KING'S College LONDON

University of London

6CCYB070 BEng Research Project

Deep Learning of neonatal connectivity to predict neurodevelopmental outcome

Yassine Taoudi Benchekroun

Supervisors:
Dr. Maria Deprez
Dr. Dafnis Batalle

Project Report submitted in partial fulfilment of the Bachelor of Engineering degree in
Biomedical Engineering

April 2019

Acknowledgments

I would like to express my gratitude to my two supervisors Dr Maria Deprez and Dr Dafnis Batalle, who have offered me the chance to work on a project I greatly enjoyed. They have both guided me throughout this project and have always provided invaluable suggestions and support. I would also like to particularly thank Mrs. Irina Grigorescu, PhD student, without whom the last section of this project would not have been possible, and for her patience and resourcefulness.

Abstract

Perinatal brain development provides important roots for expansion of motor, cognitive and behavioral abilities. It is also a period where many irreversible disorders can develop. State of the art Deep Learning methods can be extremely useful to predict neurodevelopmental outcomes from complex data such as brain connectivity – thus allowing early identification of potential disorders or delays and enabling more efficient and faster treatment. In this work, we attempt to predict various neurodevelopmental outcomes using Dense Fully Connected Neural Networks with the developing Human Connectome Project (dHCP) data set. Firstly, we achieve accurate prediction of post menstrual age at scan from structural connectome with (Mean Absolute Average of 0.7 weeks on term and preterm infants, and 0.9 weeks on preterm infants only). We also accurately predict gestational age at birth from structural connectome (Mean Absolute Average of 1.1 weeks). We also show that preterm infants have significant developmental delays in connectivity compared to term infants. Secondly, we focus on Neurodevelopmental assessment (Bayley III Score) and attempt to replicate the findings of previous studies that were able to accurately predict neurodevelopmental score from structural connectome (Relative error $\approx 5\%$). We do not achieve satisfactory results on this task with the dHCP data set and come to question some methods used in the publication. Finally, we build a term against preterm Dense Neural Network classifier (reaching accuracy of 90% on testing) that we interpret with Layer wise Relevance Propagation. We are able to identify core connections that are most affected by gestational age – these findings agree with previously published work.

Project Plan

Problem:

Understanding how cognitive function emerges from a specific anatomical substrate of the brain is one of the current research goals of Neuroscience. Diffusion MRI offers insights into structural connectivity in the brain while functional MRI provides key information about the functional connectivity. However, the exact relationship between them is still not fully understood. The problem is particularly important for neonates and their neurodevelopment: indeed, improving our understanding of how structural and functional connectivity are linked will help in understanding the normal development of cognition and how preterm birth can impair neurocognitive development.

State of the art:

Machine learning is an increasingly important tool in the field of Neuroscience. Recently, convolutional neural networks with an edge-to-edge, edge-to-node, node-to-graph convolutional filters that leverage the topological locality of structural brain networks were successfully used to predict cognitive and motor development (Kawahara et al., 2017). We hypothesize that neural networks can also be used to predict the complete functional connectivity matrix based on the structural connectivity matrix.

Aims:

Build a deep learning model to predict the functional connectivity of the brain from the structural connectivity and see how this evolves depending on gestational age. The

developed tools will facilitate deeper understanding of the impact of prematurity on neurocognitive development

Work plan:

1. Learn how to use the Deep Neural Network libraries such as Tensor Flow or Nifty Net and implement elementary neural networks. Study more complex network architectures such as auto encoders that will help to regularise the fitted models.

2. Develop and tune a deep neural network for prediction of gestational age at scan and birth from structural and/or functional and connectivity matrices.

3. Analyse the developmental patterns that drive the predictions.

4. Train convolutional neural network to predict functional connectivity matrix from structural connectivity matrix using regularised deep neural networks such as auto encoders.

Deliverables:

Accurate deep neural network tool for prediction of gestational age from structural and functional connectivity. Insights into our ability to predict functional connectivity from structural connectivity using regularised deep neural networks.

Evaluation

Accuracy of the predictions of the gestational age and connectivity matrices. Accuracy of prediction of Functional connectivity from structural connectivity.

Project Timeline:

	September	October	November	December	January	February	March	April
Phase 1								
Phase 2								
Phase 3								
Phase 4								
Report								

Project Plan Deviation

For this project, phase 1 and 2 have been executed on time as initially planned. Indeed, proficiency was reached in the deep learning library Keras (phase 1) within a few weeks after the start of the project. The first accurate models predicting age at birth and age at scan from structural connectome (phase 2) were obtained in November. Training convolutional neural networks to predict functional connectivity from structural connectivity (phase 4) was also attempted on time as initially planned, but this was put aside due to the high complexity of the task. Another reason for putting aside phase 4 was the release of neurodevelopmental outcomes data from the dHCP, which coincided with the publication of (Girault et al. 2019) who achieve accurate prediction neurodevelopmental outcome from structural connectome. Great motivation was thus found in attempting to replicate their results with the dHCP data set. Phase 3 was also executed as initially planned, although with methods that were not originally known when writing the project plan. This also took longer than originally planned. Overall, although the initial project plan was not thoroughly respected, numerous interesting experiments and findings that were not initially planned came through during the realisation of this project.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

1.1 Background

1.1.1 Brain Connectivity

Brief history of modern Neuroscience

Two radically opposed views on Neural Organization caused an important controversy during the second part of the nineteenth century. The first theory, called "Neuron Doctrine" mainly supported by Scientists such as Ramon y Cajal and Camillo Golgi, claimed that nervous system was composed of individual cellular units called nerve cell, or neuron. The other theory, called "Reticular Theory", rejected the individuality of the nerve cell, and claimed that the entire nervous system was a continuous nerve network. The neuron doctrine, by the end of the 19th century, has been accepted and has remained the foundation of modern Neuroscience (Sporns 2010). Although the theory supported by Cajal and Golgi (for which they both won a Nobel Prize in 1906) still firmly holds today; the way in which these individual nerve cells achieve continuity and collective action within the brain, thus allowing cognitive function, is far from being resolved.

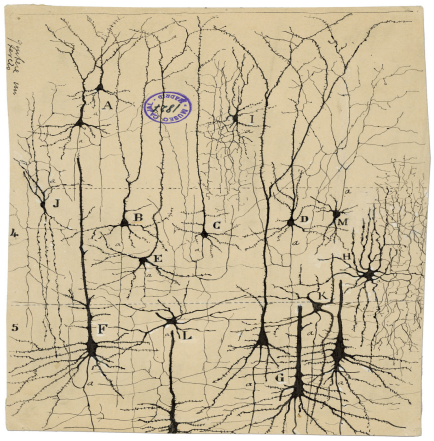


Figure 1.1: Ramon y Cajal Drawing of Neurons in 1906 (Courtesy of Cajal Institute)

The Connectome Idea

Network science is a relatively new field of study which allows consideration of individual elements (represented by "nodes") as well as the connections between these elements (represented by "edges"). This scientific paradigm has greatly benefited from the rapidly growing computing power in the last 30 years, and it's consequently had an important impact on various fields of study, including social science, economics, mathematics, physics and neuroscience (Watts 2004). Indeed, in Neuroscience, structural information is needed (i.e. knowledge of individual nerve cells) as well as a how these individual elements interact to achieve collective action. Network theory has thus appeared to be a very promising way to address these problems. Plans of building a comprehensive map of the brain's neurons and their connections hence started to emerge. The development of Electron Microscopy allowed John Graham White et al. in 1986 to build the first full nervous system map of a specie: the Caenorhabditis elegans, a roundworm measuring less than 1mm (White et al. 1986). Every single 302 neurons that this roundworm possesses has been mapped along with its connections. This is, to this day, the only specie of which we have a full nervous system map.

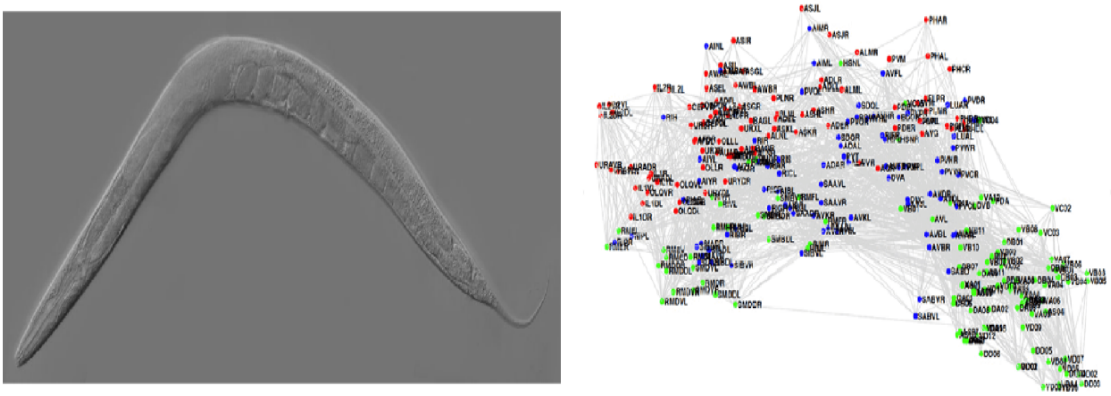


Figure 1.2: On the Left, image of Caenorhabditis elegans, on the right, its full structural connectome (White et al. 1986)

Efforts on this have carried on and soon reached the human brain, notably with the research of Olaf Sporns et al. who published a landmark paper in 2005 where the importance of working towards building a detailed map of the human brain - a "Connectome" - is strongly emphasized:

"To understand the functioning of a network, one must know its elements and their interconnections. The purpose of this article is to discuss research strategies aimed at a comprehensive structural description of the network of elements and connections forming the human brain. We propose to call this dataset the human "connectome," and we argue that it is fundamentally important in cognitive neuroscience and neuropsychology." (Sporns et al. 2005).

Different types of brain connectivities

There are 2 main different types of brain connectivities that give various information about the structure of the brain and its functioning.

- **Structural Connectivity**: refers to a set of structural connections linking neu-

ral elements. Usually is weighted and hence has information on the strength of the connections (Sporns 2010). It can be visualized with techniques such as diffusion Magnetic Resonance Imaging or Tractography.

- **Functional Connectivity**: It is a brain mapping technique that statistically evaluates regional interaction in resting state or task-based state. The basis of all functional connectivity is time series from neural recordings. It is, unlike structural connectivity, very dependant on time, and can change within milliseconds. The statistical dependence shown in functional connectivity is a correlation, not a causation. It can be visualized with techniques such as functional Magnetic Resonance Imaging (Sporns 2010, Friston et al. 1993).

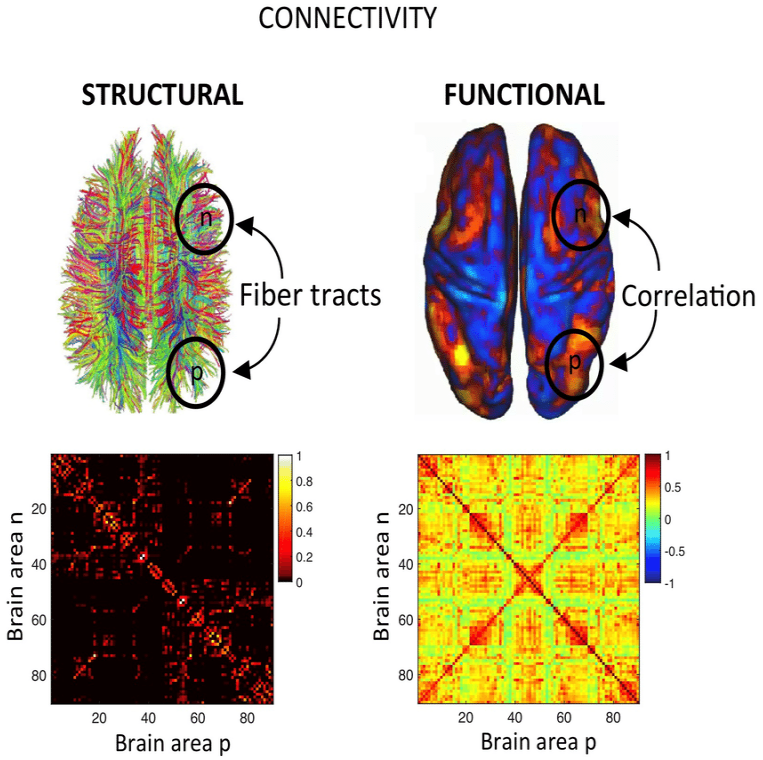


Figure 1.3: From left to right, structural and functional connectivities. (Cabral et al. 2017)

We know that both functional and structural connectivity are intrinsically linked to each other (Honey et al. 2010), however, the exact way in which they interact is still not fully understood in the world of Neuroscience. Recent progress in statistical physics and graph/network theory have thus been translated to analysis of the Neuroimaging data to improve our comprehension of structural and functional connectivities. Indeed, many complex network properties and have been successfully identified in connectivity data such as small-worldness (Bassett & Bullmore 2006). Recent studies have also showed how various disorders such as Alzheimer's disease or Schizophrenia are associated to specific differences in brain connectivity (delEtoile et al, 2017; Lynall, Basset et al 2010). Network theory as a tool to study brain connectivity is, in consequence of its success in previous years, rapidly growing and proving to be a powerful framework to have greater insight into the way the brain functions.

Imaging the Brain

Until the 1900's, observing the brain meant physically analysing it, by using various observation tools such as microscopes and developing innovative techniques for staining and sectioning nerve tissue cells. It is only with the apparition of imaging techniques that greater insight into the brain started to emerge. Today, many different imaging techniques allow to have different types of images of brain structure and activity; the most impactful one in Neuroscience research is Magnetic Resonance Imaging (MRI).

Magnetic Resonance Imaging (MRI) is the most versatile and detailed soft tissue imaging method that we have today. It is a safe non-invasive method based on the phenomenon of Nuclear Magnetic Resonance. Nuclei have a quantum mechanical property called spin that make them behave like magnets. By placing nuclei in a very strong magnetic field, a tiny excess proportion of them will align with the magnetic

field, resulting in a small net magnetic moment that is measurable. To have a better signal, MRI mainly focuses on hydrogen nuclei (H1) which are in abundance in all body tissue. There is no "Standard MRI scan", there are a great number of different contrasts and parameters to observe various things. Two recently developed methods have been widely used in Brain Connectivity research:

Diffusion Magnetic Resonance Imaging (dMRI) is a technique allowing study of the brain anatomy. dMRI takes advantage of the difference in diffusion of water molecules along the axonal bundles, which provides anatomical information about the white matter and allows to estimate anatomical organization of the brain.

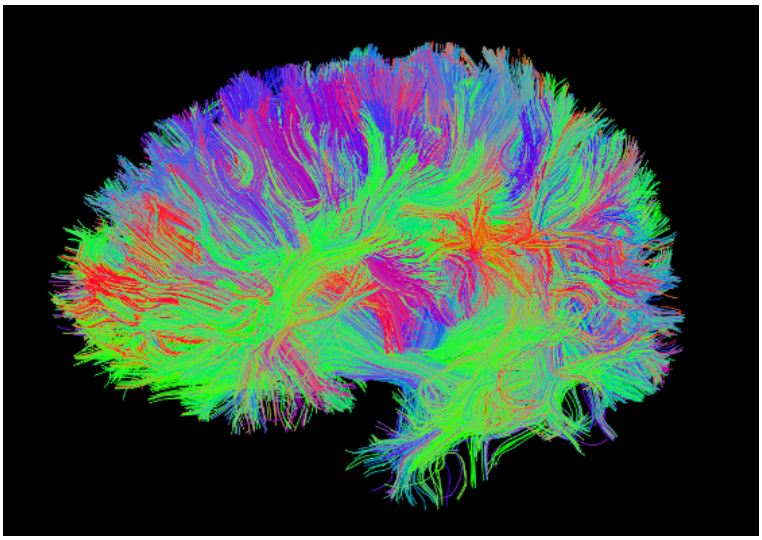


Figure 1.4: Tractography image (adapted from (Sporns 2010)

Functional Magnetic Resonance Imaging (fMRI) is a technique allowing to measure brain activity. It relies on the fact that neuronal activation and cerebral blood flow are coupled. fMRI takes advantage of the differences in magnetic field of oxygenated and deoxygenated blood in the brain in order to measure functional activity. This method is called Bold Oxygenated Level Dependant (BOLD) imaging

(Sashank J. Reddi paper, Wikipedia functional MRI).

Neurodevelopment

Human brain functional and structural early development provides highly important roots for the expansion of motor, cognitive and behavioural abilities – much of which happens perinatal period. Studying Neurodevelopment thus appears fundamental to understand adult brain functionality. The recent progresses in Neuroimaging have enabled Neurodevelopmental science to greatly progress. Indeed, with the advance of technologies like MRI, it is now possible to safely acquire detailed images of brain at prenatal and early post-natal period. However, acquiring these images has proven to be very difficult; indeed, issues like motion of the foetus/infant make the acquisition of the scan challenging. Various approaches to motion correction using both prospective (Andersson et al. 2017, Ferrazzi et al. 2014) and retrospective techniques have allowed acquisition of reliable images (Cordero-Grande et al. 2016). Such progresses have allowed the rise of important research project such as the developing Human Connectome Project (dHCP), led by King's College London, Imperial College London and Oxford University. The dHCP aims to create the first 4-dimensional connectome of early life. They have to date scanned more than 700 infants at a prenatal or early postnatal period using state of the art MRI techniques. The dHCP data has been used for execution of this work. These studies are allowing invaluable insights into the drastic changes undergone by the brain during early development. Indeed, the perinatal period is a time of establishment and consolidation of brain connectivity where fundamental connections are established (Batalle et al. 2018). Studying the development of brain networks is of particular importance in the case of preterm birth where chances to develop cognitive disorders or delays is important – it is hence crucial to understand the impact of shortened neurodevelopment and its long-term effects. This allows

risk assessment for the development of cognitive delays and disorders such as autism (Brown et al. 2018, Girault et al. 2019) The newly trending Machine Learning and Deep Learning methods are now offering new possibilities from traditional statistical analysis to observe and take advantage of hidden patterns in brain networks.

