The Mismatch Negativity: Four machine learning approaches for predicting dyslexia in infants on EEG data

Nadine Prins (9077197)
Artificial Intelligence
Utrecht University
November 18, 2022

A thesis presented for the degree of
Msc Artificial Intelligence

Supervisor and first examiner:
Dr. H.G. Schnack
Second examiner:
Prof. Dr. F.N.K. Wijnen
Daily supervisor:
Dr. C.M. Moore

Acknowledgements

I would like to express my deepest appreciation to Dr. Candace Makeda Moore for the supervision over the last 6 months. She taught me how to work in a professional environment and gave me guidance on my goals after finishing this thesis. I am super thankful for all the time she has spent on me to increase the quality of my work. I am looking forward to continuing our work together on the eegyolk publication. I want to give special thanks to Dr. Walter Baccinelli, Djura Smits, Dr. Eva Viviani, Malte Lüken, dr. Cunliang Geng, Sven van den Burg, Giulia Crocioni, and to the other researchers at the eScience Center, who were more than happy to help me with the issues I was facing while writing my thesis. They made me feel welcome and at home by inviting me to activities and adopting me into their team.

I am also very grateful for the guidance Dr. Hugo Schnack gave me. His feedback was of great value and gave me the needed directions for my thesis. His way of supervising helped me in feeling confident and more calm, which is usually challenging for me when I need to perform. I would like to thank Prof. Dr. Frank Wijnen for taking the time to be an examiner of this thesis and for his input on the preliminary results, and Dr. Karin Wanrooij for teaching me everything I needed to know about EEG data.

I want to thank my family, who are always there for me to support me and help me to be proud of my accomplishments. Without them, I wouldn't be where I am right now and I can't express in words how grateful I am for that. I also want to thank all my friends, in special Monika, Sebastiaan, Cheyenne, and Enrique for supporting me and being there for me when I needed someone to talk to.

Abstract

Electroencephalogram (EEG) in combination with machine learning (ML) techniques is becoming an increasingly popular method in medicine for clinical disorders prediction. This study applies ML techniques to the ePod dataset developed by the ePODIUM project, for the prediction of developmental dyslexia in infants. The dataset contains EEG recordings of 129 infants, existing out of two groups with dyslexic- and non-dyslexic parents, obtained from an experiment with auditory stimuli for eliciting a Mismatch Negativity (MMN). Four different approaches for feature selection are used to see the differences in performance on different ML algorithms. The baseline approach uses the MMN of all EEG channels. The second approach includes EEG channels reported in the literature as the most informative. The t-test approach uses significance testing, verified using a t-test on the ePod data, and resulted in a selection of significantly different channels between the two groups. The final approach uses the channels from the ePod data that show the highest connectivity with other channels. The algorithms used on the different feature input approaches are support vector machine, logistic regression, decision tree, multilayer perceptron, and convolutional neural network. The convolutional neural network showed the highest performance in combination with the features of the t-test approach with an accuracy of 73%. However, this result is not significant ($p=0.447$) because of high variation in model performance. The connectivity approach performs also well based on average accuracy with the convolutional neural network. The traditional machine learning algorithms support vector machines and logistic regression can learn from the t-test and connectivity with moderate accuracy of 60%. The results show that data-driven selected features, using significance testing and connectivity, are promising in predicting developmental dyslexia in infants in combination with deep learning and traditional machine learning models.

Contents

1 Introduction

Centuries ago, dyslexia was identified as word blindness by a German Professor, Adolph Kussmaul. He was the first one to recognize the possibility of the inability to read [1]. Nowadays, dyslexia is characterized by having difficulties with accurate word recognition and poor spelling abilities [2]. A rough estimate is given that 5-10% of the world population has a form of dyslexia. This implies that there are approximately 2 children with dyslexia in a classroom of 30. Although there is no evidence that dyslexia can be cured, there is scientific proof that early interventions are effective remediation of reading problems [3]. This raises the importance of detecting dyslexia as early as possible. However, detecting dyslexia in infants by observation only is an impossible job, since the children did not yet develop reading and speaking skills. For this research dyslexia refers to developmental dyslexia. An upcoming method to detect and understand developmental disorders is using neurophysiological data using electroencephalography (EEG). EEG measures the electrical activity in the brain using placed electrodes along the scalp. With EEG, scientists were able to successfully detect autism at a younger age by using machine learning methods. They achieved this by measuring brain responses after exposing subjects to a certain stimulus and by training a cross-validated machine learning model to predict whether an infant will develop autism [4] [5]. The Dutch Dyslexia Programme (DDP) created a dataset containing EEG recordings of young infants to research the risk of dyslexia [6].

The University of Utrecht started a project in collaboration with UMC Utrecht and eScience Center to explore the possibilities of using machine learning to predict later language/literacy performance on the individual level. The name of the project is: early Prediction Of Dyslexia in Infants Using Machine learning (ePODIUM). This master thesis is part of the ePODIUM project. The DDP dataset stimulus-response paradigm came out to be not suitable for predicting dyslexia, which resulted in generating a new data set to increase the ability of dyslexia detection, by using a different protocol for the experiment. This newly generated data contains EEG data of 129 infants labeled at risk or control, based on whether the parents are dyslexic or not. In this thesis, the dataset is referred to as the ePod dataset.

Related studies on developmental dyslexia and EEG show a bottom-up approach, where theory is based upon the outcome of machine learning models. For this research, a top-down approach is proposed to predict dyslexia by first assessing the neurophysiological theory of dyslexia and using the findings as extra information for the model. A theory-driven model can contribute to better model performance by reducing the dimensionality of the model to only keep relevant features.

For this project, algorithms were trained with data-driven features as input. With features is meant electrodes corresponding to a specific part of the brain.

Data-driven refers to the selection of features. The signals of those electrodes will be transformed into a mismatch negativity (MMN). MMN is the response of the brain after an abnormality in a sequence of sensory stimuli [7]. The ePod dataset is based on the MMN, and therefore it is interesting to see if a model can find a pattern in this response. Four different input approaches are used for the models. The first approach is using the MMN of all electrodes as input to set a baseline. The second approach is based on previous studies. Multiple studies will be assessed to get a better understanding of which electrodes are relevant for children with dyslexia and how this differs from non-dyslectic children. The third and fourth approaches are based on data analysis of the ePod data. Significant testing between the at risk and control groups will be done to see which electrodes are significantly different in the two groups. The final approach calculates the connectivity between the electrodes to reduce the number of features. Different algorithms were tested to discover the best-performing model for predicting dyslexia at an early age. From this, the research question is:

To what extent can data-driven features extracted from EEG recordings be used with machine learning models to predict the risk of developmental dyslexia in infants?

To answer this research question, a literature review has been done followed by data analysis. Both the literature and the analysis are of great importance for selecting the input for the models. Next, the different models are assessed on their usability. Finally, the model outcomes are evaluated to see if a theoretical approach can be useful for predicting the risk of developmental dyslexia.

2 Background

2.1 Developmental Dyslexia

Dyslexia is a widespread disability ranging from 5% to 17.5% of the human population. The variability can be explained by the loose definition of dyslexia and some different factors such as sexual bias, different ways of measuring IQ, and differences in spoken language [8]. Even though dyslexia is quite common, there are multiple misunderstandings about people with reading disabilities, for example, that reading disabilities are caused by visual perception problems or that people with dyslexia only have problems with word reversals (saw/was) [9]. The most accepted definition of dyslexia so far is:

> Dyslexia is a specific learning disability that is neurobiological in origin. It is characterized by difficulties with accurate and/or fluent word recognition and by poor spelling and decoding abilities. These difficulties typically result from a deficit in the phonological component of language that is often unexpected to other cognitive abilities and the provision of effective classroom instruction [2].

Accordingly, dyslexia is neurobiological in origin and can be strengthened by a deficiency in education. Other factors that can put a child more at risk for developing dyslexia are poverty, developmental delay, speech or hearing impairments, or learning a second language. Those conditions can be more seen as correlates to reading disabilities instead of a cause for dyslexia [10]. There are multiple hypotheses about the cause of dyslexia. The mainstream hypothesis is the deficit of access to phonemic language units retrieved from long-term memory [8]. A phoneme is the smallest unit of a sound and can help distinguish similar words from each other. The words 'pet' and 'bet' are for example distinguishable by the letters p and b. People with dyslexia find it harder to differentiate those words since they have difficulties with learning the letter and phoneme associations. Research validates this theory by showing phonemic deficits in university-educated participants [11]. The study showed that all dyslectic participants have reduced short-term verbal memory and phonemic awareness. Ramus and colleagues also tested the magnocellular theory. The magnocellular theory argues that there is a dysfunction in the neurons responsible for the visual system. The magnocellular system is important for visual attention, control of eye movements, and visual search. Those three components influence the reading ability [12]. In the study 2 out of 16 participants suffered from a visual deficit. The third theory is the cerebellar theory, where the cerebellum is dysfunctional [13]. The cerebellum has a function in motor control, which signifies plays a role in speech. It also has a function in the automatization of over-learned tasks, for example walking, biking and reading. Ramus and colleagues only found 4 out of 16 dyslexic participants with a motor deficit. Another theory is the rapid auditory processing theory [14]. Participants with dyslexia show poor performance on auditory tasks with rapidly varying sounds. There is even a higher deficit once the auditory task uses similar phonemes. 10

out of the 16 participants showed this auditory deficit. Ramus's study resulted in support of the phonological theory of developmental dyslexia, with additional auditory deficits. This could indicate that audio can play a role in analyzing dyslexia.

Technology to analyze the brain has become more advanced over the last couple of years. This makes it easier to measure the effects of interventions on dyslexia. Studies showed that early interventions on dyslexia can increase the reading ability of the participant. Aylward and colleagues did an experiment to see whether a 28-hour intervention has an impact on brain activation during tasks of identifying letter sounds. The results show that participants with dyslexia have a significant increase in brain activation in areas important for reading and language [15]. Another study showed similar results. After the intervention, the experimental group had increased activation in the left hemispheric regions which are important for reading [16].

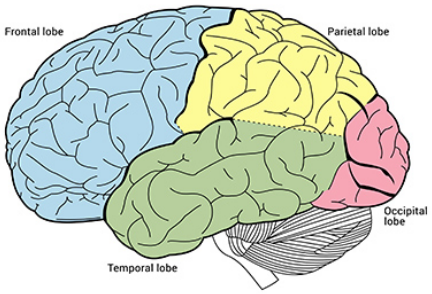
2.2 Brain anatomy

There is no specific area in the brain that has reading as function[17]. Instead, there are brain regions whose functions are involved with reading, those functions are spoken language and object recognition. The brain had two hemispheres, in most people the left hemisphere, is responsible for speech and language processing, and therefore, reading. The hemisphere can be divided into four areas illustrated in figure 1. The focus of this research is only on the cerebrum, with the exclusion of the insula since its interior location. The frontal lobe controls speech, reasoning, planning, emotions, and consciousness. Studies also showed that the frontal lobe is involved during silent reading [16]. The Parietal lobe controls sensory perceptions and can link spoken and written language to memory to give it meaning in a way we can understand what we hear and read. The primary visual cortex is located in the occipital lobe and is therefore responsible for the identification of letters and other visual inputs. Finally, the temporal lobe is responsible for encoding auditory information into memory [9].

Two other systems are also involved in language processing, the left parietotemporal area (Broca's area) which is involved in word analysis, and the left occipitotemporal area (Wernicke's area) which seems to be involved in automatic rapid access to whole words and increases fluent reading [16]. Broca's area is located in the frontal lobe of the dominant hemisphere. Wernicke's area is located in the cerebral cortex. Broca's area is primarily involved in producing language, while Wernicke's area focuses on the comprehension of spoken and written language.

The explanation of which area has which function is more complex than stated above, which makes this a rather global explanation of the functions of the different lobes. The brain area functions are more specified within the lobes.

Figure 1: Brain areas in the left hemisphere [18]



The brain itself is made out of two types of material, grey matter composed of neural cell bodies for processing information, and white matter composed of myelinated axons facilitating communication between nerves. Research has found that people with developmental dyslexia have less gray matter in the parietotemporal area, which means that there is less processing of words. It can also lead to problems in processing the sound structure of language, phonological awareness [19]. People with dyslexia also show less white matter in the same area, which lessens the ability of the brain regions to communicate with each other. fMRI studies showed that there was more activation in the brain areas important for reading in non-dyslectic children. Their left hemisphere is significantly more activated compared to their right hemisphere, while right-handed dyslexic children make more use of the right hemisphere to compensate for the lack of activation in the left hemisphere [16].

The theories in this paragraph are only valid for right-handed people since their left hemisphere is dominant. Research showed that for left-handed people, the right hemisphere shows more dominance. The brain structure is approximately symmetrical in both sides. The difference is in the higher activation of the right hemisphere for left-handed people. Left-handed people constitute about 10% of the world's population[20].

2.3 Electroencephalography

Electroencephalography (EEG) measures the electrical activity of the different brain parts using placed electrodes along the scalp. EEG signals capture the activity in a specific brain area over a period of time. This is measured by voltage fluctuations from ionic current, the flow of electrical charge through ion channels, within the neurons of the brain. This flow is measured in amperes. One ampere is equal to one coulomb, the amount of electricity flowing per second. Voltage is used to set the current in motion, and can also be called electrical potential since it is the potential of the current to flow. This voltage is measured in volt (V). On the other hand, resistance inhibits the current from flowing and is measured in Ohm. The final part concerning physics, which is important to

know, is the role of magnetism in electricity. Every current is surrounded by a magnetic field, that circles around the conductor of the current. This magnetic field can pass through a different conductor and can cause a small current in this conductor. This can result in electrical noise.

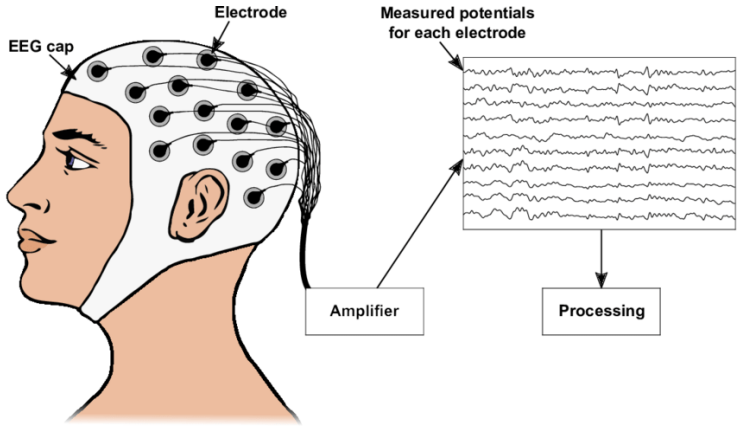
Physics tells us that electrical activity arises when there is an electrical potential, which causes a current to flow through a conductor. These currents are captured in the neurons of the brain. The electrical activity between neurons can be divided into action potentials and postsynaptic potentials. Action potentials send signals around the body by voltage changes in the membrane of the neuron. This membrane surrounds the neuron and consists of ion channels, which can allow charged atoms of sodium and potassium channels to pass into and out of a neuron. In general, there are far more sodium ions (Na+) outside the neuron compared to the potassium ions (K+) inside the neuron. This results in an electrical imbalance between the inside and the outside, which causes a voltage difference across the cell membrane. The ion channel can change shape, allowing a particular ion to pass through the membrane. When the ions pass those channels, the voltage changes. Those ion channels are a passive mechanism. The ions only migrate towards lower concentrations. More sodium is being transported out, which makes the inside of the cell negatively charged compared to the outside. This is the cell membrane's resting potential, where there is an imbalance of ions across the cell membrane. The action potential is a temporary shift, caused by a triggering event from other connecting neurons, where sodium channels open and let sodium ions into the cell, which causes depolarization of the neuron. The voltage in the cell will reach a positive peak. To gate the positive ions outside the cell, the potassium channel opens to repolarize the cell to its resting potential.

