mar, 2019). Inceptiongcn: receptive field aware graph convolutional network for disease prediction. arXiv:1903.04233.

[Kim and Ye, 2020] Kim, B.-H. and Ye, J. C. (Jun, 2020). Understanding graph isomorphism network for rs-fmri functional connectivity analysis. *Frontiers in Neuroscience*, 14:630.

[Kingma and Ba, 2014] Kingma, D. P. and Ba, J. (Dec, 2014). Adam: a method for stochastic optimization. arXiv:1412.6980.

[Kipf and Welling, 2016] Kipf, T. N. and Welling, M. (Sep, 2016). Semi-supervised classification with graph convolutional networks. arXiv:1609.02907.

[Kitchenham and Charters, 2007] Kitchenham, B. A. and Charters, S. (2007). Guidelines for performing systematic literature reviews in software engineering. Technical Report,

School of Computer Science and Mathematics, Keele University and Department of Computer Science, Durham University.

[Koch, 2015] Koch, G. (2015). Siamese neural networks for one-shot image recognition. (master thesis), Graduate Department of Computer Science, University of Toronto, Toronto, Canada.

[Krizhevsky et al., 2012] Krizhevsky, A. Sutskever, I. and Hinton, G. E. (2012). Imagenet classification with deep convolutional neural networks. In: *Proceedings of the 25th International Conference on Neural Information Processing Systems*, pages 1097–1105, Red Hook, USA. Curran Associates Inc.

[Ktena et al., 2017] Ktena, S. I. Parisot, S. Ferrante, E. Rajchl, M. Lee, M. Glocker, B. and Rueckert, D. (Mar, 2017). Distance metric learning using graph convolutional networks: application to functional brain networks. arXiv:1703.02161.

[Küseler et al., 1998] Küseler, A. Pedersen, T. Herlin, T. and Gelineck, J. (Jul, 1998). Contrast enhanced magnetic resonance imaging as a method to diagnose early inflammatory changes in the temporomandibular joint in children with juvenile chronic arthritis. *The Journal of Rheumatology*, 25(7):1406–1412.

[Lauterbur, 1973] Lauterbur, P. (Mar, 1973). Image formation by induced local interactions: examples employing nuclear magnetic resonance. *Nature*, 242(5394):190–191.

[Le Bihan et al., 2001] Le Bihan, D. Mangin, J.-F. Poupon, C. Clark, C. A. Pappata, S. Molko, N. and Chabriat, H. (Apr, 2001). Diffusion tensor imaging: concepts and applications. *Journal of Magnetic Resonance Imaging*, 13(4):534–546.

[Lecun, 1989] Lecun, Y. (1989). Generalization and network design strategies. Technical Report, Department of Computer Science, University of Toronto.

[Lee et al., 2019] Lee, P. Choi, M.-w. Kim, D. Lee, S. Jeong, H.-G. and Han, C. (2019). Deep learning based decomposition of brain networks. In: *Proceedings of the 1st International Conference on Artificial Intelligence in Information and Communication*, pages 349–354, New York, USA. IEEE.

[Li et al., 2018] Li, X. Dvornek, N. C. Papademetris, X. Zhuang, J. Staib, L. H. Ventola, P. and Duncan, J. S. (2018). 2-channel convolutional 3d deep neural network (2cc3d) for fmri analysis: Asd classification and feature learning. In: *Proceedings of the 15th International Symposium on Biomedical Imaging*, pages 1252–1255, New York, USA. IEEE.

[Li et al., 2020] Li, Y. Shafipour, R. Mateos, G. and Zhang, Z. (2020). Supervised graph representation learning for modeling the relationship between structural and functional brain connectivity. In: *Proceedings of the 45th International Conference on Acoustics, Speech and Signal Processing*, pages 9065–9069, New York, USA. IEEE.

[Li et al., 2018] Li, Y. Yu, R. Shahabi, C. and Liu, Y. (Jul, 2018). Diffusion convolutional recurrent neural network: data-driven traffic forecasting. arXiv:1707.01926.