1.1.2 Deep Learning

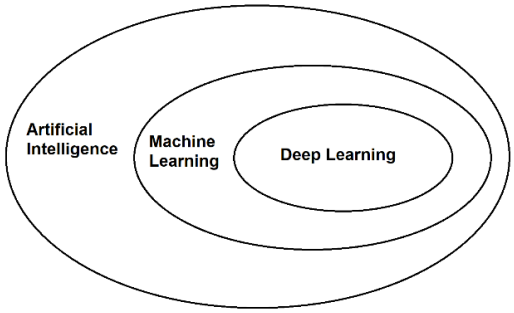


Figure 1.5: Artificial Intelligence, Machine Learning and Deep Learning

Artificial Intelligence

Artificial intelligence (AI) was born in the 1950's, a few years after the development of the very first computer, when pioneers of computing started to ask whether a computer could be made to "think" by itself. Indeed, in a 1950 landmark paper by Alan Turing (Turing 1950), where the Turing test was introduced, as well as key ideas that would come to shape AI - Could a computer go beyond what it is programmed for? This question opened doors to new programming paradigms and was at the origin of what is today called Machine Learning – an application of AI that provides the ability for a computer to learn patterns of various types without being explicitly programmed. (Francois 2017)

Machine Learning

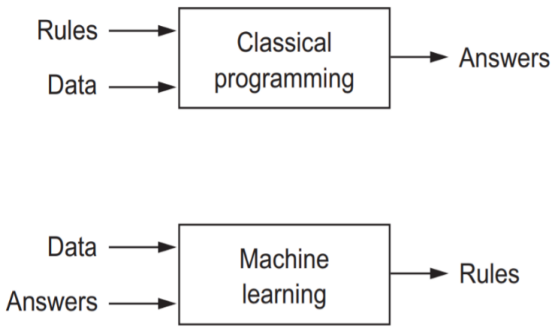


Figure 1.6: Difference between Classical Programming and Machine Learning (Francois 2017)

In opposite to the traditional programming paradigms where a set of rules are explicitly programmed, Machine Learning is trained and learns the rules by experience. Two main types of Machine Learning algorithms are to be distinguished: Supervised machine learning where the algorithm is given a set of input data points, as well as their corresponding outputs. The algorithm then learns a meaningful representation of the data which gets it as close as possible to the desired output. There are many poplar supervised ML algorithms such as decision trees, random forests, Support Vector Machines... In Supervised Learning, the corresponding outputs can either be discrete (i.e. set of classes), in which case the algorithm does a classification task; or continuous where the algorithm does a regression task 1.7. The other ML algorithm is unsupervised; the algorithm is only given a set of input data points without their corresponding output it then attempts to make sense of it by extracting features or patterns in the data on its own.

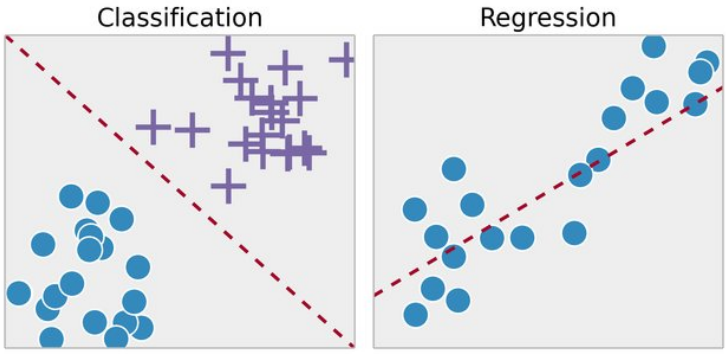


Figure 1.7: Classification and Regression (Ngoma 2017)

Deep Learning

Deep Learning is a subcategory of Machine Learning which tries to overcome some of its limitations. Indeed, for many years, ML algorithms struggles to process natural data in their raw form, careful data engineering and domain expertise was needed to obtain satisfying results on complex pattern recognition tasks (e.g. Image recognition). Deep Learning addresses the same ML issues with similar methods but putting an emphasis on learning with successive layers of increasingly meaningful representation (Francois 2017). More layers in a model (i.e. a deeper model) hence have the ability to learn more abstract representation of the original data. Hence, with enough of such transformations, increasingly more complex functions can be learned. Today, the most commonly used Deep Learning algorithm is a Neural Network (NN), an algorithm developed as a reference to biologic Neural Networks. A NN is built as shown in figure 1.8.

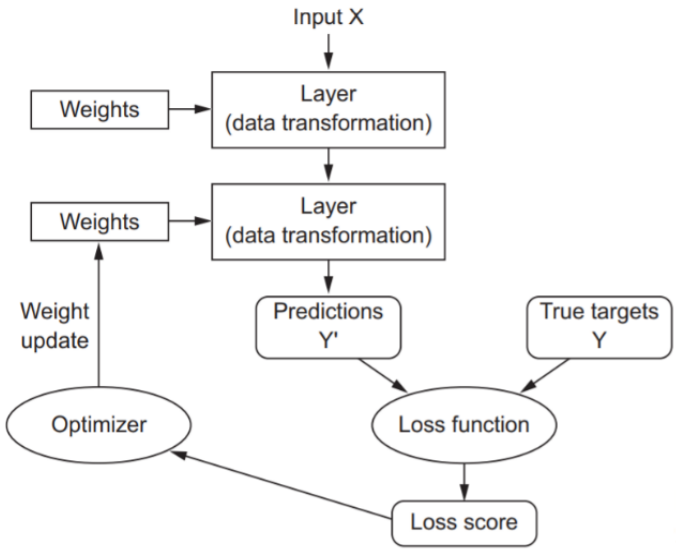


Figure 1.8: Principle of a Neural Network (Francois 2017)

When building a Neural Network, the objective is to find the best combination of weights to perfectly map the input data to the corresponding labels. This combination of weights is optimized in several iterations during training. As shown in the figure above, the Input vector passes through the various layers and (at first random) weights of layers; the forward pass being done, a prediction Y' is obtained and compared to a true expected value Y with the help of a loss function (also called "objective function"). The loss score represents how close the predicted Y' is from desired Y (e.g. mean absolute error). Based on this loss score, a backward pass is done to update the weights accordingly. This process is done at each iteration. The algorithm in charge of updating the weight is called Backpropagation and was developed by Geoffrey Hinton in 1986. (Hinton et al. 1986). Backpropagation computes the gradient of the objective function with respect to the weights through a chain rule of derivatives – this allows to update the weights of each layer depending on their contribution to the final loss score, thus allowing a better combination of weights for the next iteration.

Although this was understood as early as the 1980's, it is only with the drastic increase in computational power and the rise of modern GPU's that such computationally expensive algorithms could be put in practice. Deep Learning has achieved impressive performances in tasks such as image classification, speech recognition, machine translation etc... It is today widely used for many different applications, including healthcare and specifically Neuroscience.

Deep Learning in Neuroscience

Today, Deep Learning is having a large impact on many different aspects of healthcare such as diagnosis of diseases with visual recognition tasks from different imaging data, or segmentation tasks (Ronneberger et al. 2015). It is also having great impact in the field of Neuroscience and various of its sub fields, including Brain Connectivity. The large number of parameters and variability in Brain Connectivity data makes it very difficult for the naked eye to spot patterns in structural and functional connectivity; basic statistical analysis is also not always able to makes sense out of abstract behaviour in rich data as complex as a connectome. Deep Learning methods, having been designed to address such problems, thus appear to be of great potential to improve scientific understanding of the inner functioning of the brain.

1.2 Objectives

Deep Learning is, since 2015 with the publication of (Ronneberger et al. 2015), currently greatly impacting Healthcare technologies. It is indeed particularly useful for early diagnosis of diseases that are particularly challenging to spot. This is of crucial importance as it allows earlier treatment and hence allows longer life span, better care etc... This is the case in Neurodevelopment, where many disorders develop from a very

early age and are likely to irreversible consequences. Deep Learning can thus play a key role in predicting various neurodevelopmental outcomes at an early stage and minimize risks of developping disorders. An important issue with regard to this predictions is to understand how are Deep Learning algorithms able to correctly predict, otherwise, why should they be trusted? This work will thus attempt to predict various important neurodevelopmental outcomes with deep learning methods, and try to interpret the models' decisions with state of the art deep learning interpretation methods.

1.3 State of the Art

There has been to date various studies using Deep Learning for various tasks in Brain Connectivity and Neurodevelopment. In Neurodevelopment, an important task has been to use Deep Learning to predict clinical neurodevelopmental outcomes from Brain Networks.

BrainNetCNN (Kawahara et al. 2017) is a Convolutional Neural Network composed of edge to edge, edge to node, and node to graph convolution filters on Structural Connectivity to predict post menstrual age at scan (PMA) and cognitive performance. This model allows consideration of topological locality of structural brain networks - the connectome is thus treated as an image and not taken as a vector of features as in other publications dealing with structural connectome in Deep Learning (Munsell et al. 2015, Girault et al. 2019).

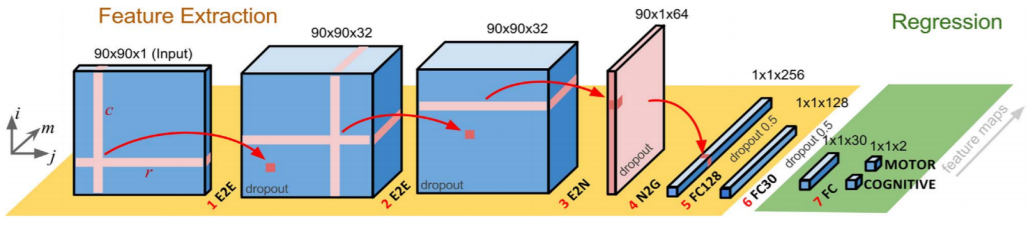


Figure 1.9: (Kawahara et al. 2017) Neural Network Architecture

Each of the three-layer (edge to edge, edge to node, node to graph) type is built around one or more convolutional filter and takes all feature maps from previous layer as input and outputs a distinct feature map for the next layer. It has been built to predict for neurodevelopmental outcomes such as cognitive and motor scores (assessed at 18 months of age) and prediction of age at scan. The data used for training the model is made of 115 infants born pre term and scanned between 27 and 46 weeks (adding up to 168 scans as some infants were scanned twice). Each network is represented as 90x90 symmetric adjacency matrix. The cognitive and neuromotor function were assessed using Bayley Scales of Infant and Toddler Development (Bayley-III) (Albers & Grieve 2007). They achieve prediction with mean absolute error (MAE) of 2.17 weeks (or 11% of age range) on prediction of age at scan. They also achieve prediction of Cognitive outcome with MAE of 10.6 points on Motor score and 10.4 points on Cognitive score.

Another recent study (Girault et al. 2019) similarly focused on using structural connectome at birth to predict cognitive abilities at age 2 with Neural Networks. The Cognitive assessment used was the Mullen score (Mullen et al. 1995) , which is slightly different to more commonly used Bayley III Their data was composed of 115 infants born between 30 and 42 weeks and scanned from 36 to 46 weeks. Due to difficulties in predicting directly cognitive score, they used a twostep approach: Firstly, classification Neural Network was trained from term babies using K fold cross validation to predict from a given structural connectome if the cognitive score was above median (AM) or below median (BM). The Neural Network was a dense fully connected Network, with one input layer of 3003 nodes (vectorized structural connectome), five hidden layers, five activation layers, 3 dropout layers and one output layer with two nodes, each outputting a class (AM or BM) probability. The second task consisted in using the classification probability to fit two different linear regression models (one for AM

classified infants and one for BM classified infants) and predict the actual cognitive score of the infant. On the classification task, they achieved an accuracy of 89.5% on term babies, and 83% on preterm babies. On the linear regression, they obtained results that were highly correlated with the actual cognitive score: r = 0.98 for term babies and r = 0.96 for preterm babies.

All these Deep Learning studies, although important and precise, are of great difficulty to understand (how precisely do they classify/predict). This is a common problem in all Deep Learning models; this is why a new field of study called interpretability has emerged to answer such concerns. Several studies have been led to date, and various methods were proposed, mainly looking at image classifiers using Convolutional Neural Networks(Lundberg & Lee 2017, Bach et al. 2015). A recent study introduced Layer-wise Relevance propagation (LRP) (Bach et al. 2015) – a backward propagation technique applicable in various computer vision applications. This method assigns a relevance score to each feature (or voxel if dealing with images) by iteratively propagating through each layer's output to its predecessor until the input layer is reached. This allows to create "Relevance maps" where visualisation of the features that were most important during regression/classification task can be highlighted. The relevance map is different for each input, meaning that activation is different across each data sample.

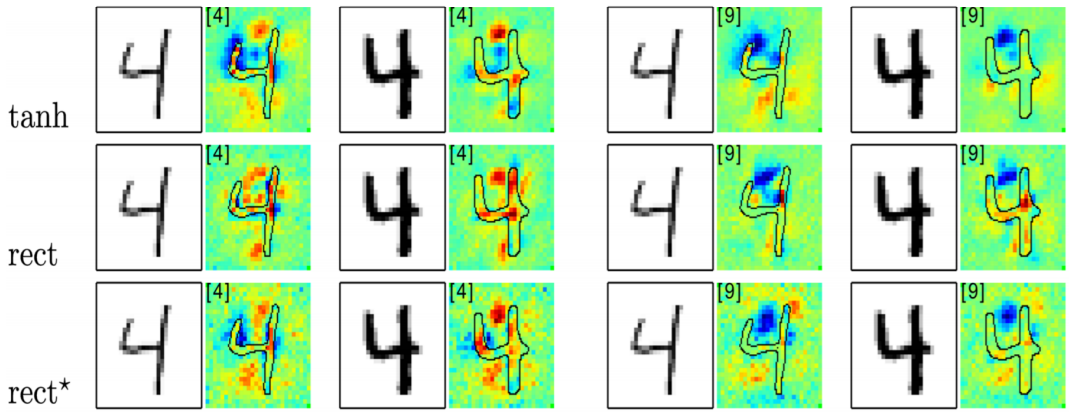


Figure 1.10: Relevance Map of NN classification between number 4 and number 9. Adapted from (Bach et al. 2015)

Figure 1.10 shows relevance map of classification task on MNIST data set (popular data set in deep learning with images of hand-written numbers). The left-hand side shows the relevance map for the image of a four on actually being a four (propagating through the output node corresponding to number 4 probability). The right-hand side figure shows the relevance map for the image of a four on being a nine (propagating through the output node corresponding to number 4 probability). Positive relevance is shown in red and negative relevance is shown is blue. It can be seen that on the left-hand side figure, the space in between the two bars is systematically shown in red, which increases the probability of it being a four. On the right-hand side, the same space between the two bars of the four is shown in blue, meaning it decreases the probability of it being a nine. This is coherent as the main difference between the number 9 and 4 is this specific gap between the two bars.

The work presented in this report will attempt to use various Deep Learning methods to evaluate all these studies using the dHCP data set. We propose a fully connected Neural Network that outperforms BrainNetCNN in prediction of age at scan (MAE of

0.7 weeks). We also propose a fully connected Neural Network which predicts Age at Birth of babies with MAE of 1.1 weeks, which has (to our knowledge) never been done in the literature. We also build a term/preterm classifier from Structural Connectivity data using a fully connected Neural Networks where we reach accuracy of 90% on classification. We interpret this model using Layer-wise relevance propagation (Bach et al 2015a) and are hence able to underline the connections that had the most impact in the classification task. We also attempt to replicate the results of (Girault et al. 2019) using dHCP data set and classification of above/below median cognitive score and prediction of exact cognitive score from Structural and Functional Connectivity data.

Chapter 2

Methods

2.1 Data

2.1.1 Scanning infants

Infants scanned

The developing Human Connectome project to date has more than 700 scans of infants at prenatal or postnatal stage; our study focuses exclusively on scans executed at a postnatal stage. Structural and functional connectivity information (acquired with dMRI and fMRI) was available for 425 infant scans to conduct this study. The age range (age at birth – Gestational Age) for all infants was from 23 weeks of gestation to 42 weeks. We count 314 term babies (Gestational Age $\geq$ 37 weeks) and 111 pre term babies (Gestational Age $<$ 37 weeks). Infants were scanned at different Post Menstrual Age (age since gestation) ranging from 35 to 45 weeks.

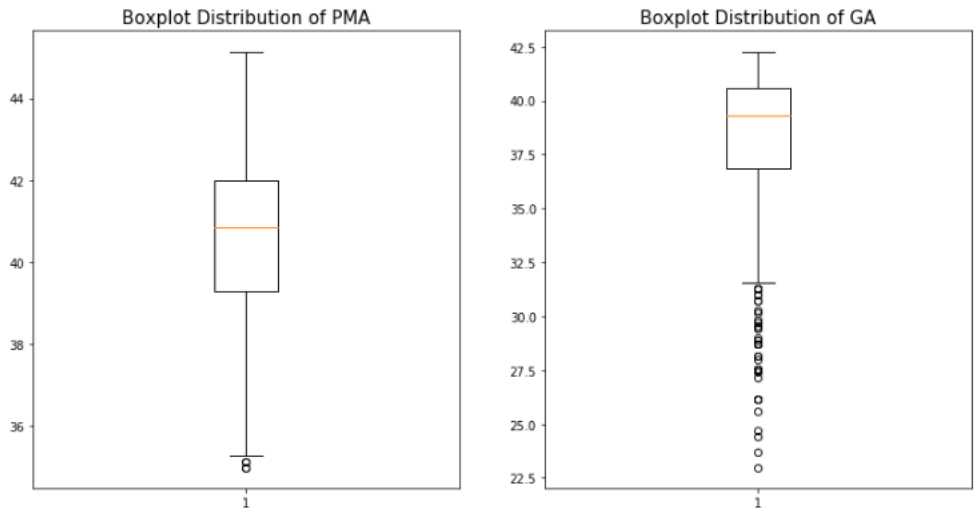


Figure 2.1: Boxplot distribution GA and PMA of available data set

Ensuring safe and comfortable acquisition for infants

As executing MRI scanning in neonates is very challenging due to motion and fragility of the infants, the developing Human Connectome project has designed a full neonatal brain imaging system around 5 main important elements: Head Coil, positioning, Immobilization, Gradient noise and transport (Hughes et al. 2017). The head coil, to maximise the Signal to noise ratio (SNR), needs to be fitted as close as possible to the head while having as many channels as possible. After estimating the maximum diametric of the oldest infant in the targeted group, the chosen size of head coil was 155 mm from anterior to posterior, 130mm from right to left and 140mm from inferior to superior. The number of channels was set at 32. The designed positioning device, which is a lightweight protective "shell" allowed preparation of the neonates away from the scanner and held infants in secure position. It also ensured easy access in case of emergency. This shell was positioned on an MRI safe trolley to ease transportation. The immobilization of the infant in the shell was done through bead filled pads to insure comfort. As they are positioned in over the ears, they serve as acoustic protections.