Postsynaptic potentials are the voltages that arise from transmissions between two neurons. The area between two neurons where they are close enough to pass information to one another is called a synapse. The sending neuron is called the presynaptic neuron and passes the signal to the receiving postsynaptic neuron. The presynaptic neuron has vesicles that contain a large group of neurotransmitters. When the presynaptic neuron reached its action potential, it releases the neurotransmitters in the synaptic space between the two neurons. the postsynaptic neuron contains receptors that can bind with the released neuron transmitters. This release and binding of neurotransmitters cause a change in voltage in the cells.

EEG measures the above-described voltage changes in the brain by placing sensors on the scalp of a participant. The sensors measure the electrical activity in the cerebral cortex, which is the outer layer of the neural tissue of the brain. The sensors, called electrodes, measure the electrical activity of groups of neurons that transmit signals at the same time. One of the reasons why the measurement is done on a group of neurons is that the measurement of a single neuron would contain a lot of noise caused by the magnetic field of the

adjacent neurons. There are different ways to place the sensors on the scalp, those placings are called montages. Montages display activity over the entire head and make it easier to localize which electrode belongs to which area of the brain [21]. The setup of an EEG experiment can be seen in figure 2. The figure shows that the electrodes are placed on the head by using an EEG cap. An amplifier strengthens the EEG signal and displays it in a diagram with a signal per electrode. Each signal is a single EEG recording.

Figure 2: Setup of an EEG experiment, adopted from [22]



The oscillations shown in an EEG recording are classified into different frequency patterns. The so-called EEG waveforms are divided based on their frequency, and every waveform explains the state of a person. The first waveform is gamma, with frequency differences between 30 to 80 hertz (Hz). Here a person is in a problem-solving state of mind and highly concentrated. The second waveform is the beta waveform (12-30Hz), where the person is active. The alpha waveform (8-12Hz) represents the brain being at rest and the theta waveform (4-8Hz) represents sleep. The final waveform is the delta (0.5-4Hz), where a person is in deep sleep. The higher the frequency, the more bumps the EEG recording shows.[23]. Fourier transformation can be used to sum the oscillations at the different frequencies, to give insight into what sine wave frequencies make up a signal [24].

Event-Related Potentials (ERP) refer to an average of EEG responses after a certain stimulus in a so-called oddball paradigm, where there is a sequence of standard stimuli and a randomly occurring deviant. In section 4.1.1 an illustration of an oddball paradigm will be given. The reaction to an event is measured by taking small segments of an EEG after the stimuli occurred. Each segment is referred to as a trial. An example of an ERP can be found in figure 3. The figure shows the response of electrode Fz of a single child for both the standard stimuli and deviant stimuli. The figure also shows the MMN. The MMN is calculated

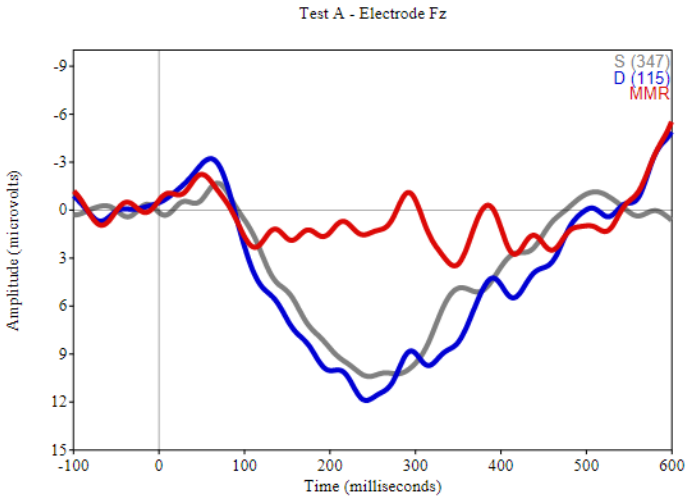
by subtracting the deviant stimuli from standard stimuli to determine the difference in response. More about the use of the MMN can be found in section 3.2.

EEG data is from origin very noisy, which makes it difficult to look at the signal of interest. Besides adding more data to cancel out random noise, the data can also be filtered to make it less noisy. There are different filtering methods with different purposes to achieve interpretable data. A band-pass filter can filter out low and high frequencies. There is also a notch filter, which can filter out noise at a specific frequency. EEG data can also contain artifacts. Those artifacts are caused by, for example, blinking of the eyes. Artifacts can also arise by simple eye movements. Those artifacts need special treatment since it is undesirable to have those anomalies in your data which can lead to false conclusions. Another problem with EEG is that sometimes a signal can be bad because the electrode isn't connected well to the scalp. Those bad signals can influence the analysis and therefore should be detected and removed or adjusted. To cut a long story short, it is highly necessary to use the correct filtering to retrieve valid information from the EEG recordings [24].

2.3.1 Advantages and disadvantages

EEG is becoming a more popular tool in psychology and clinical medicine nowadays. One of the biggest advantages is the ability to see brain activity in real-time, at the level of milliseconds (ms). This means that EEG has a high temporal resolution when it comes to brain imaging, which is also continuous. This

Figure 3: ERP example of a single trial of one participant from the ePOD data



characteristic of EEG showed success in determining which processes are influenced by experimental manipulation, identifying multiple neurocognitive processes, and by measuring behavioral responses to subjects who are incapable of making a response, like infants. EEG can also be used as biomarkers in medical applications, by measuring aspects of the brain function to detect abnormalities in the brain that can be related to neurological and psychiatric diseases [24].

There are also disadvantages when using EEG, besides the advantages of EEG. EEG is very coarse since all neurons are connected. Therefore it is hard to determine which neuron provides which signal. It is very challenging to isolate and measure the internal underlying components based on the data that you can record from the scalp. Since ERP waveforms typically reflect the sum of multiple internal, underlying components. This is called the superposition problem [24]. Another big problem with EEG is the noise created by for example blinking or the beating of the heart. There is also noise in the signals based on the magnetic field of surrounding neurons as explained earlier.

2.4 Machine Learning

Machine learning is a very popular field within AI for classification or regression problems. Machine learning models can be supervised, unsupervised, or reinforced. Supervised machine learning models contain labels and therefore the model can train on the expected outcome. Unsupervised models have an unknown outcome and are harder to validate. Reinforcement models maximize performance based on interactions with the environment by using a reward system. Multiple research has been done on which machine learning model is most effective on EEG data [25] [26] [27]. Combining their findings, a supervised classification model is the best-performing method on EEG data. Some models within this class are support vector machines (SVM), logistic regression and decision trees.

Another popular field within machine learning is deep learning. Deep learning models are inspired by the human brain and therefore exist out of multiple layers of neurons that pass information through, hence the name neural network. A neural network has different weights which represent the strength of a connection between neurons. Studies have shown that neural networks have a good performance on EEG data classification. The best-performing algorithms are convolutional neural networks (CNN), long short-term memory (LSTM) and a CNN-LSTM hybrid model [28]. Another study showed the effectiveness of a deep convolutional neural network (DCNN) on epileptic EEG classification [29].

For this research 5 different models were used to predict developmental dyslexia. The motivation for why those models were chosen can be found in section 4.3. A brief explanation of each chosen model will be given in the following sections to understand the basic principles of the algorithms.

2.4.1 Support Vector Machine

Support vector machine (SVM) is a machine learning algorithm that can be used for both regression and classification tasks. The objective of the SVM is to find an optimal hyperplane in a space with multiple dimensions. This hyperplane serves as a classification decision boundary and can be set by finding the maximum margin, which is the distance between the hyperplane and the closest data point of both classes. Those data points closest to the hyperplane are called support vectors. The equation of the hyperplane is as follows:

$$\mathbf{w}^T \phi(x) + b = 0 \tag{1}$$

In this equation, **w** represents the weight vector, $\phi(x)$ represents the mapping from the data point into the feature space and **b** is the bias term. To optimize this separation hyperplane, we need to know the distance between the hyperplane and a data point. The distance between a line and a point is the length of the line segment that is perpendicular to the hyperplane and passes through the point. The formula of the distance is:

$$d_H(\phi(x_0)) = \frac{|\mathbf{w}^T \phi(x) + b = 0|}{||w||_2} \tag{2}$$

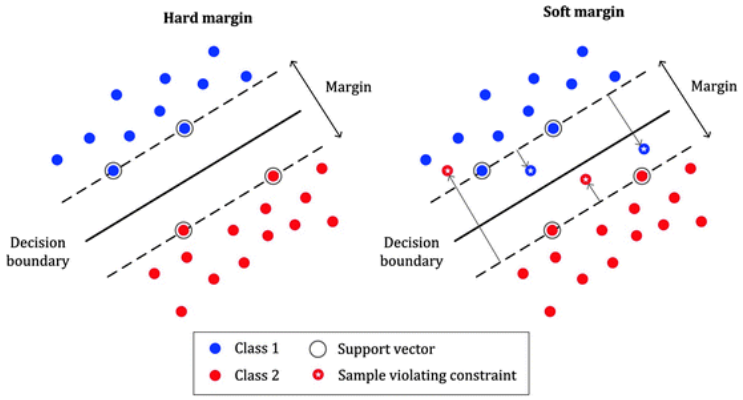
The distance is calculated by taking the absolute of the formula of the hyperplane and dividing this by the euclidean distance, which is the distance between two points in a vector space. In our case, it is the length of the weight vector. The goal of a SVM is to minimize the margin between the decision boundary and the support vectors. The equation for this is:

$$w* = arg_w max[min_n d_H(\phi(x_0))] \tag{3}$$

Here **arg max** is an operation that finds the argument that gives the maximum value from a function. It finds the maximum margin by finding the minimum distance between the data points and the hyperplane.

It can be checked whether the data points are divided in the correct class by the hyperplane. This can be done by filling in the data points in the hyperplane equation. The product of a predicted and actual label would be greater than 0 on the correct prediction. Else it would be less than 0. Figure 4 shows two examples of a SVM in a 2-dimensional space. A soft margin allows some misclassification to happen by softening the constraints of the SVM. It is used once data is not linearly separable, like the ePod data, because the data can't be perfectly divided with a hyperplane.

Figure 4: Example of a 2 dimensional hyperplane [30]



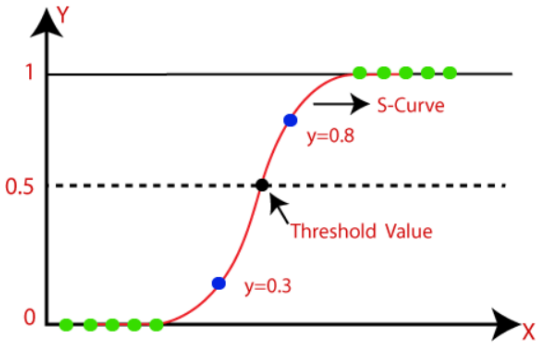
2.4.2 Logistic regression

Logistic regression is used to predict a categorical dependent variable using a set of independent variables. It gives the probability of a value between 0 and 1. In figure 5 a logistic regression function is shown. Logistic Regression uses a sigmoid function, which has an S-shape. This function goes from $-\infty$ to $+\infty$ to avoid probabilities below 0 and above 1. The equation of a logistic regression function is:

$$log[\frac{y}{1-y}] = b_0 + b_1x_1 + b_2x_2 + b_3x_3 + \cdots + b_nx_n \tag{4}$$

For logistic regression, the equation is obtained from the linear regression equation. This shows in the right part of the equation, which is the equation of a straight line. Here the **b** are the coefficients and **x** are the independent variables. In logistic regression, the dependent variable can only be between 0 and 1 in a range between $-\infty$ to $+\infty$. This is where the left part of the equation comes from. $\frac{y}{1-y}$ is 0 for $\mathbf{y = 0}$ and goes to infinity for $\mathbf{y = 0}$. The logarithm is taken to get the infinite range. The standard default of the threshold of the algorithm equals 0.5. For this research for example, once the outcome is above 0.5 the child will be at risk of developmental dyslexia. If it is below 0.5 the child belongs to the control group.

Figure 5: Example of a logistic regression function [31]



2.4.3 Decision Trees

Decision trees use simple decision rules to predict the class. A decision tree starts on the root node and will be divided into multiple sub-nodes. The decision tree splits the nodes on all available variables and then selects the split which results in the most homogeneous sub-nodes. There are multiple algorithms used in decision trees. The one used for this project is a Chi-square automatic interaction detection (CHAID) that performs multi-level splits. This is the underlying algorithm of the decision tree model from the scikit-learn library [32]. CHAID finds a significant difference between two sub-nodes and the parent node. It is measured by the sum of squares between the observed and expected values of the target variable. The formula for Chi-square is:

$$\chi^2 = \sum \frac{(O - E)^2}{E} \tag{5}$$

In this formula, **O** is the observed score and **E** is the expected score. The Chi-square will be calculated for each feature of the data set. The feature having the highest Chi-square will be the decision point. An example of a decision tree can be found in appendix C.

2.4.4 Multi-Layer Perceptron

A multi-layer perceptron (MLP) is a fully connected feedforward neural network. It consists of an input layer, an output layer, and hidden layers. Each node in the layers represents a weight, which will eventually map the output correctly. A neural network uses backpropagation to update the weights to make the network able to learn good internal representation. This means that the right nodes in a neural network should be activated when they have a positive influence on the performance of the network. A node's activation in a neural network is dependent on the incoming weights and bias term. The goal of backpropagation is to optimize the weights for each hidden layer node, so the neural network can learn how to map inputs and outputs correctly. Hidden

layer nodes don't have a target output, which means there is no error function for a single specific node. This means that the error for that node is dependent on the values of the parameters in previous layers (which are the input for that specific node), but also the following layers (because the output of the node affects these layers). Calculating the effect of each node compared to the other nodes can be complicated with a long calculation time. Backpropagation simplifies the mathematics of gradient descent between those layers and is more efficient. Backpropagation uses gradient descent to calculate the gradient of the error function concerning the neural network's weights. A gradient measures how much the output changes when the inputs change. Gradient descent is a minimization algorithm that minimizes the cost function. Backpropagation calculates the gradient backward through the network, in a way that the error of the output can adapt the weights of the nodes in the network. First, the gradient of the weights in the final layer will be calculated. The second step is to calculate the gradient of the weights of the second-last layer. As said earlier, layers affect each other and therefore the last layer is dependent on the second-last layer. More specifically, the weights are dependent on the weights and output of the previous layer. To solve this dependency for the gradient, the chain rule is used because there is a function within a function. This comes down to the fact that partial computations of the gradient from one layer are reused in the computation of the gradient from the previous layer. This backward pass of the error will continue until calculating the gradient of the weights of the first layer. Backpropagation runs in a cycle with the feedforward network. After the forward pass for each training example (or a batch of examples to speed up the process), a backward pass will be done to adjust the weights based on the error. In short, the backpropagation algorithm is efficient because the flow of error information goes through each layer, instead of calculating the gradient for each layer separately, which makes it hard to include the dependencies between layers.