[Liu et al., 2017] Liu, A. Liu, B. Lee, D. Weissman, M. Posner, J. Cha, J. and Yoo, S. (2017). Machine learning aided prediction of family history of depression. In: *Proceedings of the New York Scientific Data Summit*, pages 1–4, New York, USA. IEEE.

[Liu et al., 2020] Liu, J. Tan, G. Lan, W. and Wang, J. (Nov, 2020). Identification of early mild cognitive impairment using multi-modal data and graph convolutional networks. *BMC Bioinformatics*, 21:123.

[Liu et al., 2019] Liu, J. Zeng, D. Lu, M. and Wang, J. (2019). Mild cognitive impairment identification based on multi-view graph convolutional networks. In: *Proceedings of the 7th International Conference on Advanced Cloud and Big Data*, pages 309–314, New York, USA. IEEE.

[Liu et al., 2008] Liu, Y. Liang, M. Zhou, Y. He, Y. Hao, Y. Song, M. Yu, C. Liu, H. Liu, Z. and Jiang, T. (Feb, 2008). Disrupted small-world networks in schizophrenia. *Brain*, 131(4):945–961.

[Lynch et al., 2013] Lynch, C. J. Uddin, L. Q. Supekar, K. Khouzam, A. Phillips, J. and Menon, V. (Aug, 2013). Default mode network in childhood autism: posteromedial cortex heterogeneity and relationship with social deficits. *Biological Psychiatry*, 74(3):212–219. Oxytocin and Autism.

[Ma et al., 2018] Ma, G. Ahmed, N. K. Willke, T. Sengupta, D. Cole, M. W. Turk-Browne, N. B. and Yu, P. S. (Nov, 2018). Similarity learning with higher-order graph convolutions for brain network analysis. 1811.02662.

[Magner et al., 2019] Magner, A. Baranwal, M. and III, A. O. H. (May, 2019). Fundamental limits of deep graph convolutional networks. arXiv:1910.12954.

[Makridakis, 2017] Makridakis, S. (Jan, 2017). The forthcoming artificial intelligence (ai) revolution: its impact on society and firms. *Futures*, 90:46–60.

[Manessi et al., 2020] Manessi, F. Rozza, A. and Manzo, M. (Jan, 2020). Dynamic graph convolutional networks. *Pattern Recognition*, 97:107000.

[Mitchell, 1997] Mitchell, T. M. (1997). *Machine learning*. McGraw-Hill, Inc., New York, USA, 1st edition.

[Nandalur et al., 2007] Nandalur, K. R. Dwamena, B. A. Choudhri, A. F. Nandalur, M. R. and Carlos, R. C. (Oct, 2007). Diagnostic performance of stress cardiac magnetic resonance imaging in the detection of coronary artery disease. *Journal of the American College of Cardiology*, 50(14):1343–1353.

[Neuhaus and Popescu, 2018] Neuhaus, A. and Popescu, F. (Feb, 2018). Sample size, model robustness, and classification accuracy in diagnostic multivariate neuroimaging analyses. *Biological Psychiatry*, 84(11):e81–e82.

[Nickerson, 2018] Nickerson, L. (Dec, 2018). Replication of resting state-task network correspondence and novel findings on brain network activation during task fmri in the human connectome project study. *Scientific Reports*, 8:17543.

[Pareja et al., 2019] Pareja, A. Domeniconi, G. Chen, J. Ma, T. Suzumura, T. Kanezashi, H. Kaler, T. Schardl, T. B. and Leiserson, C. E. (Feb, 2019). Evolvegcn: evolving graph convolutional networks for dynamic graphs. arXiv:1902.10191.

[Parisot et al., 2018a] Parisot, S. Ktena, S. I. Ferrante, E. Lee, M. Guerrero, R. Glocker, B. and Rueckert, D. (Aug, 2018a). Disease prediction using graph convolutional networks: application to autism spectrum disorder and alzheimer's disease. *Medical Image Analysis*, 48:117–130.