To avoid sudden sound changes which might wake up the infant, the MRI software was modified in order to linearly increase the noise from 0 to operating point.

MRI acquisition

All scans were collected in St Thomas hospital London on a Philips 3T scanner. For the acquisition of the structural MRI, T1 weighted images were acquired using an inverse recovery Turbo Spin Echo sequence with resolution repetition time (TR) = 4.8s and echo time (TE) = 8.7ms. T2 images were acquired using Turbo spin echo sequence using parameters TR = 12s and TE = 156ms, SENSE factor 2.11 (axial) and 2.54 (sagittal) with overlapping slices (resolution = 0.8 × 0.8 × 1.6mm). Super resolution methods (Kuklisova-Murgasova et al. 2012) as well as motion correction methods (Cordero-Grande et al. 2018) were combined to maximise precision and resolution of images. Diffusion weighted imaging was acquired with 300 directions with parameters: TR = 3.8s, TE = 90ms, sensitivity encoding E: 1.2, resolution = 1.5 × 1.5 × 1.5mm, diffusion gradient encoding: b=0 s/mm (n=20), b=400 s/mm (n=64), b=1000 s/mm (n=88), b=2600 s/mm (n=128) with interleaved phase encoding.(Hutter et al. 2018) BOLD Functional MRI scans were acquired during resting state, with parameters TR = 392ms and TE = 38ms, flip angle = 34, spatial resolution = 2.15mm isotropic with 45 slices. (Ciarrusta et al. 2019)

Structural and Functional connectome construction

Tissue segmentation of T1 and T2 volumes and their parcellation into 90 cortical and subcortical regions (Shi et al. 2011) was performed using neonatal specific segmentation (Makropoulos et al. 2014) and template (Schuh et al. 2018). Diffusion MRI was distortion and motion corrected and represented in a compact q-space (Christiaens et al. 2019) where a probabilistic 10M streamlines tractography with biologically

accurate weights (SIFT2) was generated (Smith et al. 2015, Jeurissen et al. 2014). The Structural connectivity network of each infant was constructed by calculating the SIFT2-weighted sum of streamlines connecting each pair of regions. The Functional connectivity network of each infant was generated by performing partial correlation between each pair of regions (with the other regions as covariates). For simplicity, only results at 30% of SC network density are used for this work(Batalle et al. 2017).

2.1.2 Structural and Functional Connectomes

Data from dMRI and fMRI scans are presented as adjacency matrices. An adjacency matrix is a convenient way to represent a network in a matrix form. A network or graph G = (X, U) consists of:

1. A finite set $X = \{x_1, x_2, .., x_n\}$, whose elements are called nodes. They represent the fundamental elements of the system, such as people in social networks, or in our case, brain regions.

2. A subset U of the Cartesian product $X \times X$, the elements of which are called edges. This represents the connections between certain pairs of nodes. Edges can be binary (simply showing the existence of a connection) or weighted (showing the existence and strength of connection). Edges can also be in a directed (e.g. node A is connected to node B, but node B is not connected to node A) or undirected form.

A wide variety of information and analysis can be performed on graphs/networks for many different purposes and in many different contexts. Here, we will mainly be focusing on connections weights between brain regions. This is why the adjacency matrix, which clearly shows the existence of connections, their weights and directions,

is the most convenient way to represent the information for our purpose. The figure below 2.2 simply shows how an adjacency matrix is built.

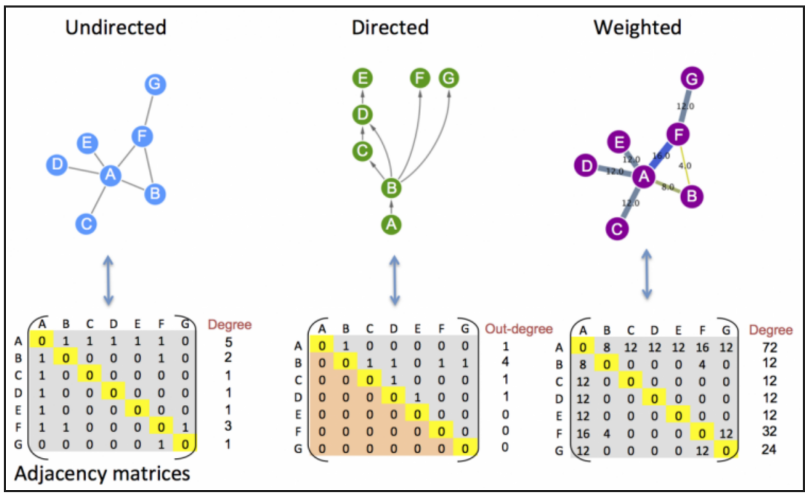


Figure 2.2: From Network to Adjacency matrix (*Graph theory: adjacency matrices* 2016)

The structural connectivity matrix aims to show the strength of the anatomical connections between different brain regions. It is hence represented by a weighted adjacency matrix. The functional connectivity matrix aims to show regional interaction through activation correlations. As the correlations vary between -1 and 1, it is also represented by a weighted adjacency matrix. Although actual brain connections are directed, the current imaging technologies do not allow to precisely identify connection directions; we thus consider the connection strengths to be the same in both directions. The adjacency matrix for structural connectome and functional connectome are hence presented as an undirected graphs, which yields to symmetric adjacency matrices.

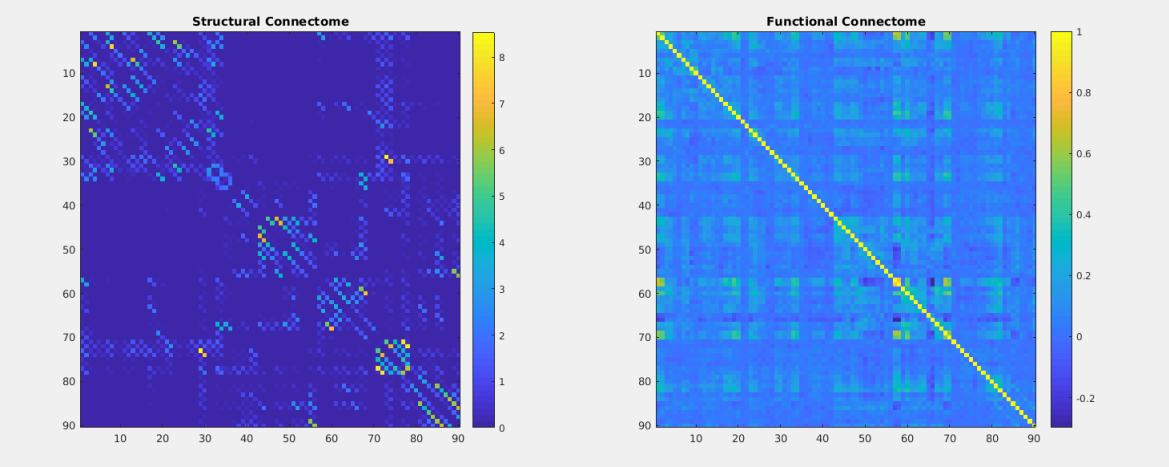


Figure 2.3: Typical Structural and Functional connectivity adjacency matrices

Both the structural and functional connectivity matrices which are used in this project are built around 90 different nodes, representing various brain regions. The figure below shows the name of each and location of the 90 brain regions.

'1 Precentral_L 2001'	'21 Olfactory_L 2501'	'41 Amygdala_L 4201'	'61 Parietal_Inf_L 6201'	'81 Temporal_Sup_L 8111'
'2 Precentral_R 2002'	'22 Olfactory_R 2502'	'42 Amygdala_R 4202'	'62 Parietal_Inf_R 6202'	'82 Temporal_Sup_R 8112'
'3 Frontal_Sup_L 2101'	'23 Frontal_Sup_Medial_L 2601'	'43 Calcarine_L 5001'	'63 SupraMarginal_L 6211'	'83 Temporal_Pole_Sup_L 8121'
'4 Frontal_Sup_R 2102'	'24 Frontal_Sup_Medial_R 2602'	'44 Calcarine_R 5002'	'64 SupraMarginal_R 6212'	'84 Temporal_Pole_Sup_R 8122'
'5 Frontal_Sup_Orb_L 2111'	'25 Frontal_Med_Orb_L 2611'	'45 Cuneus_L 5011'	'65 Angular_L 6221'	'85 Temporal_Mid_L 8201'
'6 Frontal_Sup_Orb_R 2112'	'26 Frontal_Med_Orb_R 2612'	'46 Cuneus_R 5012'	'66 Angular_R 6222'	'86 Temporal_Mid_R 8202'
'7 Frontal_Mid_L 2201'	'27 Rectus_L 2701'	'47 Lingual_L 5021'	'67 Precuneus_L 6301'	'87 Temporal_Pole_Mid_L 8211'
'8 Frontal_Mid_R 2202'	'28 Rectus_R 2702'	'48 Lingual_R 5022'	'68 Precuneus_R 6302'	'88 Temporal_Pole_Mid_R 8212'
'9 Frontal_Mid_Orb_L 2211'	'29 Insula_L 3001'	'49 Occipital_Sup_L 5101'	'69 Paracentral_Lobule_L 6401'	'89 Temporal_Inf_L 8301'
'10 Frontal_Mid_Orb_R 2212'	'30 Insula_R 3002'	'50 Occipital_Sup_R 5102'	'70 Paracentral_Lobule_R 6402'	'90 Temporal_Inf_R 8302'
'11 Frontal_Inf_Oper_L 2301'	'31 Cingulum_Ant_L 4001'	'51 Occipital_Mid_L 5201'	'71 Caudate_L 7001'	
'12 Frontal_Inf_Oper_R 2302'	'32 Cingulum_Ant_R 4002'	'52 Occipital_Mid_R 5202'	'72 Caudate_R 7002'	
'13 Frontal_Inf_Tri_L 2311'	'33 Cingulum_Mid_L 4011'	'53 Occipital_Inf_L 5301'	'73 Putamen_L 7011'	
'14 Frontal_Inf_Tri_R 2312'	'34 Cingulum_Mid_R 4012'	'54 Occipital_Inf_R 5302'	'74 Putamen_R 7012'	
'15 Frontal_Inf_Orb_L 2321'	'35 Cingulum_Post_L 4021'	'55 Fusiform_L 5401'	'75 Pallidum_L 7021'	
'16 Frontal_Inf_Orb_R 2322'	'36 Cingulum_Post_R 4022'	'56 Fusiform_R 5402'	'76 Pallidum_R 7022'	
'17 Rolandic_Oper_L 2331'	'37 Hippocampus_L 4101'	'57 Postcentral_L 6001'	'77 Thalamus_L 7101'	
'18 Rolandic_Oper_R 2332'	'38 Hippocampus_R 4102'	'58 Postcentral_R 6002'	'78 Thalamus_R 7102'	
'19 Supp_Motor_Area_L 2401'	'39 ParaHippocampal_L 4111'	'59 Parietal_Sup_L 6101'	'79 Heschl_L 8101'	
'20 Supp_Motor_Area_R 2402'	'40 ParaHippocampal_R 4112'	'60 Parietal_Sup_R 6102'	'80 Heschl_R 8102'	

Figure 2.4: Numbered and labeled brain regions used in this project

2.1.3 Neurodevelopmental Assessment

Cognitive evaluation was done for 285 infants at age 18 months using the Bayley Scale of Infant Toddler Development, third edition (Bayley III) (Albers & Grieve 2007). Bayley III is a revision of the frequently used Bayley scale, which was originally developed by psychologist Nancy Bayley. It is today the most widely used developmental

assessment tool for infants in the world. It allows assessment of developmental status for children aged 1-42 months. The Bayley III has three main subtests: Cognitive scale (mainly focusing on attention to surrounding actions), Language scale (testing for understanding and expression of Language) and Motor scale (assessing motor skills such as walking, climbing, grasping etc. . .). The three subsets scores were available for all 285 infants. Assessment of Bayley III usually takes 45-60 minutes and is done by an experimented psychologist with the presence of a parent or caregiver (known to the infant). The Bayley raw scores were converted into a scaled score where the median score was 100 and the standard deviation was 15. The following figure presents the distribution of the three different assessments for all 285 babies.

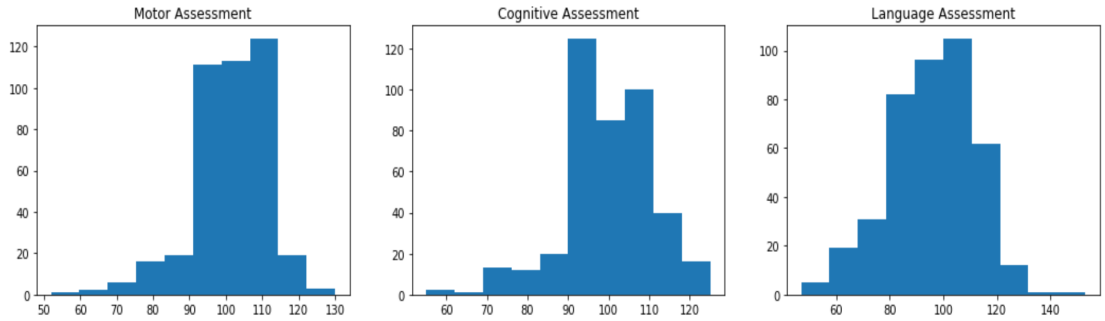


Figure 2.5: Motor, cognitive and language Bayley Score distribution for dHCP data set

2.2 Deep Learning Software and basic methods

2.2.1 Computational tools used

For this project, various computational tools have been used. All Deep Learning algorithms and data pre-processing were implemented using Python 3.6. Python is a high level, interpreted, general purpose programming language created in 1991. Python is

widely used for scientific programming due to the third-party library Numpy, which supports large multi-dimensional arrays, matrices and tensors operations with high efficiency. Today, Python is extremely popular within the Machine Learning (ML) community due the availability of various libraries that make implementation of ML algorithms extremely convenient. For this project, the Deep Learning Framework Keras (*Keras: The Python Deep Learning library* n.d.) has been used for implementation of all Deep Learning algorithms. Keras is a high-level Application Programming Interface (API) used for implementation of Neural Networks.

2.2.2 Deep Learning Methods

he vast majority of deep learning algorithms are implemented are Neural Networks (NN). Many variations of Neural Networks have emerged as suited to different data and different tasks. For example, convolutional neural networks are state of the art for semantic segmentation of images, whereas recurrent neural networks are best suited to time series data. We will mainly be using the less specialised, but still powerful, Dense Fully-Connected Neural Network (DNN).

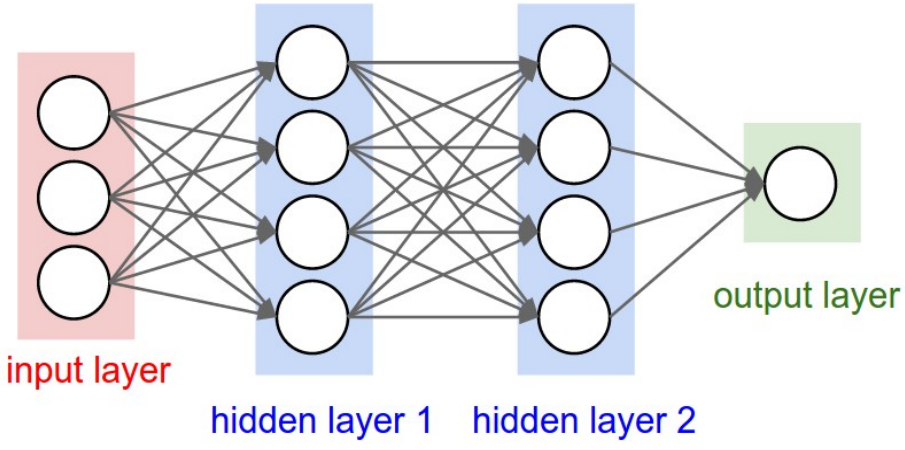


Figure 2.6: Basic Fully connected Neural Network Architecture

A DNN has, as with all NNs, a sequence of neurons arranged into layers. A neuron is a biologically inspired element, which - like an actual neuron - is connected to other neurons in a weighted manner. The key component of a DNN is full connectivity of the neurons of neighbouring layers - every neuron in one is connected to every other neuron in the next layer (see figure 2.6). This fully connectedness means that the DNN is powerful and versatile, but with the drawback of a large number of parameters to be learned. Each neuron in each layer thus carries a certain weight, which is updated at each iteration (or epochs) upon training through back-propagation (Hinton et al. 1986). The goal of "training" is to find the optimal mathematical configuration of weights to turn the input features into the desired output. Three main type of layers are needed for this type of Neural Network: an output layer which carries the desired solution, an input layer which carries all the available features to fit for the desired solution, and a number of hidden layers which are in between the input and output layer. Each layer is usually followed by an activation layer, which aims at converting an input signal from a node to an output signal with a specific non linear function.