An activation function is used to decide whether a node will be activated or not. It determines the importance of the nodes in the process of predicting the outcome. Commonly used activation functions for a MLP are: *identity, logistic, tanh and relu.*

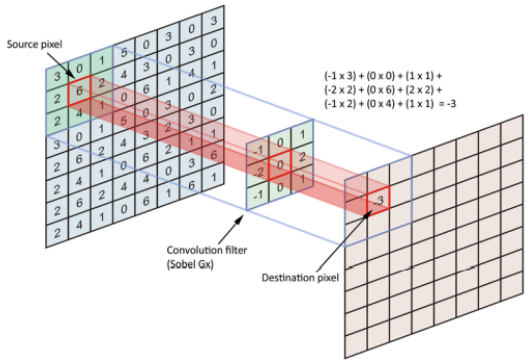
Table 1: Activation functions for a neural network

Activation	Function
Identity	$f(x) = x$
Sigmoid	$f(x) = \frac{1}{1+e^{-}x}$
Tanh	$f(x) = \frac{e^{x}-e^{-}x}{e^{x}-e^{-}x}$
Relu	$f(x) = max(0, x)$

2.4.5 Convolutional Neural Network

A convolutional neural network (CNN) is a learning network that transforms or extracts features using multiple non-linear processing units arranged in hierarchical multiple layers with different levels of representation and abstraction. Each layer of a CNN consists of a filter, an activation, pooling, and normalization. The filter of a layer looks for a pattern in the neighboring data points, with matrix multiplication a match between the filter and a small part of the input image is given using the dot product. Figure 6 shows an example of the calculation of a single filter in image classification. The output activations of a given filter are called feature maps.

Figure 6: Filter of a CNN [33]



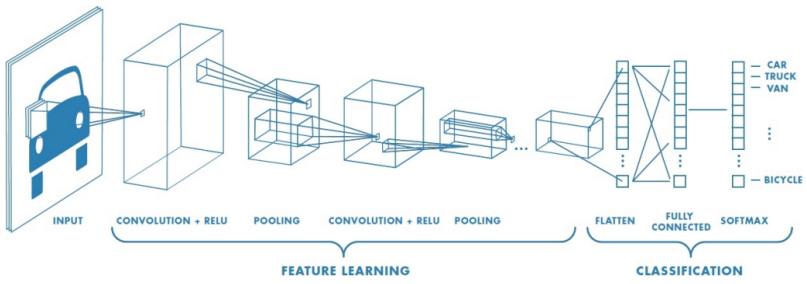
The next step is introducing a non-linear activation function. The goal of this operation is to only activate the output feature map if its value reaches a certain threshold. The most common activation function in deep neural networks is ReLU. The formula of ReLU can be found in table 2. For input values in the feature map below zero, the output of the operation will be zero. For input values above zero, the output equals the input. This introduces a simple non-linearity around zero.

Pooling down samples of the units for efficiency. The filter creates multiple feature maps and in the next layer, those maps will turn into more feature maps. To avoid an explosion of computational load, it's important to reduce the size of these maps by using pooling. Max pooling is the most used pooling method and chose the highest value of a unit from the feature map to reduce the shape.

The last step is normalizing the data. The threshold and pool operations use max functions. As a result, even if the convolution filter has a mean of zero, by the pool stage the mean activation is above zero and an arbitrary range. The normalization operation scales the data linearly to have a mean of zero for each feature map. The mean response of each feature map will be subtracted from all responses to zero-center the data. The next step is to divide the result

by the standard deviation. The normalized data now have a mean of zero and a standard deviation of one. This is useful for creating identical distributions so each feature map contributes similarly to classification. Those steps will be repeated for the number of layers in the CNN. A schematic overview can be found in figure 7. The data for this research consists out of extracted features from the MMN instead of images. However, the fundamental principles of the neural network remains the same.

Figure 7: Schematic view of a CNN in image classification [34]



3 Related Literature

3.1 Predicting disorders with EEG

EEG has become useful in diagnosing brain disorders. So far using EEG has been successful in diagnosing epilepsy, Alzheimer's disease, autism, and other conditions that affect the brain. Al Zoubi and colleagues conducted a study to predict the brain age gap, the difference between the estimated age and the chronological age of an individual, using EEG signals [35]. They used a nested-cross-validation approach combined with a set of regression algorithms, such as Random Forest and Support Vector Regression. The framework has a reliable estimation of chronological age and brain age. Another study used EEG signals to predict epileptic seizures. They used an algorithm to detect the spikes in an EEG signal during interictal, preictal, and ictal periods followed by a mean filter to smooth the spike number. The maximum spike rate of the interictal state was used as an indicator (threshold) to predict seizures. Once the signal passed the threshold, the signal would indicate a seizure. This approach reached a 92% accuracy [36]. Besthorn and colleagues used the delta and theta waves to predict Alzheimer's disease. They found out that there is an increase in the delta and theta power and a decrease in the alpha and beta power in Alzheimer's patients. Four different methods were used: classification by group means, discriminant analysis, neural network, and discriminant analysis combined with principal component analysis. They reached a maximum of 86.6% accuracy and 95.9% when they included age as variable [37]. EEG data has also been very useful in classifying different emotional states. A study showed that using a smoothing algorithm can improve emotion classification performance by SVM. Also, dimension reduction showed improved performance, by for example using principal component analysis or correlation based feature selection methods. The highest accuracy obtained was 91.77% by using linear discriminant analysis smoothing and correlation based feature reduction. A final insight was that emotion was mainly produced in a specific lobe of the brain [38]. Another interesting paper is the paper of Gibbon and colleagues. They researched classifying neural responses to rhythmic speech versus non-speech in infants. Their goal was to see whether classification with a rhythmic stimulus is possible since neural tracking of rhythm is atypical in children with developmental language disorders. Results show that both CNN and SVM can be reliably used on EEG data to classify the sound as a drumbeat or a repeated syllable. The CNN seemed to be more robust to the noisy EEG data [39]. There also have been multiple research done on predicting dyslexia using EEG data. Perera and colleagues wrote an extensive review of different classification frameworks for dyslexia, based on previous studies. They concluded that for predicting dyslexia, each class should contain at least 15 participants. It is also important to compare the differences in signals between males and females. Three different models are recommended, linear discriminant analysis, neural networks, and SVM, where SVM is called 'the classifier' to be used in EEG-based classification for dyslexia. Perera and colleagues also made a comparison

between popular EEG channels used in research. Those channels can be seen in figure 8 [40]. The bottom row is a summary of the most commonly used channels from the different contemplated research. The most commonly used channels are the channels in the frontal lobe in the left hemisphere. The sensors from the frontal lobe of the right hemisphere are also commonly used, the same as the sensors located on the axis of the brain.

Figure 8: Popular choice of EEG channels from [40]

Research	Number of channels	Channels
Different brain activation patterns in dyslexic children: evidence from EEG power and coherence patterns for the double-deficit theory of dyslexia	28	Fp1, Fp2, F7, F3, Fz, F4, F8, FC3, Fizz, FC4, T3, C3, Cz, C4, T4, CP3, Caps, CP4, T5, P3, PHz, P4, T6, O1, Oz, O2
Wavelet entropy differentiations of event-related potentials in dyslexia	15	Fp1, F3, C5, C3, Fp2, F4, C6, C4, O1, O2, P4, P3, PHz, Cz, Fz.
Detecting complexity abnormalities in dyslexia measuring approximate entropy of electroencephalographic signals	15	Fp1, F3, C5, C3, Fp2, F4, C6, C4, O1, O2, P4, P3, PHz, Cz, Fz.
Comparison between characteristics of EEG signal generated from dyslexic and normal children	4	C3, C4, P3, P4
An SVM-based algorithm for analysis and discrimination of dyslexic readers from regular readers using ERPs	64	F3, F4, P6, PHz, F8, CP4, AF7, F3, F5, T7, PO3, FC6, TP7, P7 (not all are given)
Classification of dyslexic and normal children during resting condition using KDE and MLP	8	F3, F4, C2, C3, C4, P3, P4, T3, T4
Wavelet packet analysis of EEG signals from children during writing	4	C3, C4, P3, P4
Popular EEG channels for identifying unique brainwave patterns for dyslexia	Fp1, F3, Fz, F4, F7, F8, T3, C3, Cz, C4, T4, PHz, AF3, TP7, P7	

3.2 Mismatch Negativity

Mismatch negativity is the response of the brain after an abnormality in a sequence of sensory stimuli [7]. The presentation of a deviant event embedded in a stream of repeated standard events results in an evoked response recorded with an EEG. Subtracting the response of the standard event from the deviant response results in a negative waveform, which is the MMN. The highest difference is after 100-250ms onset and the strongest intensity is in the temporal and frontal areas of the scalp [41]. A lot of research has been done on the event-related potential in clinical applications. Umbricht and colleagues studied the relation of mismatch negativity with schizophrenia. They found that patients with schizophrenia have a significantly smaller mean mismatch negativity compared to healthy participants [42]. The study of Baldeweg and

colleagues examined mismatch negativity in dyslexic subjects. They hypothesized that dyslexic subjects are impaired in auditory frequency discrimination. This hypothesis was tested by using an auditory brain potential to measure the mismatch negativity on 10 dyslexic and matched control subjects. The results showed that the mismatch potentials to changes in tone frequency were abnormal in the dyslexic subjects. This difference was not found in tone duration [43].

Analyzing the mismatch for each subject hand by hand can be time consuming and is also dependent on interpretation. Armanfard and colleagues used machine learning to detect if the mismatch negativity is present in the averaged event related potentials. The existence of the mismatch negativity in a coma patient showed a correlation with coma emergence. They tested an auditory odd-ball paradigm on 22 healthy subjects and 2 coma subjects. The used classification model reached an accuracy of 92.7% [44].

3.3 Connectivity

Connectivity between the EEG sensors gives information on the dynamic interactions of segregated brain regions. It is an estimation of the relation between brain areas. EEG features, which are used as input for a machine learning model, are determined by neurophysiological processes. The features are most of the time selected by algorithms as principal component analysis and linear discriminant analysis. Those approaches do not include the origin of the analyzed data, which are the characteristics of neurophysiological processes in either time-frequency or spatiotemporal domains. Research showed that these properties reduced the number of input signals from 31 to 8 and can achieve up to 90% accuracy [45]. Hramov and colleagues used brain connectivity as a feature reduction method. Their approach was to calculate the connectivity between the different sensors on the different frequency waves and use the sensors with the most connections as raw input for the model [45]. Another study used brain connectivity to detect Alzheimer's disease. Alzheimer's patients have a dramatic global cognitive decline, where their brains exhibit abnormal patterns of functional activity. A distinct connectivity pattern between Alzheimer's patients and non-Alzheimer's has been found based on the strength of connections between lobes[46]. Another study used brain connectivity as input to identify autism using machine learning. The SVM had the functional connectivity z-scores for all pairs of the region of interests as input along with other features for example causal path weights between the region of interests. Results show that causal connectivity path weights had the highest predictive power using support vector machine classification [47]. Martinez-Murcia and colleagues did research on differences in connectivity in the brain to detect dyslexia in participants. Temporal and spectral inter-channel EEG connectivity was estimated together with a denoising auto-encoder to learn the representation of the connectivity matrices. They reached an accuracy of around 0.7 and found a connection between the sensors on the temporal lobe and increased connectivity of the F7 electrode (located in Broca's area) [48].

4 Methods

The experiment protocol of the ePod dataset is based on the mismatch negativity. The hypothesis is that the MMN is more negative for the control group. It is expected that healthy participants in the control group will show a bigger difference in response between the standard and deviant stimuli. For this research, the input for the models will therefore be the theoretical approach of the average of epochs from the individual channels, which translates to the mismatch negativity. In this chapter, an epoch refers to a specific time window extracted from a continuous EEG signal. This is different from the term epochs used in neural networks. Building upon the mismatch response theory, three different approaches are implemented, which results in four approaches in total. The approaches are:

- Approach 1: Baseline
- Approach 2: Literature
- Approach 3: T-test
- Approach 4: Connectivity

The baseline approach uses the MMN of all channels used in the experiment. The other three approaches use different methods to select channels. By selecting specific channels as input, the model reduces complexity since it has fewer features to learn from. If the selected channels contain valuable information for the dependent variable it can increase model performance. The second approach is based upon the literature of section 3.1. The most popular electrodes in related research will be used as input features. Another approach for selecting important sensors is consulting our dataset, by looking at significant differences between the at risk and control groups. Research shows that connectivity can be used as feature reduction since the channels with high connectivity capture information about nearby channels. Those three theories; literature, t-testing, and connectivity will be compared to a baseline feature set consisting of all sensors.

This research is a binary classification problem since the dependent feature will be whether the participants are at risk or in the control group. Supervised machine learning models have a great history with classification problems. The supervised machine learning models chosen for this problem due to their performance with classification are: SVM, logistic regression, decision trees, MLP, and CNN. Motivation for the chosen models can be found in 4.3.

This chapter starts with the experimental protocol to understand the motivation behind the used ePod dataset. Then an extensive description will be given of how the data is processed to a useful input for the machine learning models. This chapter ends with a short discussion of the performance measures to validate the models.

4.1 Data

4.1.1 Experiment protocol

The ePODIUM project has gathered data over the past couple of months to create a dataset that can be of use for predicting dyslexia at an early age. They collected EEG data from infants aged 15 to 24 months. Every child got tested twice, respectively test A and test B. During test A the children's age varies between 15 to 21 months. For test B, the children's age is between 21 and 24 months. The aim is to have ideally 3 months between both tests. Both tests have the same setup, only the moment of testing is different.

During the test, the child is presented with a sequence of sound syllables, where occasionally an odd sound occurs. This is called an oddball paradigm for eliciting mismatch negativity. In an oddball paradigm, participants hear a sequence of standard stimuli, which randomly gets interrupted by a deviant stimulus. After the deviant, the standard stimulus continues. If a participant can discriminate between the two stimuli, different brain responses are expected. The average deviant minus the average standard represents the MMN, which gives information on how much the two stimuli differ from each other. This brain activity is measured by 32 sensors on the scalp. There are four conditions during tests A and B, those conditions are:

- Standard sound "giep" and deviant sound "gip" with 12 different pronunciations for both.
- Standard sound is "giep" and deviant sound "gip" with a single pronunciation.
- Standard sound is "gop" and deviant sound "goep" with 12 different pronunciations for both.
- Standard sound is "gop" and deviant sound "goep" with a single pronunciation.

The test results are processed in three different stimulus types. Those are standard 1, standard 2 and the deviant. Standard 1 is the stimuli short after the deviant, where the participant might not be adjusted yet to this sound being one of the standards. Therefore those stimuli are excluded when analyzing. In total there are 78 different sounds. For the multiple pronunciations, we have 12 x 3 = 36 and for the single pronunciations, we have 3 x 1 = 3. So for the four conditions, we have 36 + 3 + 36 + 3 = 78 sounds.

During the experiment, data is also collected on the characteristics of the children (e.g. sex and age), whether the parents are dyslexic or not and vocabulary knowledge of the child based on MacArthur Communicative Development Inventories (CDI).