[Parisot et al., 2018b] Parisot, S. Ktena, S. I. Ferrante, E. Lee, M. Guerrero, R. Glocker, B. and Rueckert, D. (Jun, 2018b). Disease prediction using graph convolutional networks: application to autism spectrum disorder and alzheimer's disease. *Medical Image Analysis*, 48:117–130.

[Pascanu et al., 2013] Pascanu, R. Mikolov, T. and Bengio, Y. (2013). On the difficulty of training recurrent neural networks. In: *Proceedings of the 30th International Conference on International Conference on Machine Learning*, pages 1310–1318, Atlanta, USA. JMLR.org.

[Pugh et al., 2000] Pugh, K. Mencl, W. Shaywitz, B. Shaywitz, S. Fulbright, R. Constable, R. Skudlarski, P. Marchione, K. Jenner, A. Fletcher, J. Liberman, A. Shankweiler, D. Katz, L. Lacadie, C. and Gore, J. (Feb, 2000). The angular gyrus in developmental dyslexia: task-specific differences in functional connectivity within posterior cortex. *Psychological Science*, 11:51–56.

[Qu et al., 2021] Qu, G. Xiao, L. Hu, W. Zhang, K. Calhoun, V. D. and Wang, Y.-P. (Jan, 2021). Ensemble manifold based regularized multi-modal graph convolutional network for cognitive ability prediction. arXiv:2101.08316.

[Radanovic et al., 2003] Radanovic, M. Azambuja, M. Mansur, L. L. Porto, C. S. and Scaff, M. (Mar, 2003). Thalamus and language: interface with attention, memory and executive functions. *Arquivos de Neuro-Psiquiatria*, 61:34–42.

[Raichle et al., 2001] Raichle, M. E. MacLeod, A. M. Snyder, A. Z. Powers, W. J. Gusnard, D. A. and Shulman, G. L. (Jan, 2001). A default mode of brain function. In: *Proceedings of*

the National Academy of Sciences, pages 676–682, New York, USA. National Academy of Sciences.

[Richards et al., 2019] Richards, B. Lillicrap, T. Beaudoin, P. Bengio, Y. Bogacz, R. Christensen, A. Clopath, C. Costa, R. Berker, A. Ganguli, S. Gillon, C. Hafner, D. Kepecs, A. Kriegeskorte, N. Latham, P. Lindsay, G. Miller, K. Naud, R. Pack, C. and Kording, K. (Nov, 2019). A deep learning framework for neuroscience. *Nature Neuroscience*, 22:1761–1770.

[Rissman et al., 2004] Rissman, J. Gazzaley, A. and D'Esposito, M. (Oct, 2004). Measuring functional connectivity during distinct stages of a cognitive task. *NeuroImage*, 23(2):752–763.

[Sankar et al., 2019] Sankar, A. Wu, Y. Gou, L. Zhang, W. and Yang, H. (Jun, 2019). Dynamic graph representation learning via self-attention networks. arXiv:1812.09430.

[Scarselli et al., 2009] Scarselli, F. Gori, M. Tsoi, A. C. Hagenbuchner, M. and Monfardini, G. (Jan, 2009). The graph neural network model. *IEEE Transactions on Neural Networks*, 20(1):61–80.

[Schuyler et al., 2010] Schuyler, B. Ollinger, J. M. Oakes, T. R. Johnstone, T. and Davidson, R. J. (Jan, 2010). Dynamic causal modeling applied to fmri data shows high reliability. *NeuroImage*, 49(1):603–611.

[Seo et al., 2016] Seo, Y. Defferrard, M. Vandergheynst, P. and Bresson, X. (Dec, 2016). Structured sequence modeling with graph convolutional recurrent networks. arXiv:1612.07659.

[Shen et al., 2013] Shen, X. Tokoglu, F. Papademetris, X. and Constable, R. (Nov, 2013). Groupwise whole-brain parcellation from resting-state fmri data for network node identification. *NeuroImage*, 82:403–415.