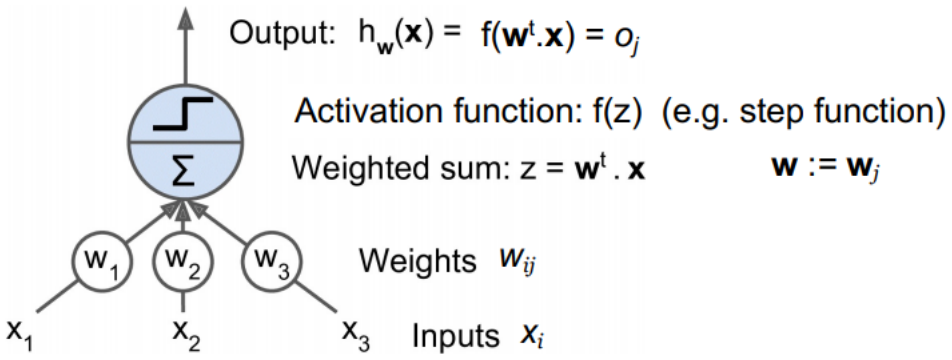


Figure 2.7: Activation function principle

As the activation functions are nonlinear, they introduce non linearity to the model (without them, Neural Networks would simply be linear regressions). There exist many

different types of Activation functions which are used in various contexts, such as Sign function, ReLu, Heaviside function, softmax etc...

For this project, we will mainly use the activation function Rectified Linear Unit (ReLU) (eq 2.1) for regression and classification tasks. The function returns 0 if it receives any negative input, but for any positive value x it returns that value back. Although very simple, it allows the model to account for non-linearities. It tends to perform very well on a wide variety of tasks. Another activation function was systematically used for classification task on the last layer: softmax function. Softmax (eq. 2.2) forces the outputs to sum to 1 so that they represent probability distribution of each classes. It indeed applies standard exponential function to each element of input vector z_i and normalize these values by dividing by the sum of all these exponential (thereby ensuring that the sum of all component of output vector is 1). The formulas and graphs for these two activations functions are shown below:

$$ReLu(z) = \begin{cases} 0 & \text{if z } < 0 \\ z & \text{if z } \geq 0 \end{cases} \tag{2.1}$$

$$\text{softmax}(z) = \frac{e^{z_i}}{\sum_i e^{z_i}} \tag{2.2}$$

At each training iteration, the output of the NN is evaluated against some "true" value according to a predefined loss function. This is crucial, as it allows the weights to be adjusted by back propagation by minimizing the loss score. There exist many different loss functions, all are useful in specific contexts. For regression tasks, mean square error (MSE) is most commonly used. It simply consists in the square of the

difference between true and predict output (eq. 2.3). Minimizing the MSE is equivalent to minimizing the difference between true and predicted output. For classification tasks, where probabilities are between 0 and 1, a commonly used loss function is the cross entropy function (eq.2.4). The categorical cross entropy calculates for class 1 to C the product of ground truth t_i (correct class for given sample) and the logarithm of predicted class probability $f(s_i)$. The objective is to minimize the entropy, thus minimizing the error. The closer $f(s_i)$ is from t_i, the lower the entropy will be.

$$\mathrm{M}SE = \frac{1}{n}\sum_{i=1}^{n}(y_i - \widetilde{y_i}) \tag{2.3}$$

$$CE = -\sum_{i}^{C} t_i log(f(s)_i)) \tag{2.4}$$

All Deep Learning algorithms need training data that will be used to fit the model. The model is hence learning its parameters from this data. Testing data is also necessary to produce an unbiased evaluation of the final mode (fit on training set) and to verify that the model generalized well from training and did not over fit. Over fitting is a common problem in Deep Learning where the a model is too closely fitted to a limited set of data points (training data). To prevent over fitting, a commonly used method is to use a third sub sample from the data, called validation data, which is held back from training but is used to test the model while it is being trained. When plotting training against validation accuracy, it is possible to spot when the model starts over fitting. As shown in the 2.8, the model starts over fitting when the training accuracy is improving, but the validation accuracy is getting worse. This means that the model is too closely fitted to the training data.

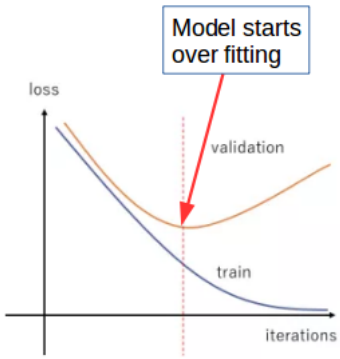


Figure 2.8: Validation and training loss per epochs - visualizing over fitting

To obtain the best possible performance from a model, there are a number of parameters that can be modified to best suit the task. One key element in deep learning is to process the input and output data in a way where the algorithm will be able to quickly see patterns. For example, it is always more difficult for Neural Network to deal with large numbers, this is why it is common practice to normalize all input and output features between 0 and 1. Another important element in building a well performing Neural Network is its architecture – the number of hidden layers and neuron per layer is important. For example, deeper (more hidden layers) Neural Networks reach higher level of abstractions, which is a useful for complex task but maybe counterproductive in simpler tasks (Poggio et al. 2017). The number of neurons per layer also has great importance as it increases or decreases the number of parameters (weights) to tune during training – a high number of parameters are difficult to tune with small data sets. There is no exact rule as to how to precisely choose the Neural Network architecture, choice is made by testing. Other parameters such as the learning rate are also important for performance of the Neural Network. The learning rate is the step size in the updating the weights during back propagation (which is essentially a gradient descent). For this project, learning rates are always chosen through grid search.

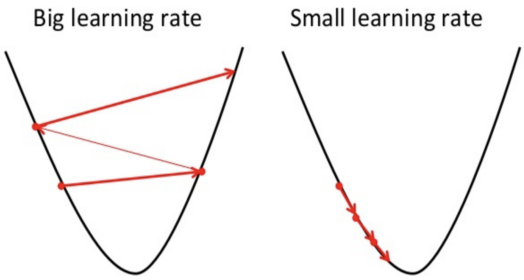


Figure 2.9: Big Learning rate vs small learning rate

There are many additional methods that allow to improve the performance of implemented Neural Networks, such as Dropout (Srivastava et al. 2014) which is a tool that prevents over fitting. Dropout consists in randomly dropping units (neurons) along with their connections on a given layer. This prevents neurons from co-adapting too much. Batch Normalization (Ioffe & Szegedy 2015) is another popular method which consists in normalizing the input layer by adjusting the activations. Both these additional methods have been used in building the various models to improve performance.

2.3 Section 1: Age of infants from Structural Connectivity: Towards showing Delay in Preterm Brain Networks

For this section, we implement various Deep Learning algorithms to predict for gestational age at birth (GA) of the infant or post menstrual age at scan (PMA) of infant from structural and functional connectivity. The main objective in doing so is to understand the various developmental process in brain connectivity, and to show that significant differences in brain connectivity can be identified.

2.3.1 Task 1: Predicting PMA from structural connectome

The first attempted task consists in predicting PMA from structural connectome using a Neural Network. Contrary to BrainNet CNN (Kawahara et al. 2017), which predicts PMA from structural connectivity using convolutional Neural Networks (reaching MAE of 2.17 weeks), we propose a dense fully connected Neural Network. Indeed, Convolutional Neural Networks (LeCun et al. 1998) are particularly useful for data known to have local correlation, for example in images where objects to be segmented are, by their very nature, locally confined into regions. Since the adjacency matrices are not actually reflective of brain region locality, we choose to instead use DNNs in our investigation. Before building the Neural Network, pre-processing the input features – the Structural and Functional connectivity adjacency matrices - is necessary. As mentioned in section 2.1.2, the adjacency matrices are symmetric, meaning that all the information is repeated twice – only the lower triangle will thus be considered (see figure below). The lower triangle is then extracted and reshaped into a 1D vector of dimension 4005, and is finally normalized between 0 and 1.

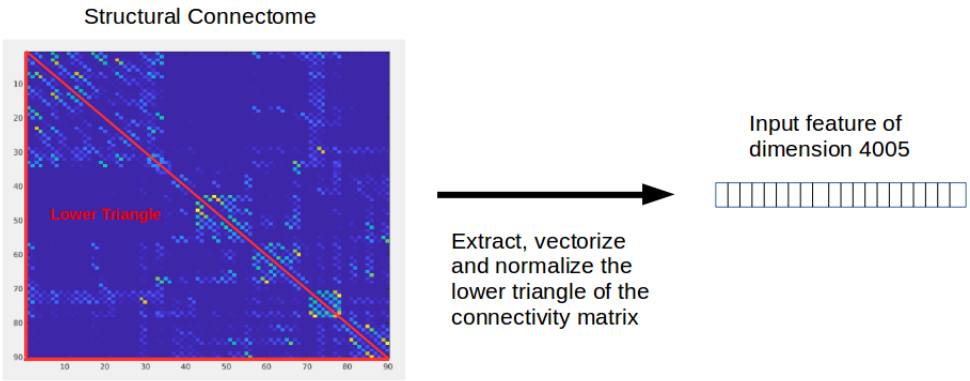


Figure 2.10: Processing of structural connectivity matrix

The architecture used for task 1 is presented in the above figure 2.10 . It consists of one input layer with the normalized vectorized structural connectome (4005

features), with 3 hidden layers, 4 activation layers with ReLU function, one dropout layer dropping 40% of connections and one output layer with one node, corresponding to the PMA associated with the structural connectome. The loss function used to adjust the weights is Mean Squared Error (MSE).Various other architectures, activation functions, dropout rates and loss function have been tried, this one performing best of all. The learning rate, chosen through grid search in increments of 0.005 from 0.08 to 0.0001, that reported highest accuracy was 0.004.

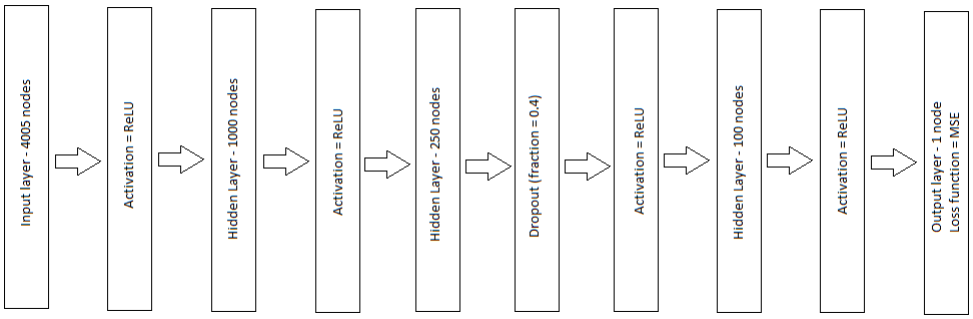


Figure 2.11: Architecture of Neural Network for prediction of PMA

As BrainNetCNN reported results for a preterm trained and tested model, we first implement our model on preterm infants only (training, testing and validation set preterm infants). We are thus considering the complete preterm cohort of 111 infants. Training was done on 70% of the preterm infants, testing on 20% and validation on 10%.

We also implement the model with the full term and preterm data set. The 425 structural connectomes available for training were separated into a training (70%), testing (15%) and validation (15%) subsamples. Training was thus done on the training data and was at each iteration evaluated with the validation tested. Reported performance of the model was evaluated with the testing data set. The train/test/validation splitting was done so that each sub sample had an equal proportion of term babies

(GA>37w), preterm babies (32w>GA >37w), and very preterm babies (GA<32w). Training was done through 80 epochs.

2.3.2 Task 2: Predicting GA from structural connectome

The second task consisted in prediction age at birth (GA) using Neural Networks from Structural connectome. Similarly, to Task 1, this was done using a DNN. The pre-processing of structural connectome was also done similarly to Task 1, however, we add to the input features the PMA. As shown in task 1, age at scan has a significant impact on the structural connectome, so adding it to the input feature to predict for age at birth strongly improves the performance. The PMA is normalized between 0 and 1 to be on the same scale as the features of the structural connectome. The architecture chosen for the Neural Network is presented in the figure below. It consists of one input layer with normalized vectorized structural connectome, plus the normalized PMA (4006 features), 4 hidden layers, 4 activation layers (using ReLU), one dropout layer (fraction = 0.5), 4 batch normalization layers and one output layer with one node corresponding to GA. As in task 1, the optimal learning rate was found through grid search: 0.001. Chosen loss function is MSE. The train/test/validation splitting was also done so that each sub sample had an equal proportion of term, preterm and very preterm babies. We use 70% for training, 15% for validation and 15% for testing. Training was done through 120 epochs.

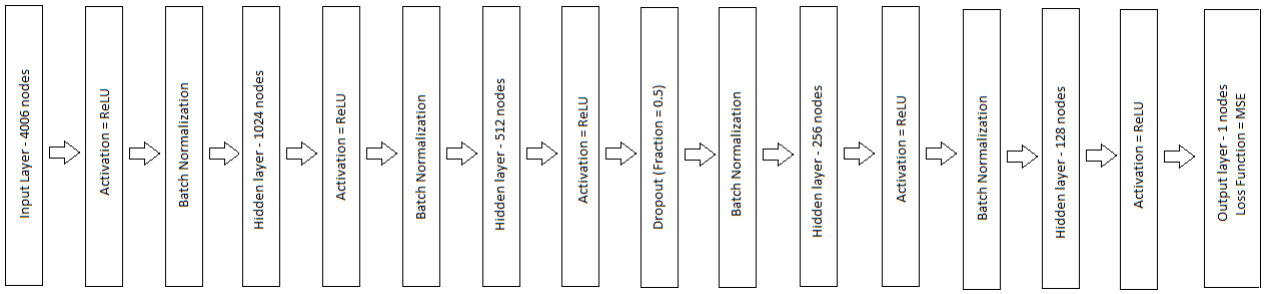


Figure 2.12: Architecture of Neural Network for prediction of GA

2.3.3 Task 3: Showing connectivity delay in structural connectome of preterm infants

The last task of this session consisted in explicitly showing a developmental delay in preterm babies compared to term babies after birth. A DNN is trained on term infants only to predict PMA from structural connectome (training and validation data were term babies). We test the model on both term and preterm babies. We hypothesized that upon testing, a significant difference would be noted between term and preterm infants, meaning that term infants develop differently than preterm babies. We first separate term babies from preterm babies. We only consider infants scanned after 37 weeks (this is to ensure that every studied infant has had 37 weeks of life); as a consequence, we have fewer infants (391 instead of 425). We also do a train/test/validation split on term babies only (70%, 20%, 10%). The model is then trained and validated with the previously defined training and validation data. It is tested separately on the testing sub sample of term babies, as well and on preterm babies. A linear regression between actual PMA and predicted PMA is executed on both term and preterm testing. The architecture and chosen parameters of the Neural Network is the same as in task 1.

2.4 Section 2: Cognitive score prediction from Structural Connectivity

For this section, we propose to fully implement the two-step method proposed in (Girault et al. 2019) with dHCP data. Two differences should be noted before implementing this method: they use a 78x78 connectome, which is different from the 90x90 connectome acquired within the dHCP. Also, the cognitive score used is based on Motor, Visual and Language assessment from Mullen test, which are combined into an Early Learning Composite (ELC). The cognitive score used by the dHCP is based on Bayley III, which is different from Mullen. We implement the method using Motor and Cognitive assessment as they show higher reliability than language assessment (Ranjitkar et al. 2018). Computing an average of the three scores as they did is not interesting in Bayley as the three score are not so highly correlated with each other - this would thus introduce considerable noise.

Replicating the methods presented in the paper as precisely as possible, we proceed as follows: A classification Neural Network is first built to fit for above or below median (AM or BM) cognitive score from structural connectome. The model is only trained on term infants. As done in section 1, only the lower triangle of the structural connectome is considered, which is then normalized and vectorized. The Neural network architecture we implement is very similar to the paper, and is thus built with one input layer, five hidden layers, five activation layers, three Dropout layers and one output layer with 2 neurons (see figure 2.13 for full model description). The last activation layer and the loss function are respectively softmax and cross entropy loss function. One output neuron outputs the above median (AM) probability and the other one outputs the below median (BM) probability. For instance, for a given connectome, the AM node could output probability 91%, the BM node would then output 9% (thus

summing to 1). The same activation function, dropout rates, learning rates and loss functions were used in our implementation

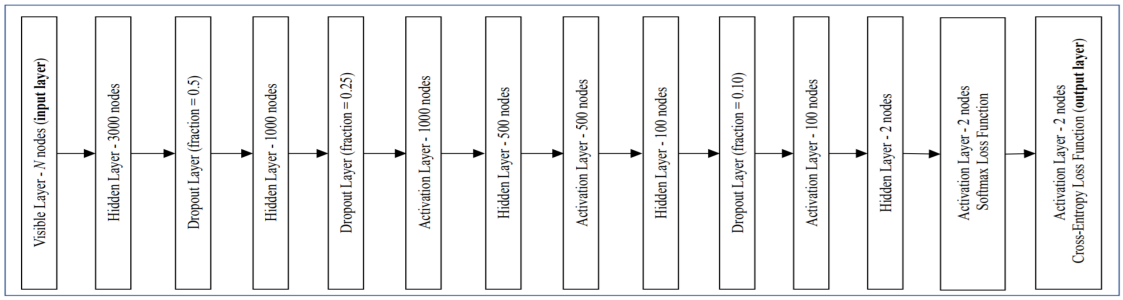


Figure 2.13: Classification Neural Network as implemented in (Girault et al. 2019)

The only difference between the architecture used in the paper and our implementation is that the input layer we implement is of size 4005 instead of 3003. This is due to the difference in connectome - the connectome used in (Girault et al) is of size 78x78 which yields to 3003 features whereas our connectome is 90*90 which yields to 4005 features. This Neural network is only trained on term babies. Training is done with K-fold cross validation (using 10 folds). K-fold cross validation is a widely used method in case of small training/testing data size. It consists in partitioning a sample of data into two separate subsets: one for training (80%) and one for testing (20%); this partitioning is done K times. We thus train and test 10 independant models on 10 different train/test split. For each K folds, the model was tested as follows: if for a given connectivity feature vector, the output response of the Neural Network classification was AM = 0.35 and BM = 0.65 then the infant is classified as BM. The model was thus trained and tested 10 different times, and the reported final accuracy of model is calculated by taking the average accuracy of all K models. The paper also proposes to test the term trained model on preterm babies, for which a voting system from the 10 different models is used – if 5 (or more) models out of the 10 models classify a given sample as AM, the infant is classified as AM (and vice versa).