4.1.2 ePodium data

The metadata collected during the experiment consists of four different text files: *cdi.txt*, *children.txt*, *codes_overview.txt* and *parents.txt*. The cdi text file contains, besides the basic information about the children, information on their vocabulary size. The vocabulary is split into different subcategories, for example, animal names, toys, food, games, and verbs. The features are all integers showing the count of words. The children's file consists out of the children's age during both test A and test B, their gender, and whether they are at risk for dyslexia. The parent's text file has information on how the parents score on different dyslexia tests. It also contains concluding variables *dyslexia_mother_accToMother* and *dyslexia_father_accToFather*. The final text file *codes_overview.txt* is a dictionary to see which event belongs to a specific key.

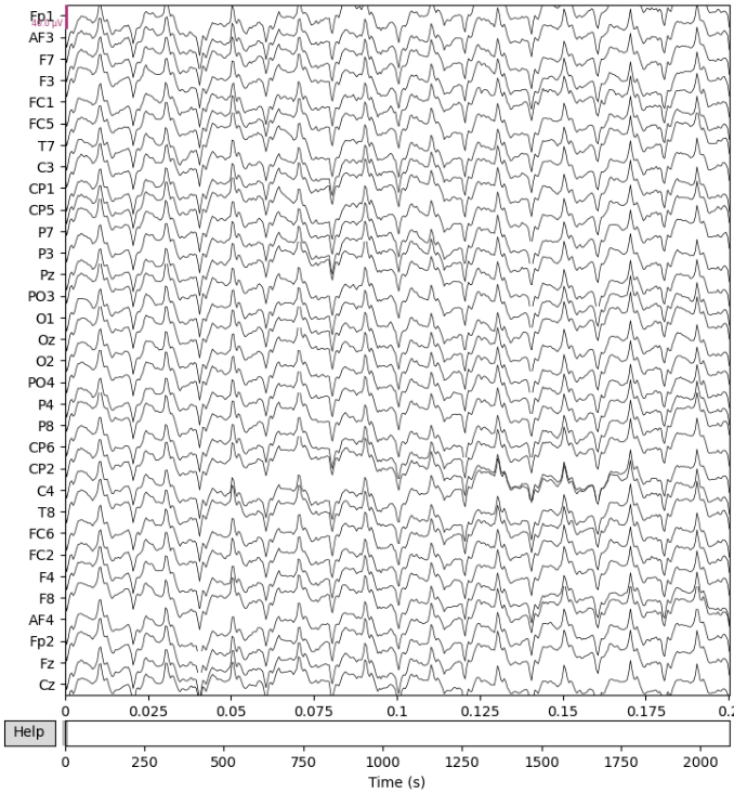
Figure 9: Information of a single bdf file

Measurement date	April 07, 2021 09:33:56 GMT
Experimenter	Unknown
Participant	Unknown
Digitized points	Not available
Good channels	40 EEG, 1 Stimulus
Bad channels	None
EOG channels	Not available
ECG channels	Not available
Sampling frequency	2048.00 Hz
Highpass	0.00 Hz
Lowpass	417.00 Hz
Filenames	105a.bdf
Duration	00:34:51 (HH:MM:SS)

There are 248 different bdf files besides the text files. A total of 129 children participated in the experiment and 22 of them didn't come back for test B. The bdf files are the recorded EEG data of a single test. An example of a bdf file can be found in figure 9. The bdf has information on the measurement date, the amount of good and bad channels, the sampling frequency, the lowpass and highpass, the name of the file, and the duration of the experiment. The sampling frequency is the number of samples per second. For our data, the sampling frequency is 2048 Hz, which means that there are 2048 data points for each second. The high and lowpass signify that all signals were measured between 0 to 417 Hz. In figure 10 the EEG recording of the same participant is

shown. On the y-axis, the 32 EEG channels for this experiment are shown. The 8 missing channels which were shown in the bdf file description are the reference channels. On the x-axis, we see the 2048 data points per second and the time itself in seconds (s).

Figure 10: EEG recording



Each bdf file has a corresponding event file. There are 248 text files containing the key of an event and the time when the event occurred. The number of distinctive events is 78, but can be reduced to 12 events by combining the different pronunciations. This has been done since the pronunciations only differ from each other acoustically and not phonologically. Research showed that difficulties in reading typically result from a deficit in the phonological component of language, as already mentioned in paragraph 2.1 [2]. Since the pronunciations do not differ phonologically, they can be combined. The events are respectively:

- GiepM_FS : 1
- GiepM_S: 2
- GiepM_D: 3

- GiepS_FS: 4
- GiepS_S: 5
- GiepS_D: 6
- GopM_FS: 7
- GopM_S: 8
- GopM_D: 9
- GopS_FS: 10
- GopS_S: 11
- GopS_D: 12

'M' stands for multiple pronunciations and 'S' for single pronunciations. '_FS' is the first standard, '_S' for standard event and '_D' stands for the deviant event.

4.2 EEG preprocessing

Before the EEG data was fed to the models, multiple data preparation steps were necessary. An additional library developed by MNE is used to perform the data preparation, such as filtering or visualizing to understand and analyze the data. The MNE library is an open-source python package for exploring, visualizing, and analyzing human neurophysiological data [49]. All mentioned custom-made functions below are added to the eegyolk library, a library made for analyzing EEG data [50].

The first step was loading the data into jupyter lab, a web-based interactive environment running on python for notebooks, code, and data [51]. To load all the raw EEG files, a function has been created called *load_dataset*. This function takes as input the folder where all the files are stored and the file extension type, which is bdf in our case. The function loops over all the files in the folder and uses the function *mne.io.read_raw_bdf* to read the bdf file. All the files are stored in a list. The name of the file, indicating the participant ID, is stored in a separate list. For loading the events a different function has been created, called *load_events*. This function has as input the folder where the event text files are stored and the list of the EEG filenames. There is a loop in the function to go over all filenames and to load the corresponding event files into a list with all events. In section 4.1.2 we talked about reducing the events to only 12 distinct events. This is done with the function *group_events_12*. For the metadata files, a separate loader is created to load the *cdi.txt*, *children.txt*, *codes_overview.txt* and *parents.txt* files, which are also stored in a list.

The next step is filtering the raw EEG data. It is better to filter before cutting the signal into small segments to accurately estimate and remove low and high signals [52]. The filtering is done by the self-created function *filter_eeg_raw*. This function needs as input a single bdf file, a lowpass value, a highpass value,

the mastoid channels and if needed, some channel names to drop. Multiple functions are inside the *filter_eeg_raw* function. The first one is the bandpass filter from the mne library. A bandpass filter is applied to filter out slow frequencies with a high-pass filter and high frequencies with a low-pass filter. The low-pass filter is set to 40 Hz since research shows that this frequency records accurate values [52] [53]. Tanner and colleagues did research on an optimal high-pass filter. They found that cutoffs above 0.3 Hz produced artifactual effects. The frequency cutoff creates a negative peak which is an artifact. Cutoffs at 0.01 Hz and 0.1 Hz do not show those effects, those waveforms are nearly identical to the unfiltered waveform [54]. Therefore the high-pass filter has been set to 0.1 Hz for this research. Another frequency that has to be filtered out is the power line noise. Power line noise is created by the flow of current between two conductors in a gap and is at 50 Hz. This is mostly caused by broken, improperly installed, or loose hardware. These artifacts should be suppressed to allow proper analysis [55]. The *mne.filter.notch_filter* is used for noise removal. The next preprocessing step is subtracting the reference from the EEG signals. We want that each measurement electrode only contains information on the changes in brain fluctuations after a certain stimulus. We don't want environmental noise that is being picked up by the measurement electrodes. Therefore, there are two reference channels, called mastoid channels, which are placed near the ears to pick up environmental noise but don't pick up too many brain signals. The signals are subtracted from the measurement electrodes to only keep the brain fluctuations caused by the controlled stimulus [24]. The two mastoids channels in the ePod dataset are *['EXG1', 'EXG2']* and are subtracted by using the function *mne.set_eeg_reference* from the mne library. The mne library also provides a function, *mne.Info.set_montage* to map the EEG electrodes to the right position. For this research, we used the standard 10-20 montage. Using a montage to standardize the EEG electrode placement, ensures that inter-electrode spacing is equal and the electrode placements become proportional to the skull size [56]. Finally, *filter_eeg_raw* uses the two functions of mne to drop self-selected channels (*mne.io.drop_channels*) and to remove the bad channels marked by the *mne.info* method. The channels dropped are *['EXG1', 'EXG2','EXG3', 'EXG4', 'EXG5', 'EXG6', 'EXG7', 'EXG8', 'Status']*, since those channels are either used as reference channels or do not contain information on the stimulus-response.

The next filtering step is cutting the raw EEG data into epochs. This is done in the self-build function *create_epoch* EEG epoching is a procedure to extract specific time windows from the raw continuous EEG signal. Creating epochs helps with interpreting the response to a specific event, by selecting the time window from right before the event to shortly after the event. For this research, the time window is from -0.2s to 0.8s. The average time calculated between two events for the ePod dataset is 0.8s. Choosing -0.2 indicates the EEG waveform before stimuli onset. The waveform has time to stabilize until 0.8s when the next stimulus is presented. This MMN occurs after 150-250ms after the stimuli onset. Each sound file is around 300ms, but since the experiment uses natural

stimuli, where the vowel is already integrated into the consonant before, the MMN can already be seen before the end of the audio. The epochs are created with the mne function *mne.Epochs*, which needs as input the raw EEG file, the events, and the time window of an epoch. This function automatically detrends the data. Also, an auto-reject function is used to automatically reject bad trials and repair bad sensors. The auto-reject function creates a threshold by splitting the data into multiple segments and calculates the mean of the signal of good trials in each set. Then it calculates the median of all trials in the test set and calculates the error between the train and test set of that set. The set with the lowest error will be the rejection threshold for global rejection [57]. All the cleaned epochs per participant per test are saved in a new folder.

To easily find the desired data, a pandas data frame has been created containing all the necessary metadata, for example, the child's age, gender, and whether it is data from test A or B. This data is merged with the corresponding data paths for the raw EEG data, the events, and the cleaned epochs. The data frame is saved in the file called *metadata.csv*. A separate function has been created to easily load in the cleaned data from the csv file, called *read_filtered_data*.

For this research, only the data for test A is used. The motivation behind this is that research shows that infants begin to engage in long-term memory at 8 months old [58]. The children may be different on test B since they have already been exposed to the test before. To rule out this possibility only test B is used. This has however not been tested since it is out of scope for this research.

The bdf files have a large size, which resulted in memory issues. Therefore, this project has been done in a CUDA workspace running on a GPU Base (A10) with Ubuntu 20.04. Even though this workspace has a high memory capacity, there still were issues with memory since we are talking about processing 304GB. Therefore filtering the raw EEG data uses a generator, which first checks if the file is already in the cleaned epoch folder. If this is not the case, the generator loads the data of a single EEG file and runs the *filter_eeg_raw* and the *create_epoch* functions and saves the cleaned file. The generator clears the memory in the terminal and repeats the process.

4.2.1 Mismatch Negativity input

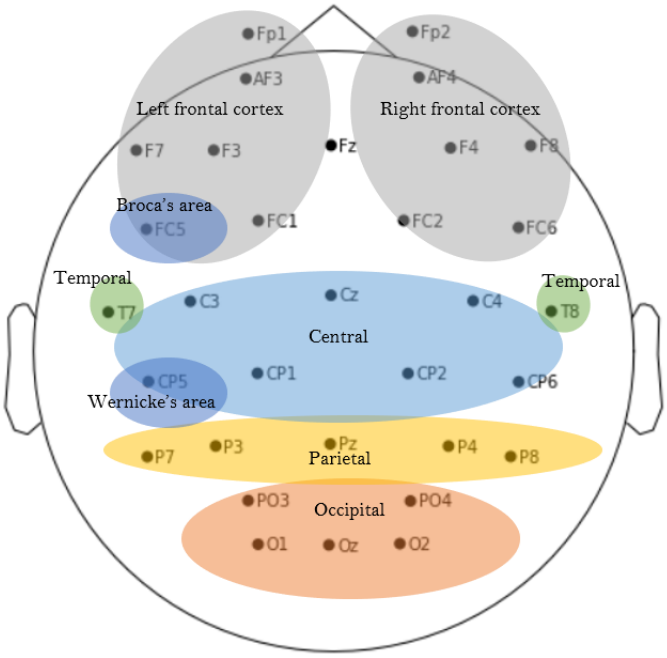
The input for the model will be the MMN of the participants since the experiment is built on this theory. The custom-made function *input_mmr_prep* calculates the mismatch negativity from each electrode. The functions need as input the *metadata.csv* file, a list with the channels (electrodes) of interest, and the event names. For each file, the event-related potential will be calculated, by taking the average of all epochs for a specific event per channel. This is done for both the standard and deviant events. Then the array of the standard event will be subtracted from the array of the deviant event, which results in a new array containing the difference between the two events. This array has the shape

(32, 2049), which are the 32 channels and 2049 data points. For each channel, features will be calculated to get an idea of the shape of the line. Those features are the surface, standard deviation, minimum, and maximum. The mean isn't calculated because it correlates with the surface. Zero crossings are also not included. Before onset the signals for both the standard and deviant aren't that different, causing a MMN near zero. Therefore zero crossings are inconclusive. The data is saved in a pandas data frame with each row containing a single file and the features of the sensor's MMN.

4.2.2 Feature selection

As mentioned at the beginning of this method section, the distinction of this research is the selection of channels. The baseline will be a feature set of all sensors. Using this as a baseline will indicate how a theory-based model is performing. The second feature set is based on the literature. The chosen channels are: *[Fp1, F3, Fz, F4, F8, T7, C3, Cz, C4, T4, AF3, P7]* based on the research of Perera and colleagues [40]. Note that T3 is replaced by T7 since T3 is not present in our data. T7 has the same position as T3, near the left ear [59]. Electrodes PHz and TP7 are also missing in our data and therefore not included in the feature set. Different t-tests will be done to see where which channels differ significantly between the at risk and the control group. This will be done only on the training data to avoid overfitting on the test data. The results of the t-tests and the corresponding selected channels can be found in section 5.3. The last feature set is based on the connectivity between the different sensors. As mentioned in section 3.3, Hramov and colleagues used connectivity between sensors as a feature reduction method with great results of 90% accuracy on EEG data [45]. The last approach is to calculate the connectivity between the sensors to reduce the number of features. Features that show strong connectivity to other sensors will be used as a final feature set. The results can be found in section 5.4. The location of the sensors and the corresponding brain areas can be found in figure 11.

Figure 11: Sensor placing on the scalp with corresponding brain areas



4.3 Machine learning

The algorithms that are for this research are SVM, logistic regression, decision tree, MLP, and CNN. Similar studies mentioned in section 3.1 on EEG data used those five models as well as some other models. The motivation for limiting this research to those models is because the objective is to see whether the MMN and the selection of channels influence predicting dyslexia. Additional to the good performance of those models on EEG data, they are also easy to implement due to numerous available libraries containing those models.

The input for those models is the four different datasets related to the four approaches of feature selection. A smaller dataset is used for the t-test and connectivity approach since a part of the data is used for an independent analysis. Before feeding the data directly to the models, the data has been divided into a training set and a test set. For this, the scikit-learn library is used. For each approach, 80% of the data belongs to the training set. The other 20% is part of the test set. This is a commonly used split in machine learning [60]. The next step is scaling the data. Scaling can boost a model's performance by reducing the chance of biases towards higher values in features. SVM, logistic regression,

decision tree, and MLP all use the standard scaler from the scikit-learn library. The CNN uses a normalizer from the tensorflow library to scale the data. A standard scaler operates on each column, so each value gets scaled on the values that are observed in all participants for that specific channel. The normalizer scales the data for each row, meaning that all features are scaled in respective of the values of the specific participant. Two different methods are used since the CNN takes a tensor as input and the other models take a data frame as input.