[Shimada et al., 2001] Shimada, T. Shimada, K. Sakane, T. Ochiai, K. Tsukihashi, H. Fukui, M. ichi Inoue, S. Katoh, H. Murakami, Y. Ishibashi, Y. and Maruyama, R. (May, 2001). Diagnosis of cardiac sarcoidosis and evaluation of the effects of steroid therapy by gadolinium-dtpa–enhanced magnetic resonance imaging. *The American Journal of Medicine*, 110(7):520–527.

[Siegel, 2001] Siegel, M. J. (Jul, 2001). Magnetic resonance imaging of musculoskeletal soft tissue masses. *Radiologic Clinics of North America*, 39(4):701–720.

[Simonyan and Zisserman, 2015] Simonyan, K. and Zisserman, A. (Apr, 2015). Very deep convolutional networks for large-scale image recognition. arXiv:1409.1556.

[Sporns and Kötter, 2004] Sporns, O. and Kötter, R. (Oct, 2004). Motifs in brain networks. *PLOS Biology*, 2(11):e369.

[Sporns et al., 2005] Sporns, O. Tononi, G. and Kötter, R. (Sep, 2005). The human connectome: a structural description of the human brain. *PLOS Computational Biology*, 1(4):e42.

[Srivastava et al., 2014] Srivastava, N. Hinton, G. Krizhevsky, A. Sutskever, I. and Salakhutdinov, R. (Jun, 2014). Dropout: a simple way to prevent neural networks from overfitting. *Journal of Machine Learning Research*, 15:1929–1958.

[Stoitsis et al., 2008] Stoitsis, J. Giannakakis, G. A. Papageorgiou, C. Nikita, K. S. Rabavilas, A. and Anagnostopoulos, D. (Apr, 2008). Evidence of a posterior cingulate involvement (brodmann area 31) in dyslexia: a study based on source localization algorithm of event-related potentials. *Progress in Neuro-Psychopharmacology and Biological Psychiatry*, 32(3):733–738.

[Suoranta et al., 2013] Suoranta, S. Holli-Helenius, K. Koskenkorva, P. Niskanen, E. Könönen, M. Aikiä, M. Eskola, H. Kälviäinen, R. and Vanninen, R. (Jul, 2013). 3d texture analysis reveals imperceptible mri textural alterations in the thalamus and putamen in progressive myoclonic epilepsy type 1, epm1. *PloS one*, 8:e69905.

[Sutskever et al., 2013] Sutskever, I. Martens, J. Dahl, G. and Hinton, G. (2013). On the importance of initialization and momentum in deep learning. In: *Proceedings of the 30th International Conference on International Conference on Machine Learning*, pages 1139–1147, Atlanta, USA. JMLR.org.

[Sutton and Barto, 2014] Sutton, R. S. and Barto, A. G. (2014). *Reinforcement learning: an introduction*. The MIT press, Cambridge, USA, 2nd edition.

[Tan et al., 2005] Tan, P.-N. Steinbach, M. and Kumar, V. (2005). *Introduction to data mining*. Addison-Wesley Longman Publishing Co., Inc., Boston, USA, 1st edition.

[Thulborn et al., 1982] Thulborn, K. R. Waterton, J. C. Matthews, P. M. and Radda, G. K. (Feb, 1982). Oxygenation dependence of the transverse relaxation time of water protons in whole blood at high field. *Biochimica et Biophysica Acta*, 714(2):265–270.

[Uddin, 2014] Uddin, L. (Nov, 2014). Salience processing and insular cortical function and dysfunction. *Nature Reviews Neuroscience*, 1:55–61.

[Uddin et al., 2017] Uddin, L. Nomi, J. Hébert-Seropian, B. Ghaziri, J. and Boucher, O. (Jul, 2017). Structure and function of the human insula. *Journal of Clinical Neurophysiology*, 34:300–306.