The second task consists in using the classification probabilities generated from the Neural Network to perform a linear regression with the actual cognitive score. The classification probabilities used are averaged out from each of the 10 folds for term infants. For preterm infant, the classification probabilities are generated by averaging the output of each of the 10 models (similar to the voting system). Based on the output probabilities, the infants are separated into two distinct groups: AM classified and BM classified infants. For each group, we normalize the probabilities between 0 and 1 and correlate these probabilities with corresponding true cognitive score (linear regression). This was done separately for term and preterm infants.

2.5 Section 3: Neural network model interpretation

In this section, we attempt to gain insight into the way Neural Networks manage to differentiate term and preterm babies – this is trying to understand which connections and brain regions are allowing this differentiation. As interpreting regression is much more challenging, we propose to implement a simple classification DNN to classify between term and preterm babies from structural connectome, which will then be interpreted using Layer wise Relevance Propagation(LRP).

Pre-processing the structural connectome is done as in section 1 and 2. The architecture of the DNN consists in one input layer, 4 hidden layers, 1 dropout layer (fraction = 0.1), 5 activation layers and one output layer with two nodes (one for term probability and one for preterm probability). Optimal learning rate was 0.005 (found through grid search); loss function used is cross entropy (Janocha & Czarnecki 2017). As in section 1, the train/test/validation splitting was done so that each sub sample had an equal proportion of term, preterm and very preterm babies. We use all avail-

able data (425 infants). Splitting is done with 70% training, 15% validation and 15% testing. Training was done through 90 epochs.

The classification DNN being built, LRP (Bach et al. 2015) algorithm was implemented to perform interpretation of the classification. LRP is a backward propagation technique that assigns a relevance score to each feature by iteratively propagating backwards from each layer's output to its predecessor until the input layer is reached (See figure 2.14). This is done for each testing sample - it hence assigns different relevance for each sample. The LRP algorithm outputs for each sample the relevance of each feature. The relevance is either positive (positive contribution) or negative (negative contribution). In this section, the LRP algorithm used is implemented as in (Montavon et al. 2018).

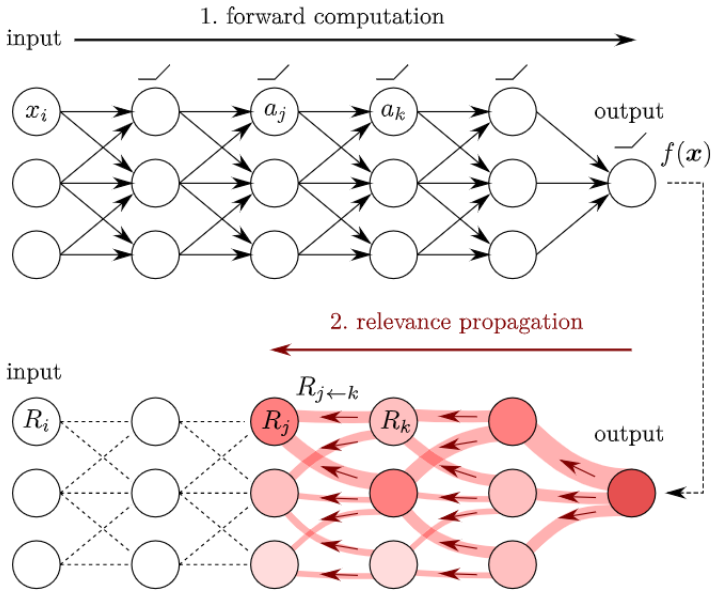


Figure 2.14: Diagram of the LRP procedure, adapted from (Montavon et al. 2018)

We apply the LRP to each testing sample, we thus obtain the relevance of all 4005 features for each testing sample. We then separate the term and preterm classified infants into two separate groups. On both groups, we compute the average feature

relevance on each feature. We thus obtain the 4005 feature. A $90x90$ matrix can thus be rebuilt for eaching testing sample, where each edge's weight corresponds to its relevance. The full process is clearly explained in the figure 2.15.

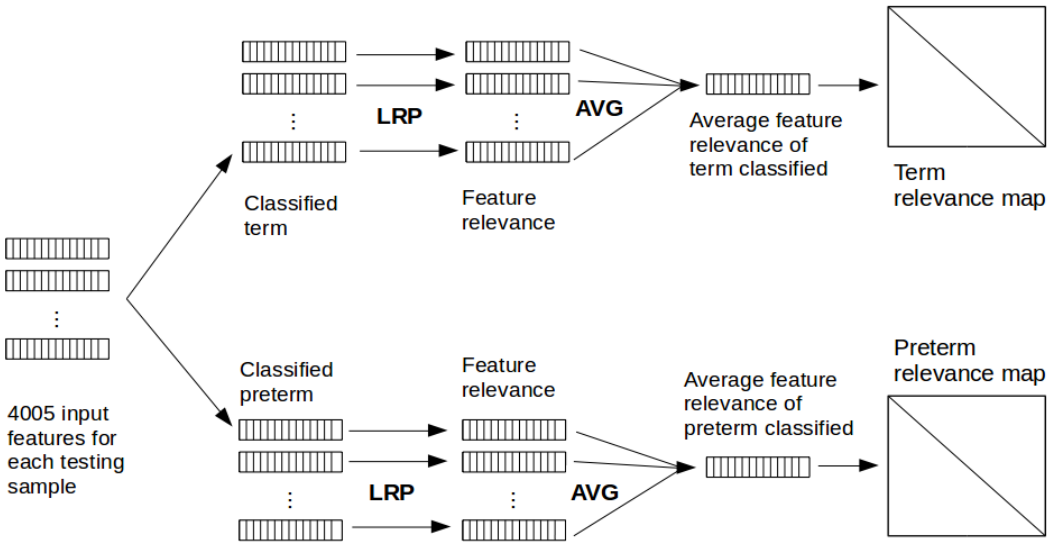


Figure 2.15: Relevance map construction process

We thus obtain a different relevance map for term infants and preterm infants. From then, we can infer the edges that are most characteristic of term infants and of preterm infants. Indeed, the most positively activated features in the term relevance map will be the most characteristic of term infants, and vice versa. This would allow to identify edges that are most important to correctly differentiate between term and preterm infants.

Chapter 3

Results

3.1 Section 1: Age of infants from Structural Connectivity: Towards showing Delay in Preterm Brain Networks

3.1.1 Task 1: Predicting PMA from structural connectome

A fully connected Neural Network was trained on preterm infants only to fit for PMA from structural connectome. We reach a mean absolute error of 0.84 weeks on prediction. The Pearson correlation between true and predicted reaches r=0.90 and p<0.001. Figure 3.1 summarizes all information for this model. Figure **??** shows the evolution of the loss as per training iteration (epoch), as well as a plot of true PMA against predicted PMA.

Training data	Testing data	Validation data	Features	Desired output	Training set MAE	Validation set MAE	Testing set MAE
75 pre term infants	25 pre term infants	11 pre term infants	Structural connectome	PMA	0.79 weeks	0.81 weeks	0.84 weeks

Figure 3.1: Task 1-a Model summary

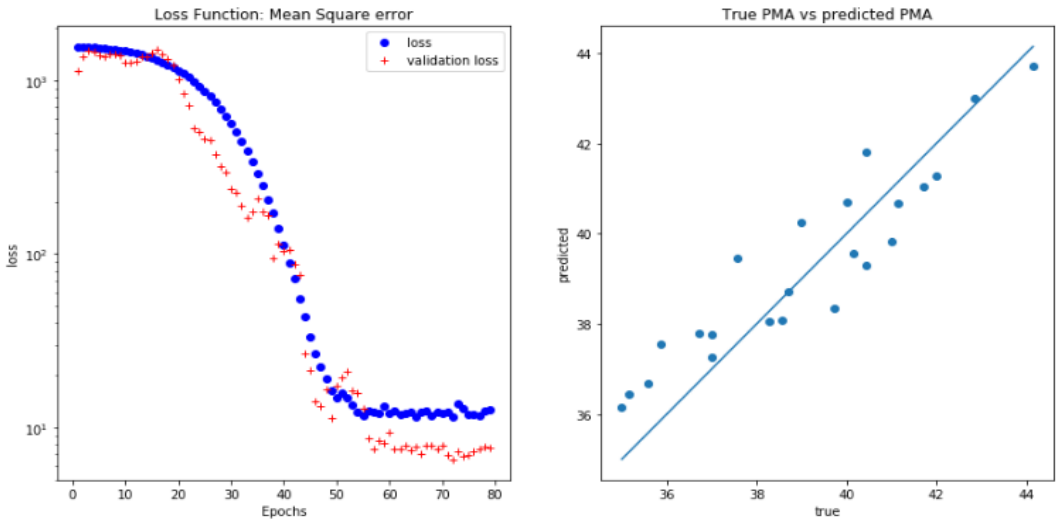


Figure 3.2: Fitting for PMA results (preterm trained model, preterm testing only)

Another fully connected Neural Network was trained, but on the complete 425 data set (term and preterm combined). Using the same architecture and parameters as the previously trained model, we obtain a mean absolute error of 0.7 weeks on the test set. Figure 3.4 shows the evolution of the loss as per training iteration (epoch), as well as a plot of true PMA against predicted PMA. The Pearson correlation between true and predicted reaches r=0.93 and p<0.001. Figure 3.5 summarizes all information for this model.

Training data	Testing data	Validation data	Features	Desired output	Training set MAE	Validation set MAE	Testing set MAE
298 infants (both term and preterm)	64 infants (both term and preterm)	63 infants (both term and preterm)	Structural connectome	PMA	0.62 weeks	0.65 weeks	0.7 weeks

Figure 3.3: Task 1-b Model summary

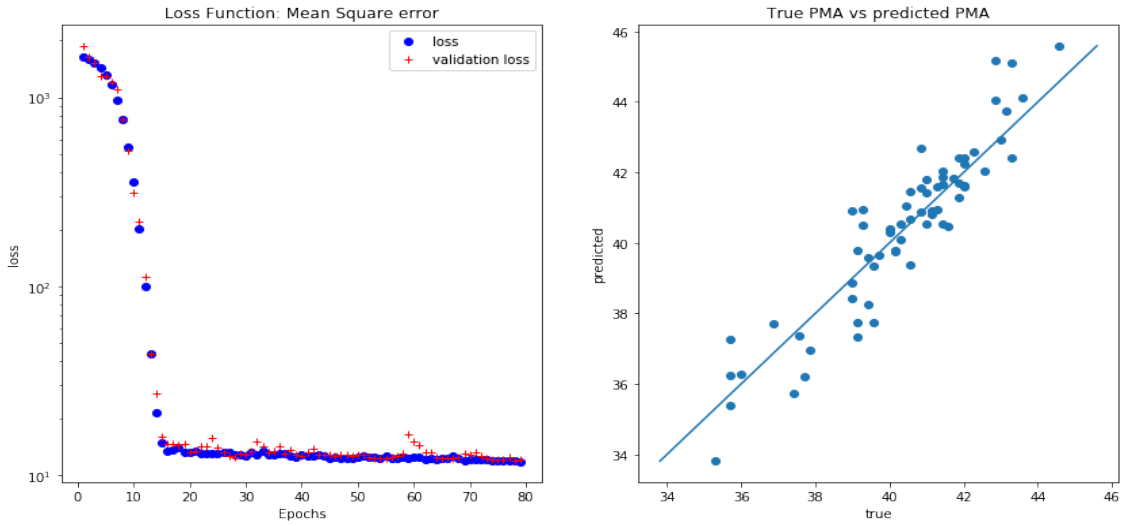


Figure 3.4: Fitting for PMA results (term + preterm testing)

3.1.2 Task 2: Predicting GA from structural connectome

For this task, we train a fully connected Neural Network to predict for GA from structural connectome and PMA. Figure 3.6 shows the evolution of the loss as per training iteration (epoch), as well as a plot of true GA against predicted GA. We reach a mean absolute error of 1.1 weeks upon testing. The Pearson correlation between true and predicted reaches r=0.89 and p<0.001. Figure **??** summarizes all information for this model.

Training data	Testing data	Validation data	Features	Desired output	Training set MAE	Validation set MAE	Testing set MAE
298 infants (both term and preterm)	64 infants (both term and preterm)	63 infants (both term and preterm)	Structural connectome + PMA	GA	0.98 weeks	1.05 weeks	1.1 weeks

Figure 3.5: Task 2 Model summary

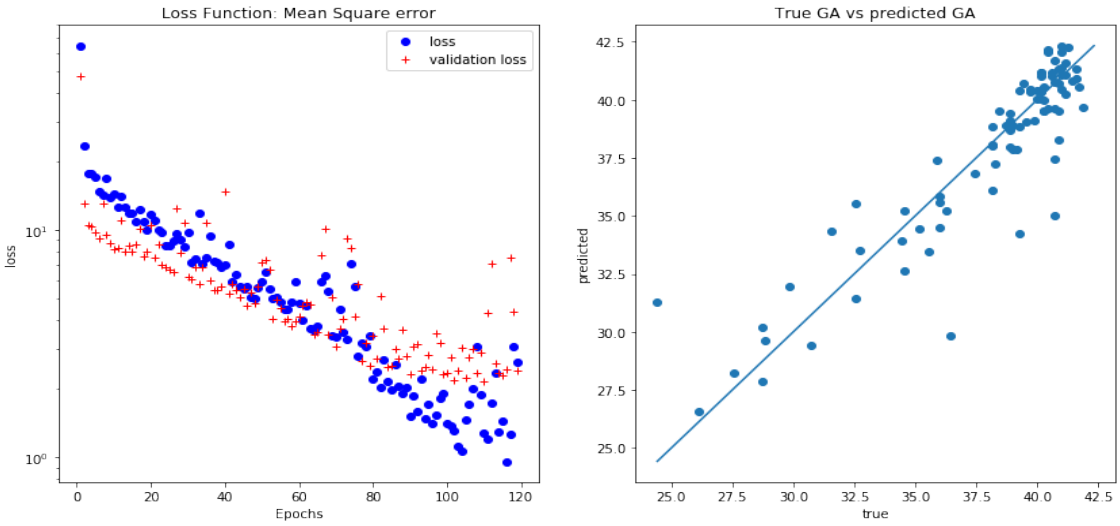


Figure 3.6: Fitting for GA

3.1.3 Task 3: Showing connectivity delay of structural connectome in preterm infants

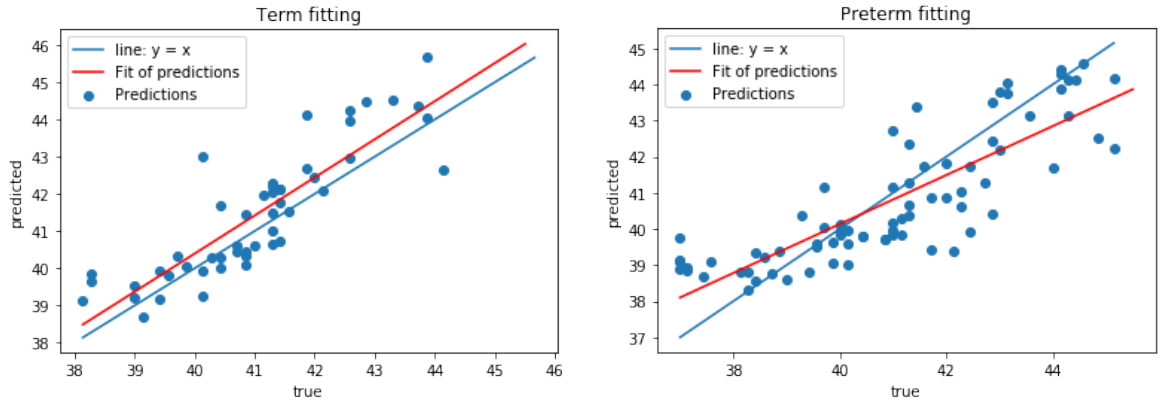
On this task, we train a model only on term infants to predict PMA from structural connectome. We test the model on a term test set as well as on a preterm testing set. Figure 3.9 shows the correlations between predicted and actual PMA upon testing on the term testing set and the preterm testing set. We perform a linear regression on both figures. It can be seen that the linear regression is very close to y=x axis on term babies, whereas it is much less so on preterm babies. The average residuals (Predicted score - actual score) on term testing is +0.3w. The average residuals preterm testing is -0.71 weeks. For clarity, figures 3.7 and 3.8 summarize the models.

Training data	Testing data	Validation data	Features	Desired output	Training set MAE	Validation set MAE	Testing set MAE
210 term infants scanned at PMA>37w	40 term infants scanned at PMA>37w	40 term infants scanned at PMA>37w	Structural connectome	PMA	0.41 weeks	0.52 weeks	0.58 weeks

Figure 3.7: Task 3 Model summary - term fitting

Training data	Testing data	Validation data	Features	Desired output	Training set MAE	Validation set MAE	Testing set MAE
210 term infants scanned at PMA>37w	80 preterm infants scanned at PMA>37w	40 term infants scanned at PMA>37w	Structural connectome	PMA	0.41 weeks	0.52 weeks	1.03 weeks

Figure 3.8: Task 3 Model summary - pre term fitting



(a) Term fitting

(b) Preterm fitting

Figure 3.9: PMA Fitting for Term trained model. Testing on term (left) and preterm (right)

3.2 Section 2: Cognitive score prediction from Structural Connectivity

In this section, we replicate the methods of (Girault et al. 2019) to try to fit for actual cognitive score from structural connectome. As in (Girault et al. 2019), we implement a classification Neural Network to predict for neurodevelopmental score. The figures and results shown below are for prediction of Cognitive Bayley III score.

The classification Neural Network is trained using a K fold cross validation. Figure 3.10 shows a typical training accuracy per epoch during training for one of the folds.