Each model can be trained with different hyperparameters, which control the learning process. Grid search is used to optimize the hyperparameters to calculate the performance of different combinations of parameters. The scikit-learn library contains a function for this. The hyperparameters for the SVM are the kernel, C, and gamma. The kernel is used to transform linearly inseparable data into linearly separable ones. The C gives a penalty for incorrect classifications and is often referred to as regularization term. Gamma controls the distance of the influence of a single training point. Besides having the C as a hyperparameter, logistic regression also has maximum iterations and a solver to tune. Different solvers have different approaches to minimizing the optimization problem. The solver needs a corresponding penalty for misclassifications, therefore the penalty is not part of the grid search. The decision tree algorithm has the hyperparameters criterion, splitter, and maximum depth to tune. The criterion controls how the impurity of the split of the node will be measured. The splitter hyperparameter decides whether a suboptimal split will be used or a random split. Maximum depth limits the size of the tree to maintain interpretability and overfitting. For the MLP algorithm grid search has been done on the hyperparameters activation, alpha, solver, learning rate, and maximum iterations. The activation hyperparameter decides which activation function will be used to activate the nodes in the neural network. Alpha is the strength of the regularization term. The learning rate controls the change in the coefficients for each iteration. The hyperparameter learning rate determines whether the learning rate changes over time or remains constant. The solver and maximum iterations have the same function as for logistic regression. For the CNN no hyperparameter tuning is used. The CNN is a simple model containing 3 dense layers, to avoid overfitting and to get a better insight if a model that can learn from the ePod data.

All models use k-fold cross-validation as an evaluation method to find out how well the model can predict the outcome of unseen data. The method divides the data into k-groups which will be used as a test set. The remaining data will be used for training the model. An 80/20 split is commonly used and since the remaining of the ePod dataset is 101 participants after filtering, 20 participants should be in the test set, which results in k=5 folds.

The accuracy is used as a performance measure as it gives a good insight into binary classification. For the CNN also the loss is calculated to see after how many errors the neural network makes over the epochs. The accuracy score

is the number of correct predictions and the loss values are the values indicating the difference from the desired target state.

4.4 Reproducibility

Reproducibility of work is highly important in scientific research since it serves as proof that an established and documented work can be verified, repeated, and reproduced. To make this research reproducible a library is created called eegyolk [50]. This is the same library as mentioned in section 4.2. This library contains all methods used for this research, including code used for previous research on age prediction. Specific instructions on how to reproduce this work can be found in the readme of the eegyolk repository. An environment file is created to list all dependencies of the used libraries to keep the versions of different libraries compatible. A configuration file is set up containing all data pathways to avoid changing the code in the notebooks. All work has been reviewed and tested on reproducibility.

5 Initial statistics

Before feeding all data to the models, some statistics have been done to get a better insight into the data. This will be a group-level analysis between the control group and at risk group. The differences in ERPs will be plotted. Also, significance tests will be done to see whether there is a significant difference in specific electrodes between the two groups. This paragraph ends with a connectivity analysis to see which sensors have the highest connectivity for each group.

5.1 Participant information

After cleaning and filtering the data only 101 distinct participants are left in the data set. This means that 28 children were dropped from the research due to missing events or bad signals. In total there are 58 children in the control group and 43 infants in the at risk group. In section 4.1.1 was already mentioned that the participants are between 15 to 24 months. Only the data for test A is used where the age varies between 15 to 21 months. The average age is 17.95 months for all infants. For the control group, the average is 17.88 months with a standard deviation of 1.44 months, and for the at risk group 18.05 months with a standard deviation of 1.50 months. In the control group, 33 participants are male, while 25 are female. For the at risk group 19 are male and 24 are female. This means that there are relatively more males (57%) in the control group compared to the at risk group (44%).

5.2 ERP group analysis

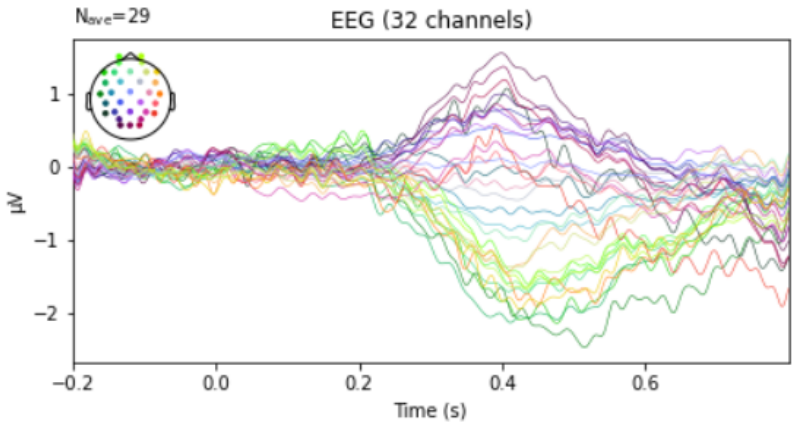
For the ERP group analysis, the MMN are averaged over all participants of either the at risk or control group. Also, the responses to the standard stimuli and the deviant stimuli are plotted separately to see if the difference in the mismatch can be explained by the standard response or the deviant response.

5.2.1 Mismatch Negativity

The MMN of both the control and the at risk group is calculated over all 4 events. This means that the standard responses *[GiepM_S, GiepS_S, GopM_S, GopS_S]* are averaged together and subtracted from the average deviant responses *[GiepM_D, GiepS_D, GopM_D, GopS_D]*. The first standards are excluded. High voltage fluctuations in the MMN indicate a bigger difference in the standard and the deviant event. If the voltages are near zero, the participant is less responsive to the deviant event. Expected is that the control group has a bigger difference between the standard and deviant event compared to the group at risk for dyslexia. In figure 12 the MMN can be seen for the 58 participants in the control group. The plot shows that the sensors in the occipital area of the brain have a peak a little bit above $1\mu V$ around 0.4s. The standard response is in this area higher compared to the deviant response. The

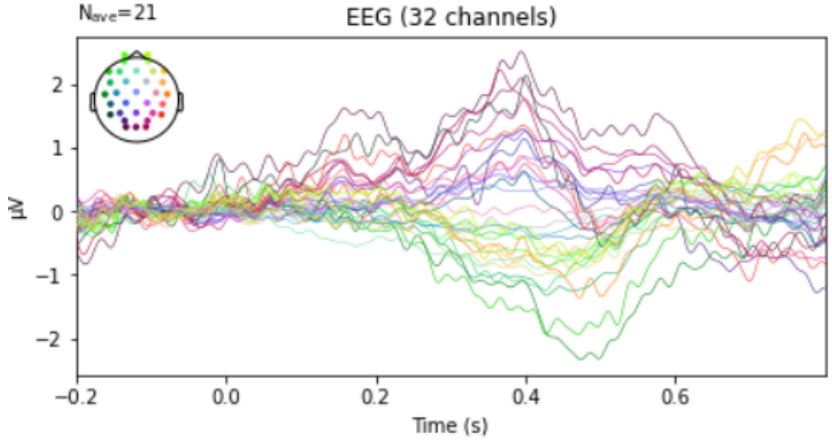
electrodes in the prefrontal cortex show a negative MMN, which indicates that there is a measured higher voltage in the deviant response.

Figure 12: Average MMN of the control group



For the group at risk for dyslexia, a similar pattern can be seen in figure 13. The occipital area has high voltages at time 0.4, while the prefrontal cortex shows negative voltages. There is also a small bump shown around 0.2s for some electrodes in the occipital and parietal area.

Figure 13: Average MMN of the at risk group



A noticeable difference between the two groups is the electrodes on the prefrontal cortex. The participants in the control group have a more negative

voltage in the electrodes placed in this area. The electrodes in the occipital area show a higher voltage in the at risk group. Expected was that the control group has higher voltages compared to the at risk group. This is only valid for the sensors in the prefrontal cortex. The electrodes in the occipital brain area show the opposite as hypothesized. Another difference is in the fluctuations before and right after the stimulus onset. The at risk group has higher voltage changes compared to the control group.

The MMN for each event per group can be found in appendix A and B. The gop single pronunciation shows the most expected pattern, which is a bigger response for the control group and a smaller response for the deviant group. This motivates the decision to use this stimulus as input for the models.

5.2.2 Standard stimuli

For the standard stimuli, all standard events are averaged, similar to the MMN group analysis. Different than the MMN, the peaks are shown at 0.2s after onset. For the control group, the occipital area shows a negative voltage, while the left prefrontal cortex shows higher voltages. The electrodes in the right prefrontal cortex seem to differ a lot in voltage. The ERP for the at risk group shows the same pattern. However, the electrodes in the occipital area have a more negative voltage after 0.4s. This can explain why the peak of those sensors in the MMN is higher for the at risk group at 0.4s.

Figure 14: Average standard response of the control group

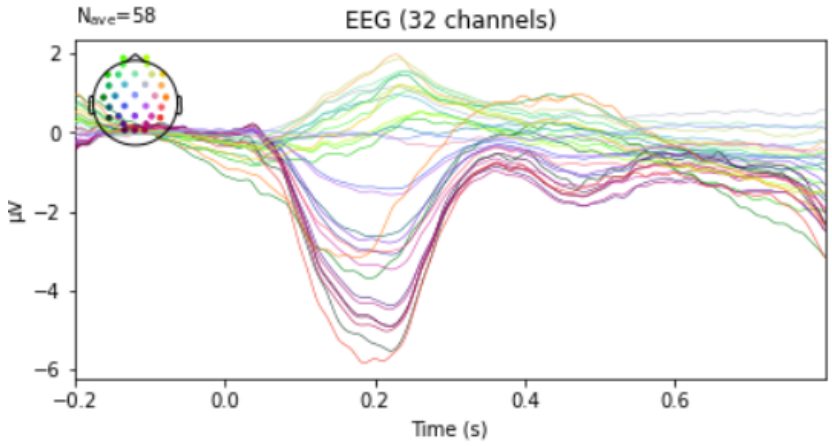
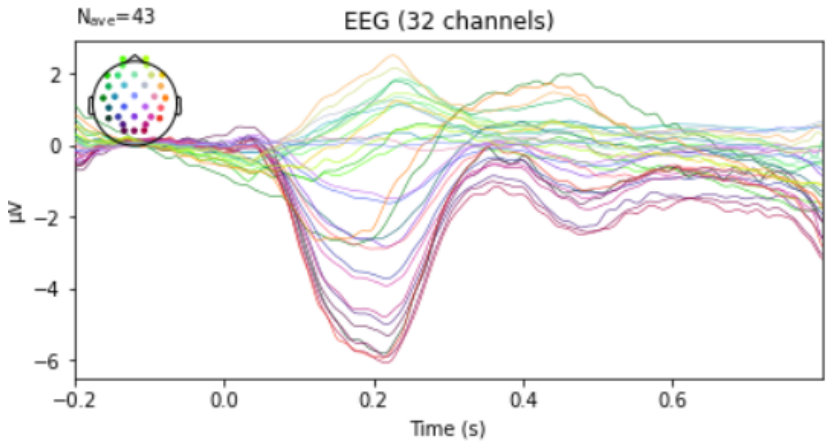


Figure 15: Average standard response of the at risk group



5.2.3 Deviant stimuli

The same process is applied for averaging the deviant stimuli as the standard stimuli. The ERPs for the control and at risk groups can be seen in figure 16 and 17 respectively. For the control group, the electrodes in the occipital area seemed to be more clustered, compared to the signals of the at risk group. Also, those electrodes have higher negative voltages for the at risk group. The response of the electrodes in the prefrontal cortex shows a similar pattern for both groups.

Figure 16: Average deviant response of the control group

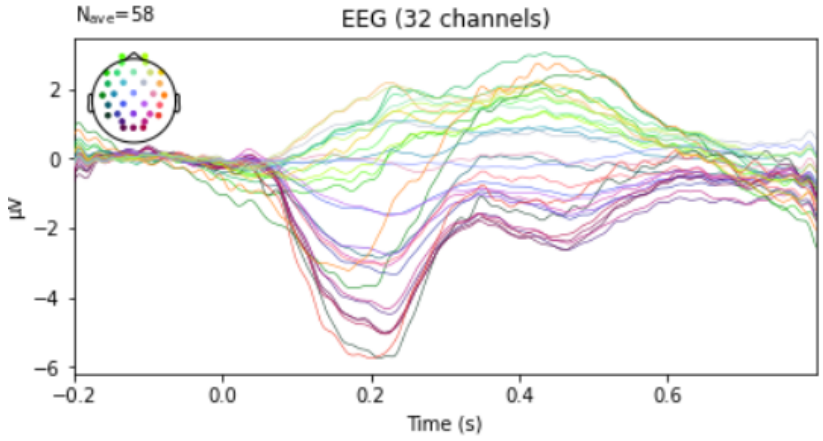
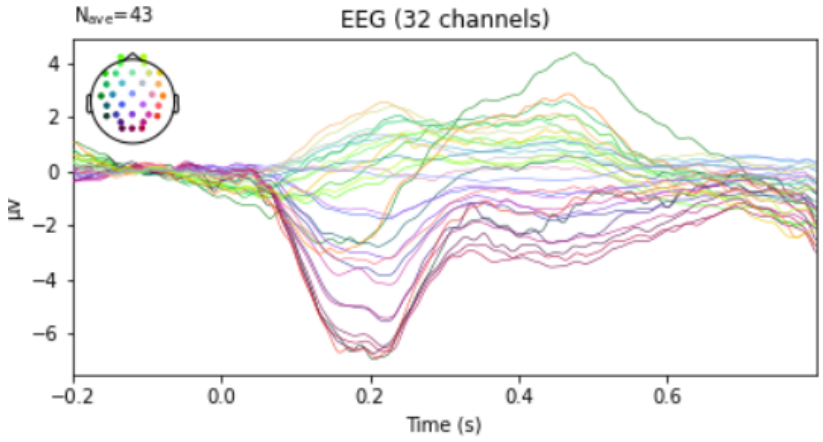


Figure 17: Average deviant response of the at risk group



5.3 Significance tests for approach 3

In the previous paragraph, some differences in electrode voltages in the ERP between the two groups were found. The electrodes that differ between the two groups can be a good indication of which electrodes to focus on when predicting dyslexia. However, if the same data as this analysis is fed into the model, the model might have a prejudice. It learns from the values which we know already influence dyslexia. To maintain the validity of the model, this part of the analysis will only be done on a small sample of the data, which will be excluded from the model training and validation. A 30% sample size has been chosen since research showed that for a small data set, 30% is necessary for the sample to be representative [61]. This results in a sample size of 30 participants, with 17 in the control group and 12 participants in the at risk group. This corresponds to the distribution of the whole dataset.

To check whether the differences in the MMN are significant, a simple t-test will be performed on each electrode for each group. Here, all the signals are taken individually for each participant. Since t-testing doesn't take a time series array, a single value had to be selected which captures the line of the ERP. The surface has been chosen as the value since most differences in ERPs were found in the voltage peaks. Due to the occurrence of differences between local minima and maxima, choosing the maximum wouldn't be a good indication. The t-test will be performed with the scipy stats library [62]. For the t-test, all standard and deviant events will be combined, similar to calculating the ERP of the MMN in section 5.2.1.