[Valenchon and Coates, 2019] Valenchon, J. and Coates, M. (2019). Multiple-graph recurrent graph convolutional neural network architectures for predicting disease outcomes. In: *Proceedings of the 44th IEEE International Conference on Acoustics, Speech and Signal Processing*, pages 3157–3161, New York, USA. IEEE.

[Varoquaux and Craddock, 2013] Varoquaux, G. and Craddock, R. C. (Oct, 2013). Learning and comparing functional connectomes across subjects. *NeuroImage*, 80:405–415.

[Vaswani et al., 2017] Vaswani, A. Shazeer, N. Parmar, N. Uszkoreit, J. Jones, L. Gomez, A. N. Kaiser, L. and Polosukhin, I. (Jun, 2017). Attention is all you need. arXiv:1706.03762.

[Wattjes, 2011] Wattjes, M. P. (Sep, 2011). Structural mri. *International Psychogeriatrics*, 23(S2):S13–S24.

[Wilson and Martinez, 2003] Wilson, D. R. and Martinez, T. R. (Dec, 2003). The general inefficiency of batch training for gradient descent learning. *Neural Netw.*, 16(10):1429–1451.

[Wu et al., 2020] Wu, Z. Pan, S. Chen, F. Long, G. Zhang, C. and Yu, P. S. (Mar, 2020). A comprehensive survey on graph neural networks. *IEEE Transactions on Neural Networks and Learning Systems*, 32(1):4–24.

[Xu et al., 2020] Xu, D. Ruan, C. Korpeoglu, E. Kumar, S. and Achan, K. (Feb, 2020). Inductive representation learning on temporal graphs. arXiv:2002.07962.

[Yan et al., 2018] Yan, S. Xiong, Y. and Lin, D. (Jan, 2018). Spatial temporal graph convolutional networks for skeleton-based action recognition. arXiv:1801.07455.

[Yao et al., 2021] Yao, D. Sui, J. Wang, M. Yang, E. Jiaerken, Y. Luo, N. Yap, P. T. Liu, M. and Shen, D. (Jan, 2021). A mutual multi-scale triplet graph convolutional network for classification of brain disorders using functional or structural connectivity. *IEEE Transactions on Medical Imaging*, 40(4):1279–1289.

[Yu et al., 2018] Yu, B. Yin, H. and Zhu, Z. (2018). Spatio-temporal graph convolutional networks: a deep learning framework for traffic forecasting. In: *Proceedings of the 27th International Joint Conference on Artificial Intelligence*, pages 3634–3640, Palo Alto, USA. The AAAI Press.

[Yuste, 2015] Yuste, R. (Jul, 2015). From the neuron doctrine to neural networks. *Nature Reviews Neuroscience*, 16:487–497.

[Zhang et al., 2019a] Zhang, S. Yao, L. Sun, A. and Tay, Y. (Feb, 2019a). Deep learning based recommender system. *ACM Computing Surveys*, 52(1):1–38.

[Zhang et al., 2019b] Zhang, X. S. Chou, J. and Wang, F. (May, 2019b). Integrative analysis of patient health records and neuroimages via memory-based graph convolutional network. arXiv:1809.06018.

[Zhang et al., 2019c] Zhang, X. S. He, L. Chen, K. Luo, Y. Zhou, J. and Wang, F. (May, 2019c). Multi-view graph convolutional network and its applications on neuroimage analysis for parkinson's disease. arXiv:1805.08801.

[Zhang and Huang, 2019] Zhang, Y. and Huang, H. (2019). New graph-blind convolutional network for brain connectome data analysis. In: *Proceedings of the 26th Information Processing in Medical Imaging*, pages 669–681, New York, USA. Springer.

[Zimmer et al., 1985] Zimmer, W. D. Berquist, T. H. McLeod, R. A. Sim, F. H. Pritchard, D. J. Shives, T. C. Wold, L. E. and May, G. R. (Jun, 1985). Bone tumors: magnetic resonance imaging versus computed tomography. *Radiology*, 155(3):709–718. PMID: 4001374.