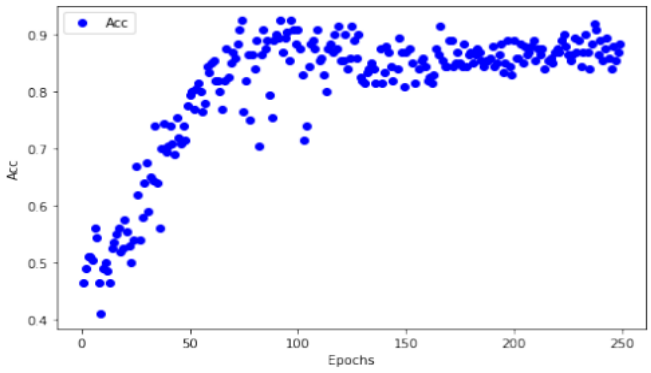


Figure 3.10: Typical accuracy per epoch for fold

Training accuracy is stable at about 90% from epoch 100; this is the case for all training folds. The final accuracy reached - which corresponds to the average accuracy on **testing set** of all 10 folds - We reach an average accuracy of 58%. The huge gap between training accuracy and testing accuracy is mostly due to overfitting. The confusion matrices upon testing on each folds are shown below.

Fold 1				
	Pred AM	Pred BM	Sum	Accuracy
true AM	10	4	14	
true BM	5	4	9	
Sum	15	8	23	
Accuracy				**61%**

Fold 2				
	Pred AM	pred BM	Sum	Accuracy
true AM	8	6	14	
true BM	3	6	9	
Sum	11	12	23	
Accuracy				**61%**

Fold 3				
	Pred AM	Pred BM	Sum	Accuracy
true AM	10	4	14	
true BM	7	2	9	
Sum	17	6	23	
Accuracy				**52%**

Fold 4				
	Pred AM	pred BM	Sum	Accuracy
true AM	11	3	14	
true BM	7	2	9	
Sum	18	5	23	
Accuracy				**57%**

Fold 5				
	Pred AM	Pred BM	Sum	Accuracy
true AM	14	0	14	
true BM	9	0	9	
Sum	23	0	23	
Accuracy				**61%**

Fold 6				
	Pred AM	Pred BM	Sum	Accuracy
true AM	14	0	14	
true BM	8	0	8	
Sum	22	0	22	
Accuracy				**64%**

Fold 7				
	Pred AM	Pred BM	Sum	Accuracy
true AM	2	12	14	
true BM	0	8	8	
Sum	2	20	22	
Accuracy				**45%**

Fold 8				
	Pred AM	Pred BM	Sum	Accuracy
true AM	10	3	13	
true BM	5	3	8	
Sum	15	6	21	
Accuracy				**62%**

Fold 9				
	Pred AM	Pred BM	Sum	Accuracy
true AM	13	0	13	
true BM	8	0	8	
Sum	21	0	21	
Accuracy				**62%**

Fold 10				
	Pred AM	Pred BM	Sum	Accuracy
true AM	13	8	21	
true BM	0	0	0	
Sum	13	8	21	
Accuracy				**62%**

Figure 3.11: Confusion Matrix for classification of term babies for each testing fold

We follow (Girault et al. 2019) method and apply a voting system on each preterm infant structural data. We reach an accuracy of 55% on classification. The following confusion matrix is obtained from applying the voting system on preterm infants.

Preterm voting system				
	Pred AM	Pred BM	Sum	Accuracy
true AM	10	12	22	
true BM	16	25	41	
Sum	26	37	63	
Accuracy				56%

Figure 3.12: Confusion Matrix for voting system on each preterm infant

Based on the probabilities, we first separate the infants into two distinct groups: AM classified and BM classified infants. For each group, we normalize the probabilities between 0 and 1 and correlate these probabilities with corresponding true cognitive score (linear regression). Results for term and preterm infants are shown in the figures below. For term infants, Pearson correlation for above term infants has r = 0.05 with p = 0.12; Pearson correlation for below median infants has r = 0.02 and p = 0.08 We thus have no correlation at all between the classification probabilities and the actual cognitive scores.

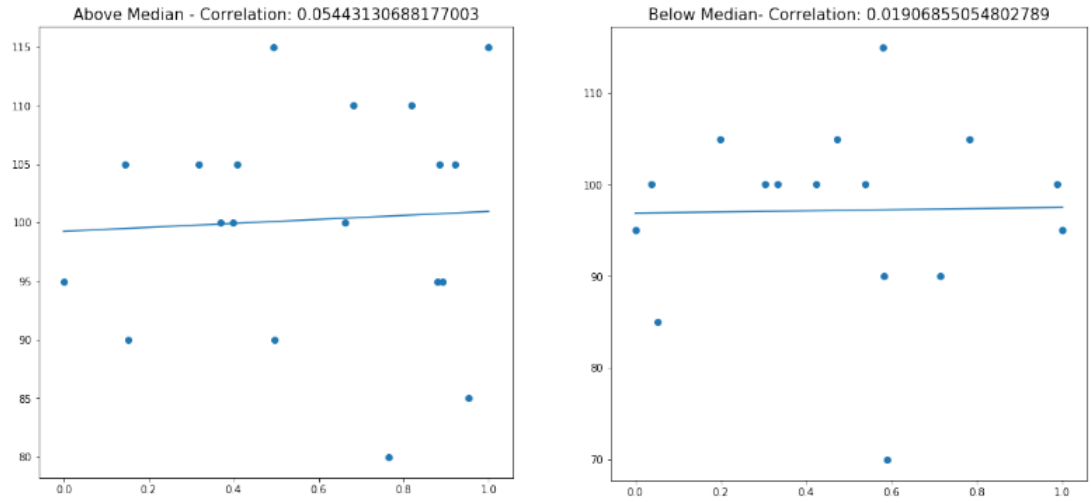


Figure 3.13: Term Probabilities correlated with actual score

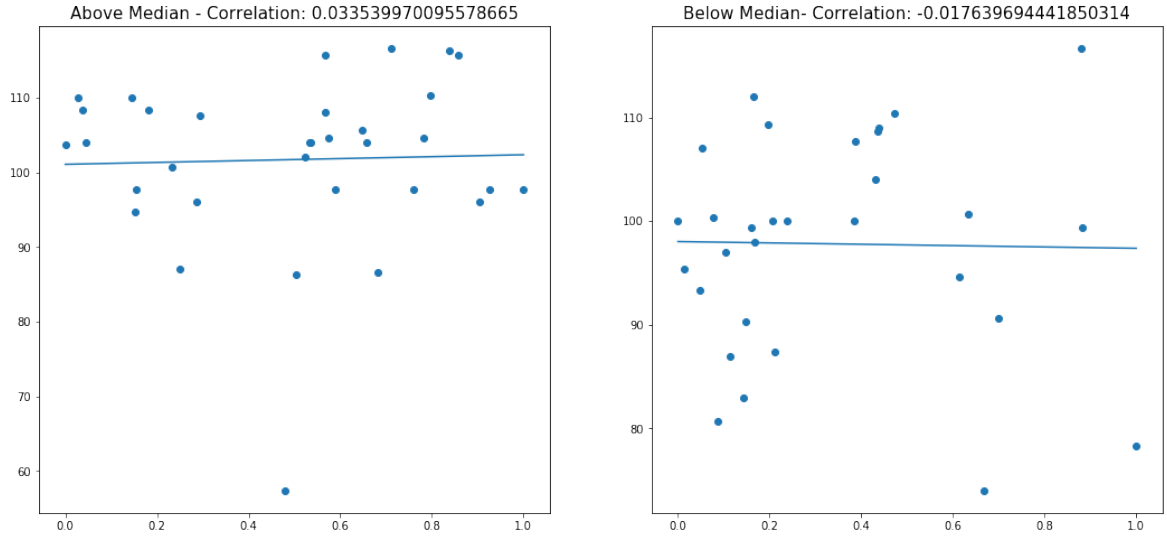


Figure 3.14: Term Probabilities correlated with actual score

In comparison, the results obtained in (Girault et al. 2019) for the classification were an average accuracy 89% on term infants, and 83% on preterm infants. For the prediction of actual cognitive score of term infants, they obtained a correlation of r=0.98 and p<.001 on below median group and a correlation of r=0.92 and p<.001 on above median group. For the prediction of actual cognitive score of preterm infants, they obtained a correlation of r=0.94 and p<.001 on below median group and a correlation of r=0.92 and p<.001 on above median group.

3.3 Section 3: Neural network model interpretation

We first build a Neural Network classifier to classify term and preterm infants from structural connectome. We reach accuracy of 90% on classification upon testing - all the classified infants have GA between 35 and 38 weeks (they are close to threshold value of 37 weeks). The confusion matrix shown below for the classification is shown

below (fig 3.15).

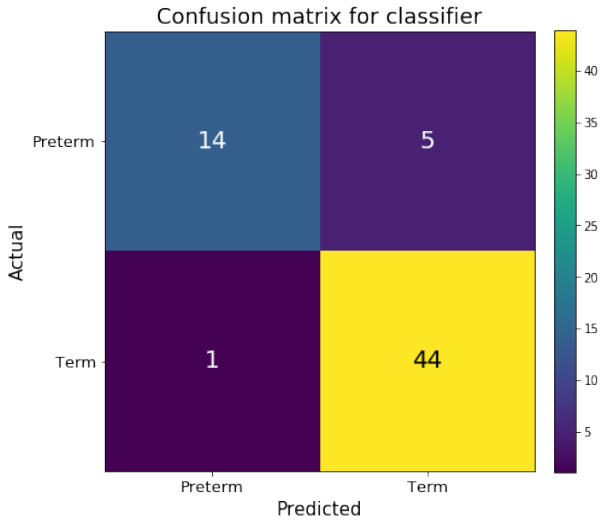


Figure 3.15: Confusion Matrix for classifier

As described in the methods section, we compute the LRP on term and preterm classified infants and obtain the average term and preterm relevance maps. The figure below (fig 3.16) shows these two relevance maps. Note that the most positively activated edges in the term relevance map are the most negatively activated edges on the preterm relevance map, and vice versa. This is coherent with the MNIST data set example given in the introduction.

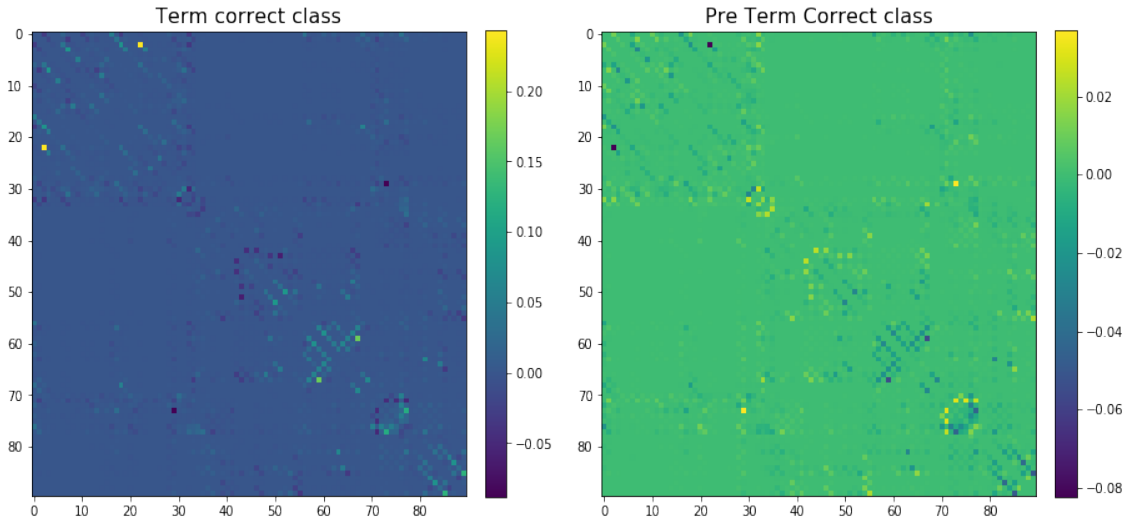


Figure 3.16: Activation maps for Term and Preterm classes

We identify the 5 most positively activated connections for both term and preterm classes - they are shown in the figure below (fig. 3.17) Full table with all 50 connections with corresponding relevance is given in the Additional Tables section (see fig. A.1, A.2)

Term most relevant connections		
Brain region 1	Brain Region 2	Relevance
'2 Precentral_R 2002'	'22 Olfactory_R 2502'	0.2433
'59 Parietal_Sup_L 6101'	'67 Precuneus_L 6301'	0.1692
'85 Temporal_Mid_L 8201'	'89 Temporal_Inf_L 8301'	0.1254
'73 Putamen_L 7011'	'77 Thalamus_L 7101'	0.1176
'50 Occipital_Sup_R 5102'	'52 Occipital_Mid_R 5202'	0.0867

Preterm most relevant connections		
Brain region 1	Brain Region 2	Relevance
'29 Insula_L 3001'	'73 Putamen_L 7011'	0.037
'71 Caudate_L 7001'	'73 Putamen_L 7011'	0.0319
'71 Caudate_L 7001'	'75 Pallidum_L 7021'	0.0315
'71 Caudate_L 7001'	'77 Thalamus_L 7101'	0.026
'30 Insula_R 3002'	'32 Cingulum_Ant_R 4002'	0.0262

Figure 3.17: Table presenting the most relevant connections for term and preterm classes

From the 50 most positively activated connections shown in fig.A.1 and A.2 we count the number of apparition (frequency) of each brain region. For instance, in the 50 most relevant connections for term infants, the parietal superior lobe is comprised in 5 different connections. We thus assign it a frequency of 5. Again, this is done for term and preterm classes. Names and frequency of apparition of the 8 most frequent regions are presented in the table below (fig. 3.18).

Term relevant regions	
Brain Region	Frequency
'60 Parietal_Sup_R 6102'	5
'77 Thalamus_L 7101'	4
'78 Thalamus_R 7102'	3
'3 Frontal_Sup_L 2101'	3
'4 Frontal_Sup_R 2102'	3
'61 Parietal_Inf_L 6201'	3
'62 Parietal_Inf_R 6202'	3
'82 Temporal_Sup_R 8112'	3

Pre term relevant regions	
Brain Region	Frequency
'33 Cingulum_Mid_L 4011'	9
'72 Caudate_R 7002'	5
'78 Thalamus_R 7102'	5
'1 Precentral_L 2001'	4
'2 Precentral_R 2002'	3
'30 Insula_R 3002'	4
'55 Fusiform_L 5401'	4
'34 Cingulum_Mid_R 4012'	3

Figure 3.18: Table presenting the most frequently called brain regions for term and preterm classes

Finally, we take the 50 most important (positive) edges for term and preterm classes and visualize them with the BrainNetView toolbox (Xia et al. 2013). The two following figures (fig. 3.19, 3.20) show the visualisations. For visualization purposes, the relevances are logarithmically scaled to obtain a uniform color distribution.

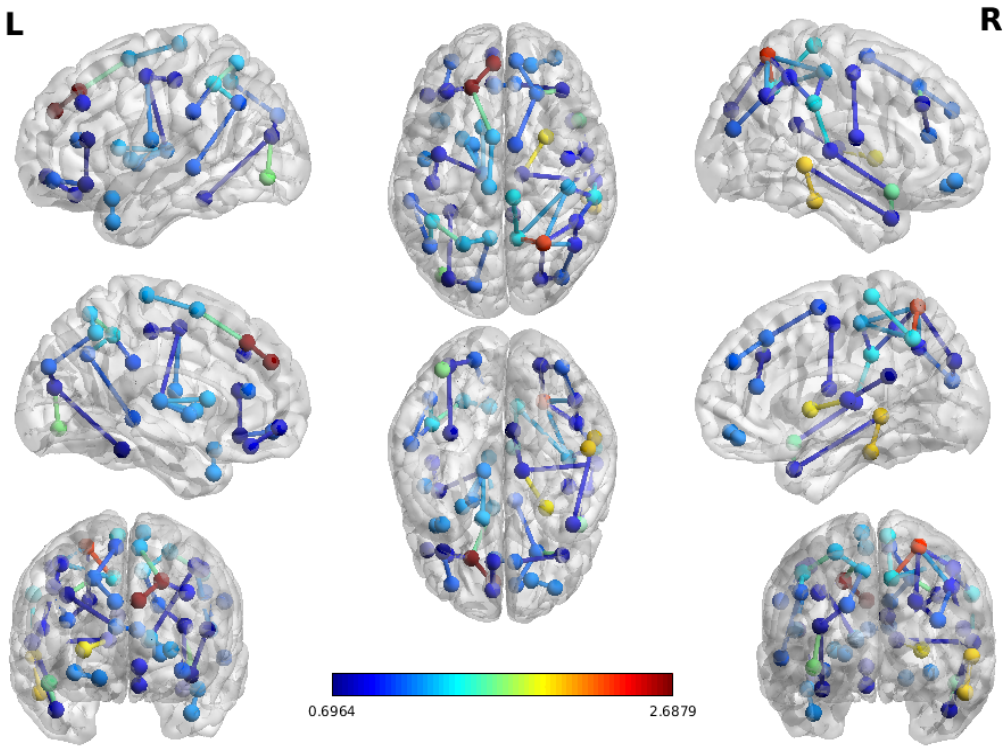


Figure 3.19: Brain visualisation of activated nodes/edges on term infants. Generated with BrainNetViewer (*NITRC: BrainNet Viewer: Tool/Resource Info* n.d., Xia et al. 2013)