Table 2: T-test on group differences of the surface

Channel	t-value	p-value
Fp1	0.850	0.402
AF3	1.020	0.316
F7	1.268	0.215
F3	1.749	0.091
FC1	0.131	0.896
FC5	2.490	0.019
T7	0.867	0.393
C3	0.113	0.911
CP1	0.175	0.862
CP5	0.235	0.816
P7	0.445	0.660
P3	0.572	0.572
Pz	0.358	0.722
PO3	0.201	0.842
O1	0.412	0.684
Oz	0.786	0.438
O2	0.911	0.370
PO4	1.227	0.230
P4	1.017	0.317
P8	0.363	0.719
CP6	0.629	0.535
CP2	1.154	0.258
C4	0.627	0.536
T8	1.798	0.083
FC6	0.492	0.627
FC2	0.549	0.587
F4	0.127	0.900
F8	1.694	0.101
AF4	2.011	0.054
Fp2	0.710	0.484
Fz	0.270	0.789
Cz	0.567	0.575

The null hypothesis of this t-test will be that there is no difference between the two groups in each electrode. The hypothesis will be rejected if the p-value ¡ 0.1. Low p-values indicate that the data did not occur by chance. Since the EEG data is very noisy and diverse, the p-value is set a bit higher than usual to include more channels that are likely to differ between the two groups. The greater the t-value, the more likely that there is a difference between the two groups. The highest t-values are from electrodes *F3, FC5, T8, F8 and AF4*. All those channels satisfy p ¡ 0.1, which implies that the null hypothesis can be rejected and that the two groups are significantly different in those electrodes.

The electrodes will be used as input for the models based on their significant difference.

5.4 Connectivity tests for approach 4

As explained earlier, measuring connectivity between electrodes can be used as a feature reduction method. The mne library contains a function to calculate the connectivity between channels, called *mne.spectral_connectivity_epochs*. This function calculates the connectivity between epochs using Phase Lag Index (PLI). PLI is based on phase locking centered around 0 [63]. The electrodes with the highest connectivity with other electrodes will be chosen as model input. The same theory is applied as for the t-test input concerning the fact that the sample used for the analysis must be excluded from the model to guarantee a valid model. The same data sample as the previous paragraph 5.3 will be used. To calculate the group connectivity, all epochs are concatenated from all participants belonging to the sample of one of the groups.

The connectivity between the electrodes can be seen in 18. Most connections are formed between the electrodes in the prefrontal cortex. The electrodes with the most connections are *Fp2, Fp1, Fz, FC1 and AF4* from high to low. Channel *Fp1* has the strongest connections. Then the electrodes *Fz, Fp2 and FC1* show high connectivity. The connectivity for the at risk group is shown in figure 19. The electrodes with the most connections are *Fp2, Fp1, Fz, AF4, FC1 and AF3*, also all located in the frontal cortex. *Fz* shows the highest connectivity, followed by *Fp1 and Fp2*.

Figure 18: Connectivity plot of the people in the control group

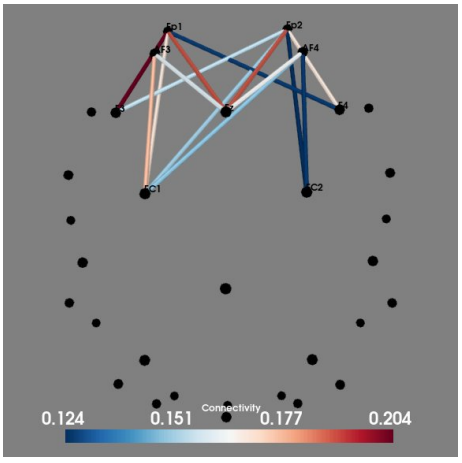
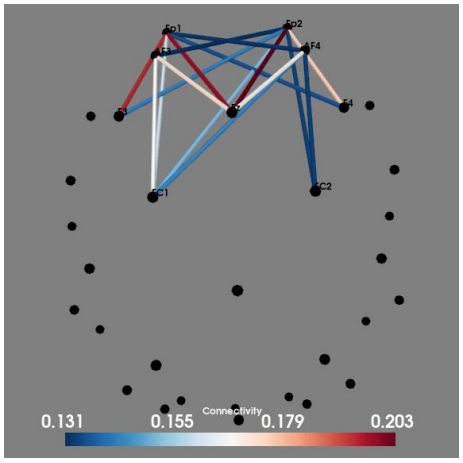


Figure 19: Connectivity plot of the people in the at risk group



For both groups, *Fp2, Fp1 and Fz* show the most connections and the highest connectivity. This means that those electrodes have a strong connection with other electrodes. A side note is that PLI ranges from 0 to 1, from which can be inferred that the connections of our data aren't that strong. However, those connections are the strongest and therefore will be used as input for the model.

6 Results

In total, 5 different models were used on 4 different feature sets as input data. The 4 different feature sets follow from the 4 different approaches to reduce the number of features. The approaches are the MMN of all channels as a baseline, a selection of channels based on related literature, selected channels based on a significant difference between the at risk and control group of a subset of the ePod data, and the channels that show the highest connectivity in the subset of the ePod data. The used models are a SVM, logistic regression, a decision tree, a MLP, and a CNN. The model input differs in the selection of channels. The results can be found in table 3, where the performance is measured in accuracy.

Table 3: Performance of different models

Model	baseline	literature	t-test	connectivity
SVM	0.573	0.524	0.605	0.633
Logistic regression	0.564	0.572	0.609	0.605
Decision Tree	0.564	0.545	0.421	0.521
MLP	0.593	0.495	0.549	0.607
CNN	0.667	0.571	0.733	0.682

CNN is the best performing model for all 4 approaches. Decision trees, however, do not seem like a suitable algorithm for the mismatch response approach. The literature approach has the worst performance. Connectivity seems to be a good approach to reducing the number of features, resulting in high accuracy. However, the accuracy in combination with the CNN is not significantly better compared to the accuracy of the baseline approach with CNN over the different folds with a p-value of 0.266. The t-test approach has the highest performance of all in combination with the CNN, however, the result is still not significant compared to the baseline with a p-value of 0.447. All results are not significant because of the high variation between the different folds of each model.

In this chapter, the epochs refer to the number of times that the learning algorithm will work through the entire training set. This is different compared to the epochs in the data preprocessing, where an epoch referred to a time window around an event in an EEG signal.

6.1 Approach 1: Baseline

The baseline input contains all channels, reference channels excluded, used in the experiment. The channels are *['Fp1', 'AF3', 'F7', 'F3', 'FC1', 'FC5', 'T7', 'C3', 'CP1', 'CP5', 'P7', 'P3', 'Pz', 'PO3', 'O1', 'Oz', 'O2', 'PO4', 'P4', 'P8', 'CP6', 'CP2', 'C4', 'T8', 'FC6', 'FC2', 'F4', 'F8', 'AF4', 'Fp2', 'Fz', 'Cz']*. Grid search is used to find the most optimal values for the hyperparameters. For the SVM model, the selected hyperparameters are *SVC(C=1000,*

gamma='auto', kernel='linear'). For the Logistic Regression model, the hyperparameters are *LogisticRegression(C=1000, solver='sag')*. The hyperparameters used for the Decision Tree are *DT(criterion='entropy', max_depth=2, splitter='random')*. The decision tree can be found in figure 36. For the neural network, MLP, grid search is also used to choose the hyperparameters. The hyperparameters are *MLPClassifier(activation='logistic', alpha=1e-05, learning_rate='invscaling', max_iter=5000, solver='lbfgs')*. The cross-validated accuracy after 5 folds is shown in table 5. For all four models, it can be seen that the performance for each fold varies a lot. This can bring the validity of the models in question.

Table 4: K-fold accuracy baseline input

Model	Accuracy	Mean	Standard Deviation
SVM	[0.67, 0.55, 0.65, 0.45, 0.55]	0.57	0.08
Logistic regression	[0.48, 0.60, 0.55, 0.60, 0.55]	0.56	0.04
Decision Tree	[0.52, 0.60, 0.65, 0.40, 0.65]	0.56	0.09
MLP	[0.62, 0.60, 0.50, 0.70, 0.55]	0.59	0.07
CNN	[0.62, 0.60, 0.60, 0.65, 0.60]	0.59	0.07

For the CNN no hyperparameter tuning is used. The CNN is a simple model containing 3 dense layers, to avoid overfitting and to get a better insight if a model can learn from the ePod data. It uses an adam optimizer and the loss is calculated with binary cross entropy. The training and the loss of each fold of the model can be found in appendix D. In figure 20 and figure 21 the results of the third fold are shown. There can be seen that the accuracy stabilizes around 0.60. The validation loss is higher compared to the training loss. The ePod dataset is a small dataset, which makes it harder for a model to learn patterns in the training data and can result in more errors. After 20 epochs the loss goes up, which indicates that the model gets worse in performance. The other folds also show an increasing loss between 10 to 25 epochs. Training the model on more epochs did not result in better model performance.

Figure 20: CNN accuracy on baseline input fold 3

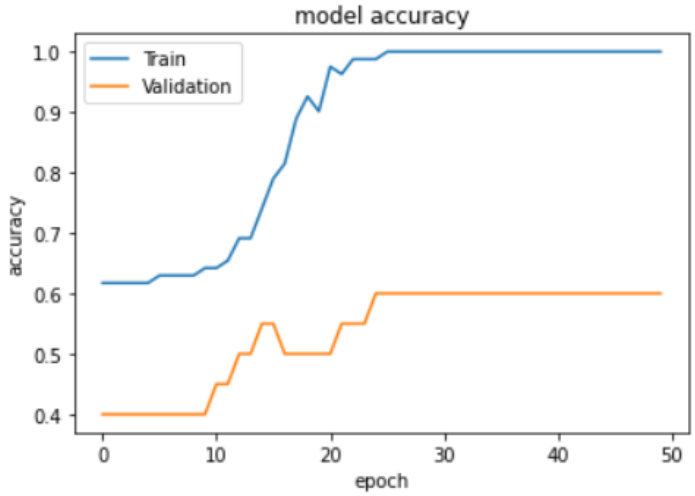
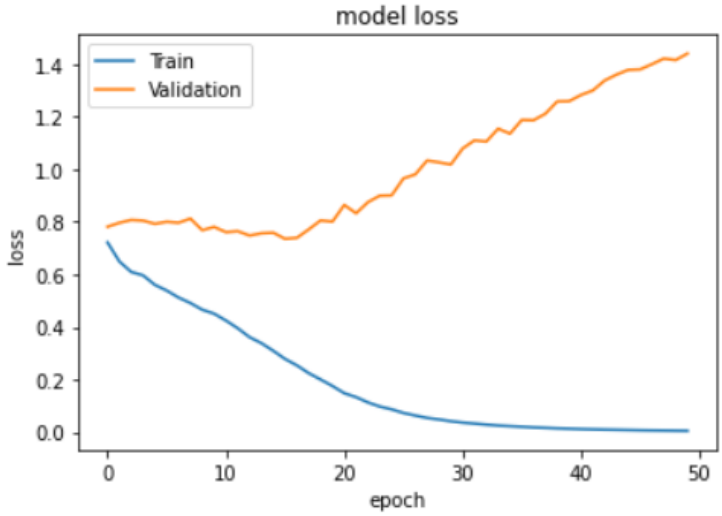


Figure 21: CNN loss on baseline input fold 3



6.2 Approach 2: Literature

For the literature input, the channels *['Fp1', 'F3', 'Fz', 'F4', 'F8', 'T7', 'C3', 'Cz', 'C4', 'AF3', 'P7']* are used. The found hyperparameters for the SVM are *SVC(C=1000, gamma='auto', kernel='linear')*. For Logistic Regression, the parameters are *LogisticRegression(C=1000, solver='liblinear')*. *DecisionTreeClassifier(max_depth=2, splitter='random')* are the found parameters for

the Decision Tree. The grid search found the same optimal hyperparameters for the SVM and Decision Tree as for the baseline input. The MLP uses the hyperparameters *MLP(activation='tanh', alpha=1e-05, learning_rate='adaptive', max_iter=4000, solver='lbfgs')*. The literature input shows a lower accuracy for all the different models. Also, there is still a lot of variation in performance between the 5 folds.

Table 5: K-fold accuracy literature input

Model	Accuracy	Mean	Standard deviation
SVM	[0.57, 0.55, 0.55, 0.55, 0.4]	0.52	0.06
Logistic regression	[0.57, 0.35, 0.60, 0.70, 0.60]	0.56	0.12
Decision Tree	[0.52, 0.60, 0.65, 0.40, 0.65]	0.56	0.09
MLP	[0.52, 0.65, 0.35, 0.45, 0.50]	0.49	0.10
CNN	[0.57, 0.55, 0.50, 0.70, 0.60]	0.58	0.07

The CNN is the same model with the same parameters as used on the baseline input. The performance of each fold can be found in appendix E. For all folds the loss starts increasing after 25 epochs. The accuracy is slightly improving over each epoch for most of the folds. It is the best performing model on the literature input. The model did not increase in performance when training on more epochs.

Figure 22: CNN accuracy on literature input fold 5

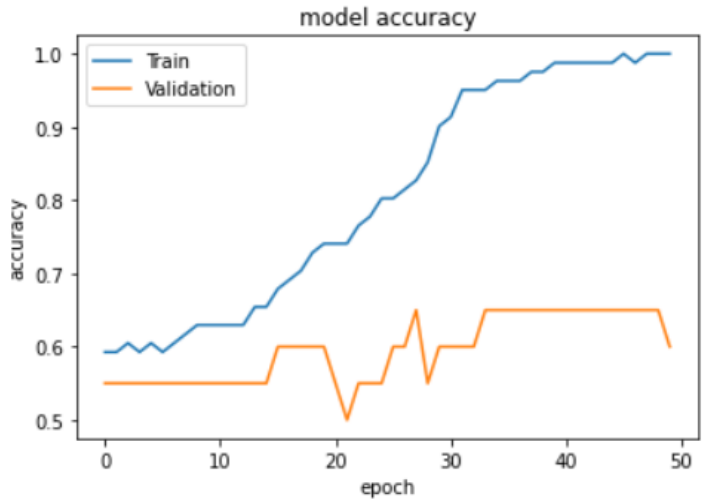
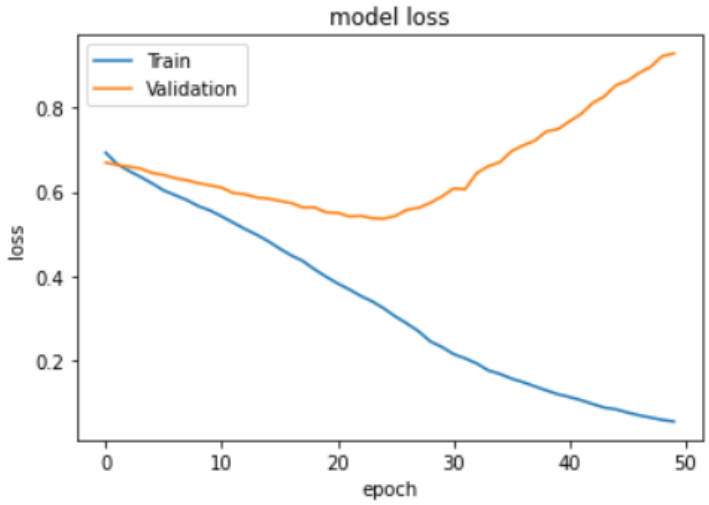


Figure 23: CNN loss on literature input fold 5



6.3 Approach 3: T-test

For the t-test input, the sensors were used which showed a significant difference between the at risk and the control group. The participants used for the analysis are excluded from the model training and validation. This resulted in less data to train the model on. The channels used as input data are *['F3', 'FC5', 'T8', 'F8', 'AF4']*. The hyperparameters used for the SVM are *SVC(C=10000, gamma='auto', kernel='linear')*. The model shows way less variation between the different folds compared to the previous two inputs. Logistic Regression has *LogisticRegression(C=1000, solver='liblinear')* as optimal parameters. The model results in higher accuracy, however, the accuracy of the folds differs a lot, which makes the model more unpredictable. The Decision Tree has a depth of 2 and is split into two nodes randomly. The model performs worse than a random prediction would do, which makes it clear that the model is incapable of learning the data. The best hyperparameters for the MLP model are *MLP(activation='relu', alpha=1e-05, learning_rate='invscaling', max_iter=4000, solver='lbfgs')*. The different folds show less variation, the same as SVM. The performance is lower compared to the SVM model.