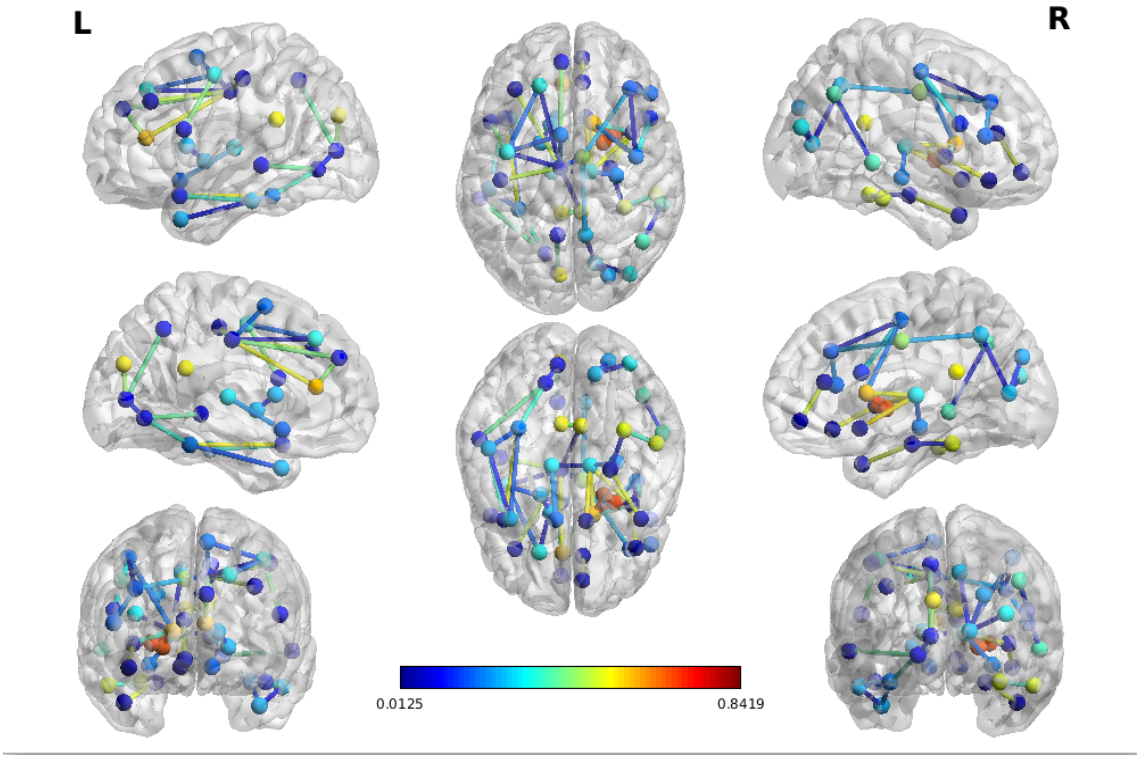


Figure 3.20: Brain visualisation of activated nodes/edges on Preterm infants. Generated with BrainNetViewer (*NITRC: BrainNet Viewer: Tool/Resource Info* n.d., Xia et al. 2013)

Chapter 4

Discussion

4.1 Analysis

4.1.1 Section 1: Age of infants from Structural Connectivity: Towards showing Delay in Preterm Brain Networks

In this section, we have successfully predicted both age at scan and age at birth form structural connectome with high accuracy. This proves that brain connectivity undergoes significant changes during perinatal period. It is particularly important to note that successful prediction of age at birth confirms previous findings which suggest that gestation time has a significant impact on later neurodevelopment. The DNN implemented in task 1 was able to outperform (Kawahara et al. 2017)'s BrainNet CNN. Indeed, they achieved prediction of PMA with MAE of 2.29 weeks and a correlation of 0.85 (predicted vs actual) on prediction of preterm infants. In comparison, our DNN reaches a MAE of 0.84 weeks and correlation of 0.90 on prediction for preterm infants only - a significant improvement from BrainNetCNN. This is with a similar data set size (115 preterm infants in Kawahara et al. and 111 preterm infants with dHCP data used). We also predict PMA on term and preterm infants combined, reaching MAE of 0.7 weeks with correlation of 0.93.

In Task 3, we hypothesized that training a PMA regressor DNN on term infants would lead to significant differences when tested on term and preterm infants. This

was indeed confirmed: prediction upon testing on term infants reached a average residuals (Predict PMA - true PMA) of +0.2 weeks; testing on preterm infants reached mean average error of -0.7 weeks. As the average predictions residuals for preterm is negative, we can conclude that preterm infants have notable developmental delay in brain connectivity compared to term infants.

4.1.2 Section 2: Cognitive score prediction from Structural Connectivity

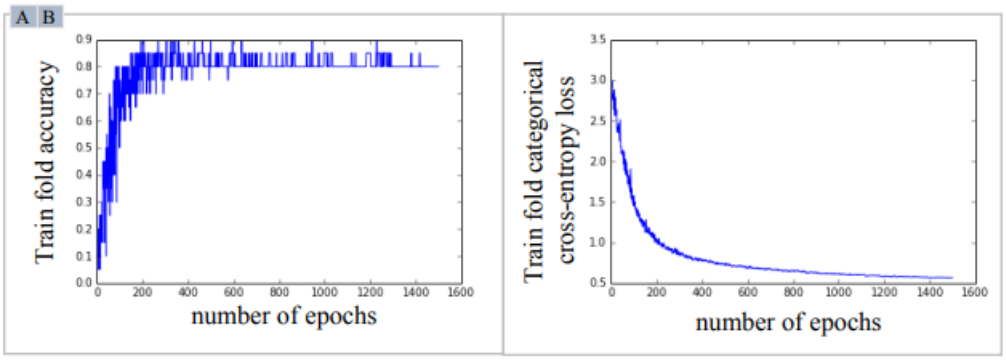
In this section, we were not able to replicate the findings of (Girault et al. 2019) with the dHCP data set, despite precise implementation of their method. This might be the case because of differences in data (both raw and processing methods), as well as potential issues in the proposed methods.

First, Girault et al.'s cognitive assessment is done with Mullen test, which is different from Bayley III test (although both tests have similar internal consistencies). Also, the cognitive assessment they use was assessed at age 24 months, whereas cognitive assessment for dHCP infants was assessed at age 18 months. Another important difference is the connectome extraction method: Girault et al used a probabilistic tractography (Puechmaille et al. 2017) to determine the diffusion connectome which yielded to a 78x78 structural connectome. In comparison, the dHCP constructs the diffusion connectome by calculating the SIFT2-weighted sum of streamlines connecting each pair of regions (Smith et al. 2015), which yields to a 90x90 structural connectome. These factors could have had a significant impact on our inability to replicate the results.

Regarding their results, a surprising fact is that they reach a higher accuracy on prediction of actual neurodevelopmental score than the consistency of the test itself. Indeed, in their method section they report an internal consistency of 91% (test/retest

error) and reliability of 84% for the Mullen Scale at this age. Their reported average error on prediction is of 3.45 points (3.5% of relative error) for term and of 4.47 point (4.5% of relative error) for preterm infants, which is more precise than the test neurodevelopmental test itself.

With regards to the method used, the supplemental material provided in their publication shows the following graph which presents the evolution of accuracy and loss as per epochs for a given fold.



Supplemental Figure 5. *Example training fold classification accuracy and loss plots for one round in the 10-fold cross validation strategy: (A) shows the classification accuracy, and (B) shows the categorical cross-entropy loss.*

Figure 4.1: Evolution of loss function and accuracy per epochs as implemented in (Girault et al. 2019)

The high number of epochs (1500) is extremely prone to overfitting for such a low data size (115 infants). To illustrate this, we implement the same classification DNN architecutre and parameters with dHCP data set, but this time with an additional validation data set. Note that we use a bigger data set, and are thus less prone to overfitting. The following graph shows that, although we reach similar training accuracy and loss, we notice from validation accuracy and loss that the model starts to over fit as soon as the 70th epoch.

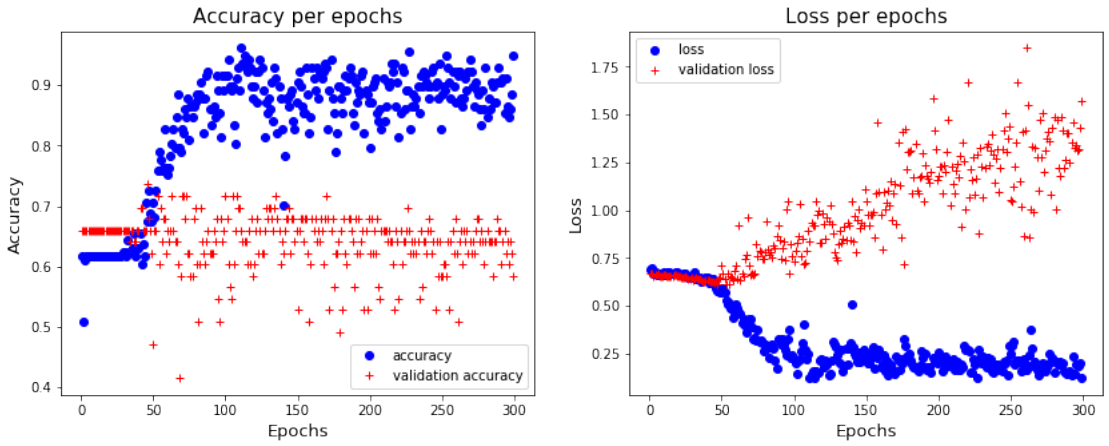


Figure 4.2: Training vs validation accuracy and loss - Overfitting from the 70th epoch

The very high number of parameters used in their DNN architecture (3000 nodes on first hidden layer, 1000 nodes on 2nd hidden layer, 1000 nodes on 3rd hidden layer, 500 nodes on 4th hidden layer, and 100 nodes on 5th hidden layer) is also very prone to overfitting for such a small data size.

Finally, they justify the two step approach used by a difficulty to obtain satisfactory results when directly attempting regression of actual cognitive score from structural connectome using a DNN (with very similar architecture to the proposed classification DNN): *"the single neural network design consistently predicts the ELC 2-year score as either 88.01 (+/- 1.02) or 124.07 (+/- 1.03). This result strongly suggests that predicting one continuous measure using high dimension connectivity feature vectors is too complex, and as a result, the single neural network design is overfit"*. In this case, 88 and 125 correspond to the median score respectively for BM and AM cohort. A DNN consistently outputting the median of the training data set can be due to many different factors such as an inappropriate loss function (in this case, they use MSE), a number of parameters so high that learning cannot converge - this yields in the network being unable to produce any kind of relevant representation. To minimize the error,

the network will thus constantly output the median of the data set. We also implement a regression DNN with same architecture and obtain similar results: output stuck at 114 (+/- 4) for AM model and 87 (+/- 3.5) for BM cohort.

They implement a similar architecture in the classification than the mentioned regression task. The two step approach used the classification output probability to perform a linear regression with the actual cognitive score. This signifies that, for instance, if an infant is classified AM with high probability, the infant should have a high neurodevelopmental score (and vice versa). This suggests that the classification is not only able to correctly classify between AM and BM, but also that the output probability is a significant indicator of the actual score. It is very difficult to believe that using the same architecture and data, the classification DNN outputs probabilities strongly correlated with actual score whereas the regression neural network is unable to find any relevant representation.

It is also expected that variations in socio economic status and education of the family (Raizada & Kishiyama 2010) would be responsible for important variation of the cognitive scores at age 24 months.

4.1.3 Section 3: Neural network model interpretation

In this section, we first manage to classify correctly between term and preterm infants with high accuracy (90%) from structural connectome. Applying the LRP to the model, we are able to identify the connections that were on average most relevant to classifying a given connectome as term or preterm, as well as the brain regions that were most frequently part of these connections. The results are in accordance with previously published work attempting to identify (with statistical analysis) specific connections and regions that are significantly affected by gestational age at birth (Batalle et al. 2017). We indeed identify connections such as Parietal Superior-Pecuneus Left;

Putamen-Thalamus; intra occipital connections. These were also identified in Batalle et al. as being importantly affected by gestational age. Batalle et al. observed that regions such as the temporal, frontal and parietal cortex are showing faster increases in relative weighted connectivity, suggesting earlier development. The LRP method frequently identified these regions as being relevant to classifying term infants. This could suggest that the development of these regions is important throughout the whole gestational period, and that term infants have significantly stronger connections weights within these regions. It is also noted in Batalle et al. that regions such as cingulate gyrus and caudate nucleus matured more slowly. These regions were frequently identified by the LRP method as highly relevant for preterm classified infants. This could suggest that preterm infants are identifiable due to lower connection weights on these regions than term infants. Also, the thalamus is the only brain region appearing to be relevant for both term and preterm classes. This is also coherent with previous findings which showed Thalamocortical connections are essential for brain function, established early in development and significantly impaired following preterm birth (Ball et al. 2015). Finally, it is interesting to note that the connection with highest relevance between the Precentral and the Olfactory lobes (relevance 0.2433) was never identified (to our knowledge) in the literature. We do not know what this connection represent.

4.2 Limitations

Several limitations are to be noted for the work produced in this project. Firstly, although the dHCP currently holds the largest neonate brain connectivity data set in the world, it remains relatively small for reliable Deep Learning applications. Various data augmentation methods could be attempted to improve the performance and robustness of the networks, but this remains very challenging for such complex data set as

connectomes. Also, the small size of the data set does not guarantee the possibility to generalize the findings over larger populations. Even though we implement the Layer wise Relevance Propagation method which gives us an idea of how the Neural Network is able to correctly classify term and preterm infants, we are not able to replicate these interpretations on regression tasks, which are much more complex to interpret. The difficulty to interpret the neurodevelopmental outcomes is preventing us to reliably use models for real diagnosis.

4.3 Conclusion

In this study, Deep Learning has proven to be a successful method that achieved high accuracy on prediction of various neurodevelopmental outcomes from data as complex as brain connectivity. Indeed, Deep Learning appears to be a very promising alternative to traditional statistical analysis, as it is able not only able to replicate statistical findings as shown in section 3 with the identifications of relevant brain regions and connections, but also manages to achieve great performance on complex tasks such as prediction of gestational age at birth and post menstrual age at scan.

Chapter 5

Future Work

A major difficulty encountered in this project was to replicate the results with Functional Connectivity. Indeed, it is much less straightforward to link Functional Connectivity to the various neurodevelopmental scores we've worked with. This is most likely due to the high level of noise in the data, as well as the high temporal variability. Furthermore, Functional connectivity is much more dependant on locality and inter-regional connections; Deep Fully Connected Neural Networks are thus unlikely to perform well for prediction of neurodevelopmental scores as they only consider edge weight. Applying the recent discoveries in Deep Learning for Graphs (Hamilton et al. 2017, Grover & Leskovec 2016) to connectivity data as in (Rosenthal et al. 2018) would be very interesting. Indeed, these novel methods are designed to find representations of graph that preserve neighbooring. This would allow to emphasize information characterizing relation between brain regions, and can be used to infer links which are lacking from a simple adjacency matrix.

As initially planned in the project, a DNN was also implemented to predict functional connectivity from structural connectivity. Although no satisfactory results were obtained from these attempts with DNNs, we believe that using the previously mentioned methods in Deep Learning for Graphs could originate novel conclusions regarding the link between structural and functional connectivity.

Finally, the deep learning interpretation methods only offer a hint of what the model uses to correctly classify. It would be particularly interesting to understand how

decision is made for each test sample, rather than just highlighting notable differences between classes as has been done in this work. This would allow to conclude on specific brain regions that are abnormally developed or undeveloped for each infant - and thus allowing according action.

Bibliography

Albers, C. A. & Grieve, A. J. (2007), 'Test review: Bayley, n.(2006). bayley scales of infant and toddler development–third edition. san antonio, tx: Harcourt assessment', *Journal of Psychoeducational Assessment* **25**(2), 180–190.

Andersson, J. L., Graham, M. S., Drobnjak, I., Zhang, H., Filippini, N. & Bastiani, M. (2017), 'Towards a comprehensive framework for movement and distortion correction of diffusion mr images: Within volume movement', *Neuroimage* **152**, 450–466.

Bach, S., Binder, A., Montavon, G., Klauschen, F., Müller, K.-R. & Samek, W. (2015), 'On pixel-wise explanations for non-linear classifier decisions by layer-wise relevance propagation', *PloS one* **10**(7), e0130140.

Ball, G., Pazderova, L., Chew, A., Tusor, N., Merchant, N., Arichi, T., Allsop, J. M., Cowan, F. M., Edwards, A. D. & Counsell, S. J. (2015), 'Thalamocortical connectivity predicts cognition in children born preterm', *Cerebral cortex* **25**(11), 4310–4318.

Bassett, D. S. & Bullmore, E. (2006), 'Small-world brain networks', *The neuroscientist* **12**(6), 512–523.

Batalle, D., Edwards, A. D. & O'Muircheartaigh, J. (2018), 'Annual research review: Not just a small adult brain: understanding later neurodevelopment through imaging the neonatal brain', *Journal of Child Psychology and Psychiatry* **59**(4), 350–371.

Batalle, D., Hughes, E. J., Zhang, H., Tournier, J.-D., Tusor, N., Aljabar, P., Wali, L., Alexander, D. C., Hajnal, J. V., Nosarti, C. et al. (2017), 'Early development of structural networks and the impact of prematurity on brain connectivity', *Neuroimage* **149**, 379–392.

Brown, C. J., Kawahara, J. & Hamarneh, G. (2018), Connectome priors in deep neural networks to predict autism, *in* '2018 IEEE 15th International Symposium on Biomedical Imaging (ISBI 2018)', IEEE, pp. 110–113.

Cabral, J., Vidaurre, D., Marques, P., Magalhães, R., Moreira, P. S., Soares, J. M., Deco, G., Sousa, N. & Kringelbach, M. L. (2017), 'Cognitive performance in healthy older adults relates to spontaneous switching between states of functional connectivity during rest', *Scientific reports* **7**(1), 5135.

Christiaens, D., Cordero-Grande, L., Hutter, J., Price, A. N., Deprez, M., Hajnal, J. V. & Tournier, J.-D. (2019), 'Learning compact q space representations for multi-shell diffusion-weighted mri', *IEEE transactions on medical imaging* **38**(3), 834–843.

Ciarrusta, J., O'Muircheartaigh, J., Dimitrova, R., Batalle, D., Cordero-Grande, L., Price, A., Hughes, E., Steinweg, J. K., Kangas, J., Perry, E. et al. (2019), 'Social brain functional maturation in newborn infants with and without a family history of autism spectrum disorder', *JAMA Network Open* **2**(4), e191868–e191868.

Cordero-Grande, L., Hughes, E. J., Hutter, J., Price, A. N. & Hajnal, J. V. (2018), 'Three-dimensional motion corrected sensitivity encoding reconstruction for multi-shot multi-slice mri: application to neonatal brain imaging', *Magnetic resonance in medicine* **79**(3), 1365–1376.