Table 6: K-fold accuracy t-test input

Model	Accuracy	Mean	Standard deviation
SVM	[0.67, 0.57, 0.57, 0.64, 0.57]	0.60	0.04
Logistic regression	[0.40, 0.64, 0.64, 0.86, 0.50]	0.61	0.16
Decision Tree	[0.53, 0.50, 0.36, 0.36, 0.36]	0.42	0.08
MLP	[0.60, 0.57, 0.57, 0.50, 0.50]	0.55	0.04
CNN	[0.67, 0.64, 0.71, 0.50, 0.71]	0.65	0.08

Training the CNN model resulted in some issues. The network is not improving accuracy for most folds, which can indicate that the model has trouble with learning from the input data. This is confirmed by the loss which shows a constant increase for most folds. However, the model does perform well based on accuracy. A possible scenario is that the model has fewer features to train and therefore already reaches high accuracy in the beginning and stops improving. To test this, the model has been run on more epochs. The results in figure 24 and figure 25 show that the accuracy does improve over time and the loss increase after 75 epochs. The CNN is therefore the best performing model on the t-test input. The accuracy isn't smooth over the epochs because of the small size of the testing set.

Figure 24: Accuracy of the CNN on the t-test data with more epochs

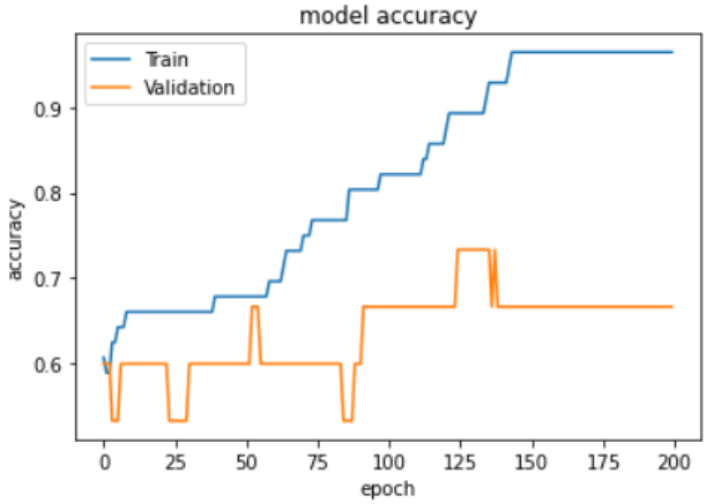


Figure 25: Loss of the CNN on the t-test data with more epochs

6.4 Approach 4: Connectivity

There are only three different channels used as input to see whether feature reduction using connectivity can help predict dyslexia. The used channels are *['Fz', 'Fp1', 'Fp2']*. Those three channels were found by performing analysis on partial data. This data is excluded same as in the previous paragraph 6.3. The hyperparameters found for the SVM model are *SVC(C=10000, gamma='auto', kernel='linear')*. The model performs well, even though the high variation in the different folds. Logistic Regression shows less variation, but a higher accuracy compared to the SVM. It uses hyperparameters *LogisticRegression(C=1000, solver='liblinear')*. The Decision Tree has a maximum depth of 20, which is remarkable since the input contains fewer features while resulting in a more complex model compared to the other used inputs. The MLP model uses hyperparameter *MLP(activation='tanh', alpha=1e-05, learning_rate='adaptive', max_iter=4000, solver='lbfgs')*. The MLP model is most compatible with the connectivity input.

Table 7: K-fold accuracy connectivity input

Model	Accuracy	Mean	Standard deviation
SVM	[0.67, 0.79, 0.50, 0.43, 0.79]	0.64	0.15
Logistic regression	[0.67, 0.57, 0.64, 0.50, 0.64]	0.60	0.06
Decision Tree	[0.53, 0.50, 0.640, 0.43, 0.50]	0.52	0.07
MLP	[0.53, 0.71, 0.71, 0.50, 0.57]	0.60	0.09
CNN	[0.56, 0.81, 0.81, 0.56, 0.67]	0.68	0.11

The CNN model has the highest performance on the connectivity input compared to the other models. However, the accuracy, based on training over 50 epochs, did not increase or even decrease for most folds as seen in appendix G. The loss also increases. The model has been trained on more epochs since this approach also uses fewer input features compared to approach 1 and approach 2, and for the t-test, the model performed better using more epochs. This resulted in higher accuracy and a decreasing training loss as seen in figure 26 and figure 27.

Figure 26: Accuracy of the CNN on the connectivity data with more epochs

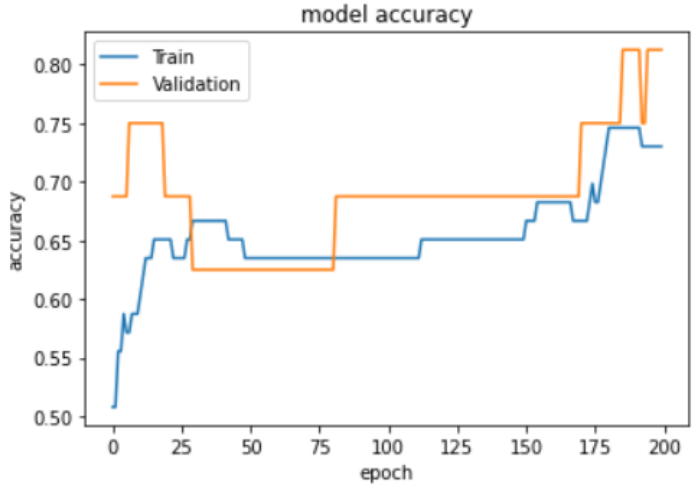
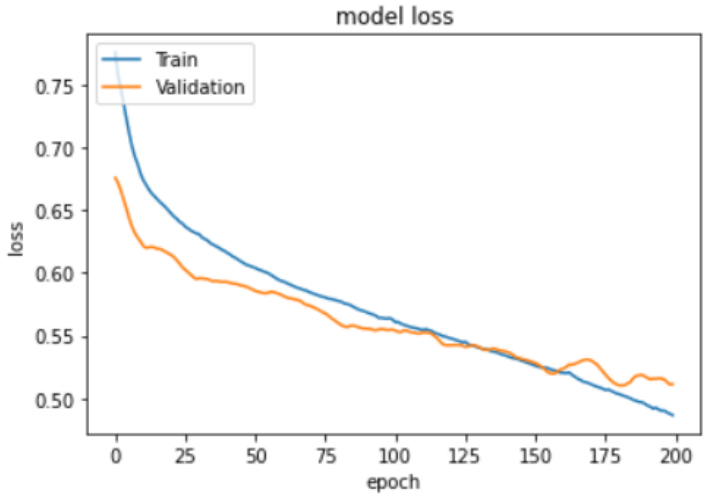


Figure 27: Loss of the CNN on the connectivity data with more epochs



7 Conclusion

Multiple research has been done to predict dyslexia in infants. Most methods were based on using the epochs and feeding them directly to a model. This research aimed to develop a theory-driven approach, based on the literature and the ePod dataset. The formulated research question was:

To what extent can data-driven features be used with machine learning models to predict developmental dyslexia in infants?

After extensive research on the literature, 4 different approaches were found to answer the research question. Since the ePod data was collected in a way that the mismatch negativity (MMN) easily could be calculated, the main goal was to validate this theory by using machine learning. The first approach is to use all the channels and calculate the MMN of each channel and feed this as input to a model. The second approach is to pick channels as input based on previous research. Different studies already tried to predict dyslexia using EEG data and machine learning. The most commonly used channels will be the features for a literature-based input. The other two approaches are based on the ePod dataset. By performing data analysis on partial data, some channels showed a significant difference between the at risk and control groups. The significantly different channels were used on an independent part of the data as a third input feature set, called the t-test input. Connectivity between EEG electrodes was used in some studies as a feature reduction method. Therefore, connectivity analysis has been done for both at risk and control groups, using t-testing, to determine which channels are most connected during the EEG trial. Those channels were used as the fourth and final input of this research.

The selected models are support vector machines, logistic regression, decision tree, multilayer perceptron, and a convolutional neural network, based on their usage in previous studies and their predictive capabilities in binary classification.

The CNN is the best-performing model using the t-test approach with an accuracy of 73%. The model also has a high performance on the connectivity input with an accuracy of 68% followed by the baseline input with an accuracy of 67%. Using the t-test and connectivity as input for the traditional machine learning algorithms SVM and Logistic Regression results in a relatively high performance around 60%. However, there is high variation in model performances and therefore the results are insignificant.

To answer the research question, data-driven selected features, using significance testing and connectivity, show promising results in predicting developmental dyslexia in infants using deep learning and traditional machine learning models, nevertheless, the results are so far not significant.

8 Discussion

The best-performing model is the CNN model with the t-test input followed by the connectivity input. However, both approaches needed more epochs to train the model to increase performance. Adding more epochs on the baseline and literature approach did not result in better performance. A possible reason for this can be that the baseline and literature approach contain more features and therefore easily overfit. The t-test and connectivity approach have less features and with not enough epochs the model might underfit. Therefore the model uses a different amount of epochs to train on. The CNN in combination with the baseline input with all channels also show high accuracy and is learning from the data. Looking at the simpler machine learning models, SVM, and Logistic Regression, the t-test, and connectivity input both have higher accuracy. Overall can be said, that the t-test and connectivity approach are best performing with both traditional machine learning and deep learning. The decision tree seems to be not suitable for this problem since it doesn't show good performance on any of the different data inputs. The literature input also doesn't show high performances. This can be because the most common channels from different studies were used, however, the channels combined may not correspond well with one another.
None of the models are performing significantly better compared to the other models. This is caused by the high variation of performance in the different runs of each model. The performances can vary because the model has a hard time finding patterns in the dataset. A possible cause is that the size of the dataset is either too small or the EEG is too noisy for the model to find a pattern.

The different approaches for feature selection do not have much overlap in the used electrodes. The features for the connectivity approach are for example located in the prefrontal cortex, while the used channels for the t-test approach are more spread over the scalp. Further analysis can be done on the influence of selecting individual channels on predicting dyslexia. Since the results of this research are not significant due to high variation in performance, no model evaluation on feature importance has been done.

The created CNN model is a simple model containing 3 dense layers. Increasing the complexity of the model can result in better performance. For this research, the decision has been made to keep the model simple, since the scope is to see how different inputs can affect the performance and a more complex model will make it harder to interpret the results. The model complexity of the CNN can be increased in further research to see if this can improve the model performance. Hyperparameter tuning can increase the complexity and performance to of the CNN. An additional recommendation is to explore different neural network architectures of their effects on the ePod data since this research is limited with the MLP and CNN.

The accuracy from the different folds highly varies for most models and

model inputs. Although this is very normal since the input of the model changes for every fold, it might interfere with the model's integrity. A way to solve this problem is to look for each fold if the model is bad at performing a specific class. It might be that the model learned a good representation to predict the control group, but when a fold contains more at risk values, the model might perform worse. Due to time constraints, this problem will be pushed to future research.

For this research, only the data from test A is used because of the possibility that the children still have a memory of the first test during test B. However, it might be that there is useful information on dyslexia in test B, also because the children are already 3 months older during test B. Since the scope of this research is to see in which extend models based on theory can predict dyslexia, the decision is made to not dive deeper into the difference in performance between test A and test B. However, this can be done in further research and might result in more training data.

There is a difference in response between the different events. All models are trained on the single gop event since this event showed the expected difference between the control and at risk groups. However, it is recommended to perform the same research on other events to see whether a machine learning model can detect a better pattern than the human eye. All standard events and all deviant events weren't combined, since research showed that there is a difference between dyslexic children and non-dyslexic children in the single and multiple stimuli, but also in the perception of different vowels [64] [65].

Another discussion point of this research is the dependent variable. The dependent variable is based on whether the parents have dyslexia. It is unknown whether the infant will develop dyslexia or not. The only way to find out is to gather additional data about the infants after a few years. A potential direction for future research is to look at creating a continuous variable based on the cdi scores of the children. For this research the decision is made to go for a binary at risk variable based on the parents.

Finally, a lot of data has been lost by averaging over all epochs and calculating the MMN. This results in less data to feed into deep learning algorithms while the data still have a high dimensionality of 2048 features. Deep learning algorithms have difficulties learning due to this curse of dimensionality. To solve this, a simple solution has been chosen to collect information about the time series, such as the surface and the maximum, to reduce the number of features. Another option was to create batches of the raw data, however, the focus of this research was to see if a theoretical approach performs better compared to existing research by using basic machine learning models, which do not require high dimensional input features. For further research a recommendation will be to not average all standard and deviant epochs to create the MMN but to calculate the MMN in batches of the standard and deviant epochs. In this way, there is more training data left to use deep learning algorithms.

References

[1] P. Kirby, "Dyslexia debated, then and now: a historical perspective on the dyslexia debate," *Oxford Review of Education*, vol. 46, no. 4, p. 472–486, 2020.

[2] G. R. Lyon, "A definition of dyslexia," *Annals of Dyslexia*, vol. 53, pp. 1–14, 2003.

[3] R. L. Peterson and B. F. Pennington, "Seminar: Developmental dyslexia," *Lancet*, vol. 379, pp. 1997–2007, 2012.

[4] R. W. Emerson, "Functional neuroimaging of high-risk 6-month-old infants predicts a diagnosis of autism at 24 months of age," *Science Translational Medicine*, vol. 9, 2017.

[5] J. J. Wolff and J. Piven, "Predicting autism in infancy," *Journal of the American Academy of Child Adolescent Psychiatry*, vol. 60, no. 8, 2021.

[6] B. Maassen, "Dutch dyslexia programme (ddp): Early neurophysiological precursors of developmental dyslexia.," 2014. ANT Burgundy Neuromeeting 2014 ; Conference date: 29-01-2014 Through 01-02-2014.

[7] M. Garrido, J. Kilner, K. Stephan, and K. Friston, "The mismatch negativity: A review of underlying mechanisms," *Clinical Neurophysiology*, vol. 120, no. 3, pp. 453–463, 2009.

[8] J. F. Démonet, M. J. Taylor, and Y. Chaix, "Developmental dyslexia," *The Lancet*, vol. 363, no. 9419, pp. 1451–1460, 2004.

[9] S. Al Otaiba, "Dyslexia and the brain: What does current research tell us?," *International Reading Association*, pp. 506–515, 2007.

[10] C. Snow, M. Burns, and P. Griffin, "Preventing reading difficulties in young children," *Washington DC: National Academy Press*, 1998.