Cordero-Grande, L., Teixeira, R. P. A., Hughes, E. J., Hutter, J., Price, A. N. & Hajnal, J. V. (2016), 'Sensitivity encoding for aligned multishot magnetic resonance reconstruction', *IEEE Transactions on Computational Imaging* **2**(3), 266–280.

Ferrazzi, G., Murgasova, M. K., Arichi, T., Malamateniou, C., Fox, M. J., Makropoulos, A., Allsop, J., Rutherford, M., Malik, S., Aljabar, P. et al. (2014), 'Resting state

fmri in the moving fetus: a robust framework for motion, bias field and spin history correction', *Neuroimage* **101**, 555–568.

Francois, C. (2017), 'Deep learning with python'.

Friston, K., Frith, C., Liddle, P. & Frackowiak, R. (1993), 'Functional connectivity: the principal-component analysis of large (pet) data sets', *Journal of Cerebral Blood Flow & Metabolism* **13**(1), 5–14.

Girault, J. B., Munsell, B. C., Puechmaille, D., Goldman, B. D., Prieto, J. C., Styner, M. & Gilmore, J. H. (2019), 'White matter connectomes at birth accurately predict cognitive abilities at age 2', *NeuroImage* **192**, 145–155.

Graph theory: adjacency matrices (2016).

URL: *https://www.ebi.ac.uk/training/online/course/network-analysis-protein-interaction-data-introduction/introduction-graph-theory/graph-0*

Grover, A. & Leskovec, J. (2016), node2vec: Scalable feature learning for networks, *in* 'Proceedings of the 22nd ACM SIGKDD international conference on Knowledge discovery and data mining', ACM, pp. 855–864.

Hamilton, W. L., Ying, R. & Leskovec, J. (2017), 'Representation learning on graphs: Methods and applications', *arXiv preprint arXiv:1709.05584* .

Hinton, G. E., Rumelhart, D. & Williams, R. J. (1986), 'Learning representations by back-propagating errors', *Nature* **323**(9), 533–536.

Honey, C. J., Thivierge, J.-P. & Sporns, O. (2010), 'Can structure predict function in the human brain?', *Neuroimage* **52**(3), 766–776.

Hughes, E. J., Winchman, T., Padormo, F., Teixeira, R., Wurie, J., Sharma, M., Fox, M., Hutter, J., Cordero-Grande, L., Price, A. N. et al. (2017), 'A dedicated neonatal brain imaging system', *Magnetic resonance in medicine* **78**(2), 794–804.

Hutter, J., Tournier, J. D., Price, A. N., Cordero-Grande, L., Hughes, E. J., Malik, S., Steinweg, J., Bastiani, M., Sotiropoulos, S. N., Jbabdi, S. et al. (2018), 'Time-efficient and flexible design of optimized multishell hardi diffusion', *Magnetic resonance in medicine* **79**(3), 1276–1292.

Ioffe, S. & Szegedy, C. (2015), 'Batch normalization: Accelerating deep network training by reducing internal covariate shift', *arXiv preprint arXiv:1502.03167* .

Janocha, K. & Czarnecki, W. M. (2017), 'On loss functions for deep neural networks in classification', *arXiv preprint arXiv:1702.05659* .

Jeurissen, B., Tournier, J.-D., Dhollander, T., Connelly, A. & Sijbers, J. (2014), 'Multi-tissue constrained spherical deconvolution for improved analysis of multi-shell diffusion mri data', *NeuroImage* **103**, 411–426.

Kawahara, J., Brown, C. J., Miller, S. P., Booth, B. G., Chau, V., Grunau, R. E., Zwicker, J. G. & Hamarneh, G. (2017), 'Brainnetcnn: convolutional neural networks for brain networks; towards predicting neurodevelopment', *NeuroImage* **146**, 1038–1049.

Keras: The Python Deep Learning library (n.d.).
URL: *http://keras.io/*

Kuklisova-Murgasova, M., Quaghebeur, G., Rutherford, M. A., Hajnal, J. V. & Schnabel, J. A. (2012), 'Reconstruction of fetal brain mri with intensity matching and complete outlier removal', *Medical image analysis* **16**(8), 1550–1564.

LeCun, Y., Bottou, L., Bengio, Y., Haffner, P. et al. (1998), 'Gradient-based learning applied to document recognition', *Proceedings of the IEEE* **86**(11), 2278–2324.

Lundberg, S. M. & Lee, S.-I. (2017), A unified approach to interpreting model predictions, *in* 'Advances in Neural Information Processing Systems', pp. 4765–4774.

Makropoulos, A., Gousias, I. S., Ledig, C., Aljabar, P., Serag, A., Hajnal, J. V., Edwards, A. D., Counsell, S. J. & Rueckert, D. (2014), 'Automatic whole brain mri segmentation of the developing neonatal brain', *IEEE transactions on medical imaging* **33**(9), 1818–1831.

Montavon, G., Samek, W. & Müller, K.-R. (2018), 'Methods for interpreting and understanding deep neural networks', *Digital Signal Processing* **73**, 1–15.

Mullen, E. M. et al. (1995), *Mullen scales of early learning*, AGS Circle Pines, MN.

Munsell, B. C., Wee, C.-Y., Keller, S. S., Weber, B., Elger, C., da Silva, L. A. T., Nesland, T., Styner, M., Shen, D. & Bonilha, L. (2015), 'Evaluation of machine learning algorithms for treatment outcome prediction in patients with epilepsy based on structural connectome data', *Neuroimage* **118**, 219–230.

Ngoma, Y. M. (2017), Analysis of Control Attainment in Endogenous Electroencephalogram Based Brain Computer Interfaces, PhD thesis, Tshwane University of Technology.

NITRC: BrainNet Viewer: Tool/Resource Info (n.d.).
URL: *https://www.nitrc.org/projects/bnv/*

Poggio, T., Mhaskar, H., Rosasco, L., Miranda, B. & Liao, Q. (2017), 'Why and when can deep-but not shallow-networks avoid the curse of dimensionality: a review', *International Journal of Automation and Computing* **14**(5), 503–519.

Puechmaille, D., Styner, M. & Prieto, J. C. (2017), Civility: cloud based interactive visualization of tractography brain connectome, *in* 'Medical Imaging 2017: Biomedical Applications in Molecular, Structural, and Functional Imaging', Vol. 10137, International Society for Optics and Photonics, p. 101370R.

Raizada, R. D. & Kishiyama, M. M. (2010), 'Effects of socioeconomic status on brain development, and how cognitive neuroscience may contribute to leveling the playing field', *Frontiers in human neuroscience* **4**, 3.

Ranjitkar, S., Kvestad, I., Strand, T. A., Ulak, M., Shrestha, M., Chandyo, R. K., Shrestha, L. & Hysing, M. (2018), 'Acceptability and reliability of the bayley scales of infant and toddler development-iii among children in bhaktapur, nepal', *Frontiers in psychology* **9**, 1265.

Ronneberger, O., Fischer, P. & Brox, T. (2015), U-net: Convolutional networks for biomedical image segmentation, *in* 'International Conference on Medical image computing and computer-assisted intervention', Springer, pp. 234–241.

Rosenthal, G., Váša, F., Griffa, A., Hagmann, P., Amico, E., Goñi, J., Avidan, G. & Sporns, O. (2018), 'Mapping higher-order relations between brain structure and function with embedded vector representations of connectomes', *Nature communications* **9**(1), 2178.

Schuh, A., Makropoulos, A., Robinson, E. C., Cordero-Grande, L., Hughes, E., Hutter, J., Price, A. N., Murgasova, M., Teixeira, R. P. A., Tusor, N. et al. (2018), 'Unbiased construction of a temporally consistent morphological atlas of neonatal brain development', *bioRxiv* p. 251512.

Shi, F., Yap, P.-T., Wu, G., Jia, H., Gilmore, J. H., Lin, W. & Shen, D. (2011), 'Infant brain atlases from neonates to 1-and 2-year-olds', *PloS one* **6**(4), e18746.

Smith, R. E., Tournier, J.-D., Calamante, F. & Connelly, A. (2015), 'Sift2: Enabling dense quantitative assessment of brain white matter connectivity using streamlines tractography', *Neuroimage* **119**, 338–351.

Sporns, O. (2010), *Networks of the Brain*, MIT press.

Sporns, O., Tononi, G. & Kötter, R. (2005), 'The human connectome: a structural description of the human brain', *PLoS computational biology* **1**(4), e42.

Srivastava, N., Hinton, G., Krizhevsky, A., Sutskever, I. & Salakhutdinov, R. (2014), 'Dropout: a simple way to prevent neural networks from overfitting', *The Journal of Machine Learning Research* **15**(1), 1929–1958.

Watts, D. J. (2004), 'The "new" science of networks', *Annu. Rev. Sociol.* **30**, 243–270.

White, J. G., Southgate, E., Thomson, J. N. & Brenner, S. (1986), 'The structure of the nervous system of the nematode caenorhabditis elegans', *Philos Trans R Soc Lond B Biol Sci* **314**(1165), 1–340.

Xia, M., Wang, J. & He, Y. (2013), 'Brainnet viewer: a network visualization tool for human brain connectomics', *PloS one* **8**(7), e68910.

Appendix A

Additional Tables

Node 1	Node 2	Relevance
2 Precentral_R 2002	22 Olfactory_R 2502	0.2433
59 Parietal_Sup_L 6101	67 Precuneus_L 6301	0.1692
85 Temporal_Mid_L 8201	89 Temporal_Inf_L 8301	0.1254
73 Putamen_L 7011	77 Thalamus_L 7101	0.1176
50 Occipital_Sup_R 5102	52 Occipital_Mid_R 5202	0.0867
3 Frontal_Sup_L 2101	7 Frontal_Mid_L 2201	0.0854
2 Precentral_R 2002	18 Rolandic_Oper_R 2332	0.0822
83 Temporal_Pole_Sup_L 8121	87 Temporal_Pole_Mid_L 8211	0.0813
58 Postcentral_R 6002	60 Parietal_Sup_R 6102	0.0797
67 Precuneus_L 6301	69 Paracentral_Lobule_L 6401	0.0654
63 SupraMarginal_L 6211	81 Temporal_Sup_L 8111	0.0649
60 Parietal_Sup_R 6102	64 SupraMarginal_R 6212	0.0629
58 Postcentral_R 6002	66 Angular_R 6222	0.0595
18 Rolandic_Oper_R 2332	68 Precuneus_R 6302	0.0575
57 Postcentral_L 6001	67 Precuneus_L 6301	0.0570
70 Paracentral_Lobule_R 6402	76 Pallidum_R 7022	0.0566
59 Parietal_Sup_L 6101	65 Angular_L 6221	0.0562
72 Caudate_R 7002	76 Pallidum_R 7022	0.0559
57 Postcentral_L 6001	59 Parietal_Sup_L 6101	0.0550

Node 1	Node 2	Relevance
82 Temporal_Sup_R 8112	86 Temporal_Mid_R 8202	0.0536
90 Temporal_Inf_R 8302	16 Frontal_Inf_Orb_R 2322	0.0529
5 Frontal_Sup_Orb_L 2111	9 Frontal_Mid_Orb_L 2211	0.0521
74 Putamen_R 7012	76 Pallidum_R 7022	0.0518
3 Frontal_Sup_L 2101	23 Frontal_Sup_Medial_L 2601	0.0483
30 Insula_R 3002	31 Cingulum_Ant_L 4001	0.0482
48 Lingual_R 5022	66 Angular_R 6222	0.0476
64 SupraMarginal_R 6212	84 Temporal_Pole_Sup_R 8122	0.0475
51 Occipital_Mid_L 5201	65 Angular_L 6221	0.0467
8 Frontal_Mid_R 2202	14 Frontal_Inf_Tri_R 2312	0.0465
3 Frontal_Sup_L 2101	19 Supp_Motor_Area_L 2401	0.0462
60 Parietal_Sup_R 6102	62 Parietal_Inf_R 6202	0.0459
61 Parietal_Inf_L 6201	63 SupraMarginal_L 6211	0.0441
57 Postcentral_L 6001	63 SupraMarginal_L 6211	0.0421
61 Parietal_Inf_L 6201	65 Angular_L 6221	0.0398
59 Parietal_Sup_L 6101	61 Parietal_Inf_L 6201	0.0398
2 Precentral_R 2002	6 Frontal_Sup_Orb_R 2112	0.0394
91 Cerebelum_L 9001	17 Rolandic_Oper_L 2331	0.0383
81 Temporal_Sup_L 8111	83 Temporal_Pole_Sup_L 8121	0.0372
24 Frontal_Sup_Medial_R 2602	26 Frontal_Med_Orb_R 2612	0.0371
90 Temporal_Inf_R 8302	76 Pallidum_R 7022	0.0367
77 Thalamus_L 7101	81 Temporal_Sup_L 8111	0.0365
90 Temporal_Inf_R 8302	56 Fusiform_R 5402	0.0362
85 Temporal_Mid_L 8201	87 Temporal_Pole_Mid_L 8211	0.0357
48 Lingual_R 5022	50 Occipital_Sup_R 5102	0.0350

Node 1	Node 2	Relevance
50 Occipital_Sup_R 5102	54 Occipital_Inf_R 5302	0.0342
7 Frontal_Mid_L 2201	31 Cingulum_Ant_L 4001	0.0337
49 Occipital_Sup_L 5101	59 Parietal_Sup_L 6101	0.0335
35 Cingulum_Post_L 4021	77 Thalamus_L 7101	0.0320
49 Occipital_Sup_L 5101	51 Occipital_Mid_L 5201	0.0318
12 Frontal_Inf_Oper_R 2302	14 Frontal_Inf_Tri_R 2312	0.0318

Table A.1: 50 most important edges for term

Node 1	Node 2	Relevance
29 Insula_L 3001	73 Putamen_L 7011	0.0370
71 Caudate_L 7001	73 Putamen_L 7011	0.0319
71 Caudate_L 7001	75 Pallidum_L 7021	0.0315
30 Insula_R 3002	32 Cingulum_Ant_R 4002	0.0262
71 Caudate_L 7001	77 Thalamus_L 7101	0.0260
42 Amygdala_R 4202	44 Calcarine_R 5002	0.0244
34 Cingulum_Mid_R 4012	35 Cingulum_Post_L 4021	0.0232
55 Fusiform_L 5401	89 Temporal_Inf_L 8301	0.0220
33 Cingulum_Mid_L 4011	67 Precuneus_L 6301	0.0198
32 Cingulum_Ant_R 4002	33 Cingulum_Mid_L 4011	0.0178
65 Angular_L 6221	85 Temporal_Mid_L 8201	0.0170
42 Amygdala_R 4202	46 Cuneus_R 5012	0.0170
90 Temporal_Inf_R 8302	32 Cingulum_Ant_R 4002	0.0158
32 Cingulum_Ant_R 4002	35 Cingulum_Post_L 4021	0.0157
39 ParaHippocampal_L 4111	55 Fusiform_L 5401	0.0156

Node 1	Node 2	Relevance
2 Precentral_R 2002	32 Cingulum_Ant_R 4002	0.0156
43 Calcarine_L 5001	51 Occipital_Mid_L 5201	0.0154
76 Pallidum_R 7022	77 Thalamus_L 7101	0.0148
91 Cerebelum_L 9001	7 Frontal_Mid_L 2201	0.0147
43 Calcarine_L 5001	67 Precuneus_L 6301	0.0136
70 Paracentral_Lobule_R 6402	74 Putamen_R 7012	0.0134
86 Temporal_Mid_R 8202	88 Temporal_Pole_Mid_R 8212	0.0134
43 Calcarine_L 5001	49 Occipital_Sup_L 5101	0.0130
90 Temporal_Inf_R 8302	18 Rolandic_Oper_R 2332	0.0126
54 Occipital_Inf_R 5302	88 Temporal_Pole_Mid_R 8212	0.0124
54 Occipital_Inf_R 5302	86 Temporal_Mid_R 8202	0.0123
37 Hippocampus_L 4101	77 Thalamus_L 7101	0.0122
90 Temporal_Inf_R 8302	2 Precentral_R 2002	0.0121
18 Rolandic_Oper_R 2332	32 Cingulum_Ant_R 4002	0.0119
91 Cerebelum_L 9001	71 Caudate_L 7001	0.0119
20 Supp_Motor_Area_R 2402	76 Pallidum_R 7022	0.0117
7 Frontal_Mid_L 2201	13 Frontal_Inf_Tri_L 2311	0.0116
7 Frontal_Mid_L 2201	33 Cingulum_Mid_L 4011	0.0113
91 Cerebelum_L 9001	11 Frontal_Inf_Oper_L 2301	0.0104
82 Temporal_Sup_R 8112	88 Temporal_Pole_Mid_R 8212	0.0101
29 Insula_L 3001	71 Caudate_L 7001	0.0101
46 Cuneus_R 5012	54 Occipital_Inf_R 5302	0.0100
46 Cuneus_R 5012	84 Temporal_Pole_Sup_R 8122	0.0096
42 Amygdala_R 4202	66 Angular_R 6222	0.0095
90 Temporal_Inf_R 8302	10 Frontal_Mid_Orb_R 2212	0.0092

Node 1	Node 2	Relevance
22 Olfactory_R 2502	32 Cingulum_Ant_R 4002	0.0092
22 Olfactory_R 2502	30 Insula_R 3002	0.0091
6 Frontal_Sup_Orb_R 2112	32 Cingulum_Ant_R 4002	0.0088
32 Cingulum_Ant_R 4002	56 Fusiform_R 5402	0.0087
25 Frontal_Med_Orb_L 2611	31 Cingulum_Ant_L 4001	0.0086
39 ParaHippocampal_L 4111	87 Temporal_Pole_Mid_L 8211	0.0085
29 Insula_L 3001	77 Thalamus_L 7101	0.0083
15 Frontal_Inf_Orb_L 2321	29 Insula_L 3001	0.0083
54 Occipital_Inf_R 5302	82 Temporal_Sup_R 8112	0.0082
21 Olfactory_L 2501	77 Thalamus_L 7101	0.0080

Table A.2: 50 most important edges for preterm