[11] F. Ramus, S. Rosen, S. Dakin, and et al., "Theories of developmental dyslexia: insights from a multiple case study of dyslexic adults," *Brain*, vol. 126, pp. 841–865, 2003.

[12] J. Stein and V. Walsh, "To see but not to read; the magnocellular theory of dyslexia," *Trends Neuroscience*, vol. 20, pp. 147–152, 1997.

[13] R. Nicolson and A. Fawcett, "Automaticity: a new framework for dyslexia research?," *Cognition*, vol. 35, pp. 159–182, 1990.

[14] P. Tallal, "Auditory temporal perception, phonics, and reading disabilities in children," *Brain Lang*, vol. 9, pp. 182–198, 1980.

[15] E. Aylward, T. Richards, V. Berninger, W. Nagy, K. Field, A. Grimme, and et al., "Instructional treatment associated with changes in brain activation in children with dyslexia," *Neurology*, vol. 61, pp. 212–219, 2003.

[16] B. A. Shaywitz, S. Shaywitz, K. Pugh, W. Menci, R. Fulbright, and P. Skudiarksi, "Disruption of posterior brain systems for reading in children with developmental dyslexia," *Biological Psychiatry*, vol. 52, pp. 201–110, 2002.

[17] S. Dehaene and L. Cohen, "Cultural recycling of cortical maps," *Neuron*, vol. 56, pp. 384–398, 2007.

[18] G. H., "Anatomy of the human body," *Bartleby.com*, 1918.

[19] J. Booth and D. Burman, "Development and disorders of neurocognitive systems for oral language and reading," *Learning Disability Quarterly*, vol. 24, pp. 205–215, 2001.

[20] Y. Masud and A. Ajmal, "Left-handed people in a right-handed world: A phenomenological study," *Pak. J. Soc. Clin. Psychol.*, vol. 9, pp. 49–60, 01 2012.

[21] . A. V. J. Acharya, J. N., "Overview of eeg montages and principles of localization," *Journal of clinical neurophysiology : official publication of the American Electroencephalographic Society*, vol. 36, no. 5, pp. 325–329, 2019.

[22] S. Nagel, *Towards a home-use BCI: fast asynchronous control and robust non-control state detection*. PhD thesis, 12 2019.

[23] A. A. Nayak CS, "Eeg normal waveforms," *StatPearls*, 2022.

[24] S. J. Luck, *An introduction to the Event-Related potential technique*. The MIT Press, 2014.

[25] M. Luján and et. al., "A survey on eeg signal processing techniques and machine learning: Applications to the neurofeedback of autobiographical memory deficits in schizophrenia," *Electronics*, vol. 10, no. 3037, 2021.

[26] S. Dhivya and A. S. Nithya, "A review on machine learning algorithm for eeg signal analysis," *2018 Second International Conference on Electronics, Communication and Aerospace Technology (ICECA)*, pp. 54–57, 2018.

[27] M. Hosseini, A. Hosseini, and K. Ahi, "A review on machine learning for eeg signal processing in bioengineering," *IEEE Reviews in Biomedical Engineering*, vol. 14, pp. 204–218, 2001.

[28] Y. Zhang and et al., "An investigation of deep learning models for eeg-based emotion recognition," *Frontiers in Neuroscience*, 2020.

[29] Y. Gao and et al., "Deep convolutional neural network-based epileptic electroencephalogram (eeg) signal classification," *Frontiers in Neuroscience*, 2020.

[30] "Math behind SVM(Support Vector Machine), howpublished = ://ankitnitjsr13.medium.com/math-behind-svm-support-vector-machine-864e58977fdb." Accessed: 2022-11-11.

[31] "Logistic Regression in Machine Learning kernel description." ://www.javatpoint.com/logistic-regression-in-machine-learning.

[32] L. Buitinck, G. Louppe, M. Blondel, F. Pedregosa, A. Mueller, O. Grisel, V. Niculae, P. Prettenhofer, A. Gramfort, J. Grobler, R. Layton, J. VanderPlas, A. Joly, B. Holt, and G. Varoquaux, "API design for machine learning software: experiences from the scikit-learn project," in *ECML PKDD Workshop: Languages for Data Mining and Machine Learning*, pp. 108–122, 2013.

[33] A. Dertat, "Applied deep learning - part 4: Convolutional neural networks," 2017.

[34] R. Yuda, C. Aroef, Z. Rustam, and H. Alatas, "Gender classification based on face recognition using convolutional neural networks (cnns)," *Journal of Physics: Conference Series*, vol. 1490, p. 012042, 03 2020.

[35] O. Al Zoubi, C. Ki Wong, R. Kuplicki, H. Yeh, A. Mayeli, H. Refai, M. Paulus, and J. Bodurka, "Predicting age from brain eeg signals-a machine learning approach," *Front. Aging Neuroscience*, vol. 10, 2018.

[36] I. Slimen, L. Boubchir, and H. Seddik, "Epileptic seizure prediction based on eeg spikes detection of ical-preictal states," *Joural of biomedical research*, vol. 34, no. 3, pp. 162–169, 2020.

[37] C. Besthorn, R. Zerfass, C. Geiger-Kabisch, S. Sattel, S. Daniel, U. Schreiter-Gasser, and H. Förstl, "Discrimination of alzheimer's disease and normal aging by eeg data," *Electroencephalography and clinical Neurophysiology*, vol. 103, pp. 241–248, 1997.

[38] X. Wang, D. Nie, and B. Lu, "Emotional state classification from eeg data using machine learning approach," *Neurocomputing*, vol. 129, pp. 94–106, 2014.

[39] S. Gibbon and et al., "Machine learning accurately classifies neural responses to rhythmic speech vs. non-speech from 8-week-old infant eeg," *Brain and Language*, vol. 220, 2021.

[40] H. Perera, M. Shiratuddin, and K. Wong, "Review of eeg-based pattern classification frameworks for dyslexia," *Brain Inform*, vol. 5, no. 2, 2018.

[41] P. P. Sams, M., K. Alho, and R. Näätänen, "Auditory frequency discrimination and event-related potentials," *Electroencephalogr Clin Neurophysiol*, vol. 62, pp. 437–448, 1985.

[42] D. Umbricht, R. Koller, L. Schmid, A. Skrabo, C. Grübel, T. Huber, and H. Stassen, "How specific are deficits in mismatch negativity generation to schizophrenia?," *Biological Psychiatry*, vol. 53, no. 12, pp. 1120–1131, 2003.

[43] T. Baldeweg, A. Richardson, S. Watkins, C. Foale, and J. Gruzelier, "Impaired auditory frequency discrimination in dyslexia detected with mismatch evoked potentials," *Annals of Neurology*, vol. 45, no. 4, pp. 495–503, 2001.

[44] N. Armanfard, M. Komeili, J. Reilly, and J. Connolly, "A machine learning framework for automatic and continuous mmn detection with preliminary results for coma outcome prediction," *IEEE Journal of Biomedical and Health Informatics*, vol. PP, pp. 1–1, 10 2018.

[45] A. Hramov and et al., "Perept-related eeg classification using machine learning approach and features of functional brain connectivity," *Chaos*, vol. 29, 2019.

[46] J. Ye, T. Wu, J. Li, and K. Chen, "Machine learning approaches for the neuroimaging study of alzheimer's disease," *AI Redux*, 2011.

[47] G. Deshpande, L. LIbero, K. Sreenivasan, H. Deshpande, and R. Kana, "Identification of neural connectivity signatures of autism using machine learning," *Brain Health and Clinical Neuroscience*, 2013.

[48] F. Martinez-Murcia and et al., "Eeg connectivity analysis using denoising autoencoders for the detection of dyslexia," *International Journal of Neural Systems*, vol. 30, no. 7, 2020.

[49] A. Gramfort, M. Luessi, E. Larson, D. A. Engemann, D. Strohmeier, C. Brodbeck, R. Goj, M. Jas, T. Brooks, L. Parkkonen, and M. S. Hämäläinen, "MEG and EEG data analysis with MNE-Python," *Frontiers in Neuroscience*, vol. 7, no. 267, pp. 1–13, 2013.

[50] C. M. Moore, N. Prins, F. Pauwels, B. Bruns, and F. Huber, "eegyolk," 7 2022.

[51] T. Kluyver, B. Ragan-Kelley, F. Pérez, B. Granger, M. Bussonnier, J. Frederic, K. Kelley, J. Hamrick, J. Grout, S. Corlay, P. Ivanov, D. Avila, S. Abdalla, and C. Willing, "Jupyter notebooks – a publishing format for reproducible computational workflows," in *Positioning and Power in Academic Publishing: Players, Agents and Agendas* (F. Loizides and B. Schmidt, eds.), pp. 87 – 90, IOS Press, 2016.

[52] A. Newman, *Data Science for Psychology and Neuroscience - in Python.* Creative Commons Attribution-NonCommercial-ShareAlike 4.0, 2020.

[53] D. Guldenring, D. D. Finlay, A. Kennedy, R. R. Bond, M. Jennings, and J. McLaughlin, "The effects of 40 hz low-pass filtering on the magnitude of the spatial ventricular gradient," in *2019 Computing in Cardiology (CinC)*, pp. Page 1–Page 4, 2019.

[54] D. Tanner, K. Morgan-Short, and S. Luck, "How inappropriate high-pass filters can produce artifactual effects and incorrect conclusions in erp studies of language and cognition," *Psychophysiology*, vol. 52, no. 8, pp. 997–1009, 2015.

[55] J. Lin, X. Sun, J. Wu, S. Chan, and W. Xu, "Removal of power line interference in eeg signals with spike noise based on robust adaptive filter," *The University of Hong Kong, Pokfulam Road, Hong Kong and Guangdong University of Technology, Guangzhou, P. R. China*, 2016.

[56] A. Morley and L. Hill, "10-20 system eeg placement," *European Respiratory Society*, 2016.

[57] M. Jas, D. Engemann, Y. Bekhti, F. Raimondo, and A. Gramfort, "Autoreject: Automated artifact rejection for meg and eeg data," *NeuroImage*, vol. 159, pp. 417–429, 2017.

[58] P. W. Jusczyk and E. A. Hohne, "Infants' memory for spoken words," *Science*, vol. 277, no. 5334, pp. 1984–1986, 1997.

[59] G. M. Rojas, C. Alvarez, C. E. Montoya, M. de la Iglesia-Vayá, J. E. Cisternas, and M. Gálvez, "Study of resting-state functional connectivity networks using eeg electrodes position as seed," *Frontiers in Neuroscience*, vol. 12, 2018.

[60] A. Gholamy, V. Kreinovich, and O. Kosheleva, "Why 70/30 or 80/20 relation between training and testing sets: A pedagogical explanation," *Departmental Technical Reports (CS)*, 2018.

[61] P. Nardi, *Doing survey research: A guide to quantitative methods*. 2018.

[62] P. Virtanen, R. Gommers, T. E. Oliphant, M. Haberland, T. Reddy, D. Cournapeau, E. Burovski, P. Peterson, W. Weckesser, J. Bright, S. J. van der Walt, M. Brett, J. Wilson, K. J. Millman, N. Mayorov, A. R. J. Nelson, E. Jones, R. Kern, E. Larson, C. J. Carey, İ. Polat, Y. Feng, E. W. Moore, J. VanderPlas, D. Laxalde, J. Perktold, R. Cimrman, I. Henriksen, E. A. Quintero, C. R. Harris, A. M. Archibald, A. H. Ribeiro, F. Pedregosa, P. van Mulbregt, and SciPy 1.0 Contributors, "SciPy 1.0: Fundamental Algorithms for Scientific Computing in Python," *Nature Methods*, vol. 17, pp. 261–272, 2020.

[63] G. N. Cornelis J. Stam and A. Daffertshofer, "Phase lag index: assessment of functional connectivity from multi channel eeg and meg with diminished bias from common sources.," *Human Brain Mapping*, vol. 28, no. 11, pp. 1178–1193, 2007.

[64] M. Noordenbos, E. Segers, W. Serniclaes, H. Mitterer, and L. Verhoeven, "Neural evidence of allophonic perception in children at risk for dyslexia," *Neuropsychologia*, vol. 50, pp. 2010–2017, 2012.

[65] A. Thiede, P. Virtala, I. Ala-Kurikka, E. Partanen, M. Huotilainen, K. Mikkola, P. H. Leppänen, and T. Kujala, "An extensive pattern of atypical neural speech-sound discrimination in newborns at risk of dyslexia," *Clinical Neurophysiology*, vol. 130, no. 5, pp. 634–646, 2019.

A Mismatch negativity of control group for each event

Figure 28: Mismatch GiepM of control group

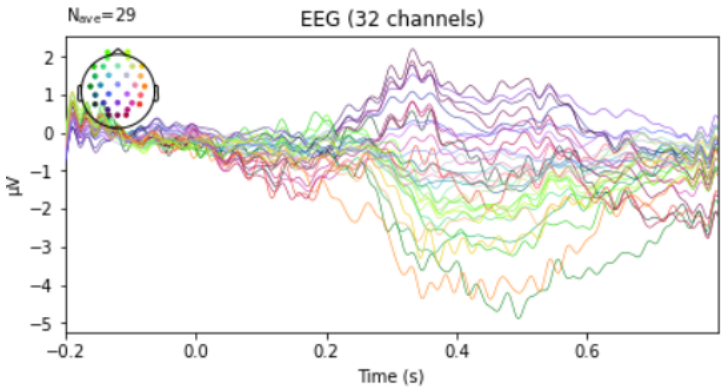


Figure 29: Mismatch GiepS of control group

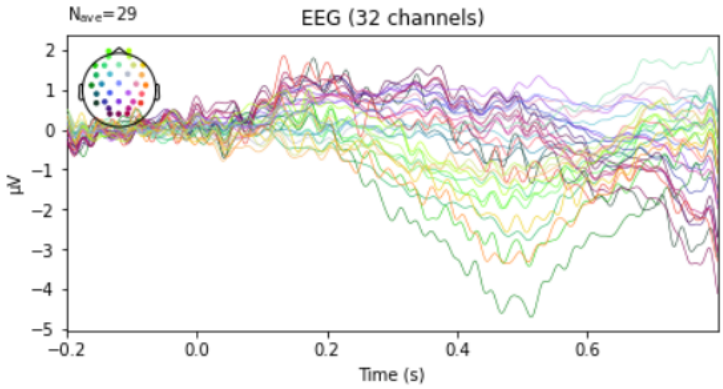


Figure 30: Mismatch GopM of control group

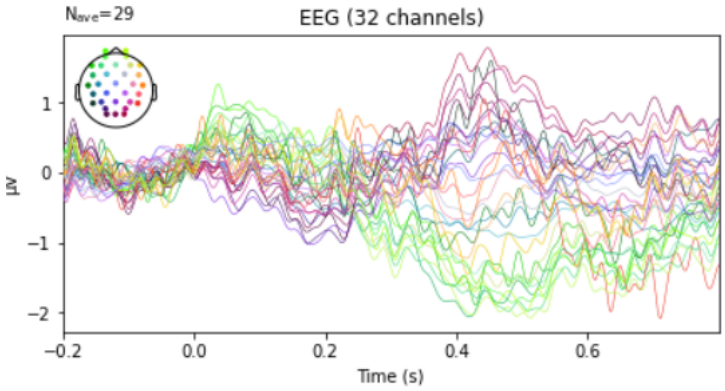
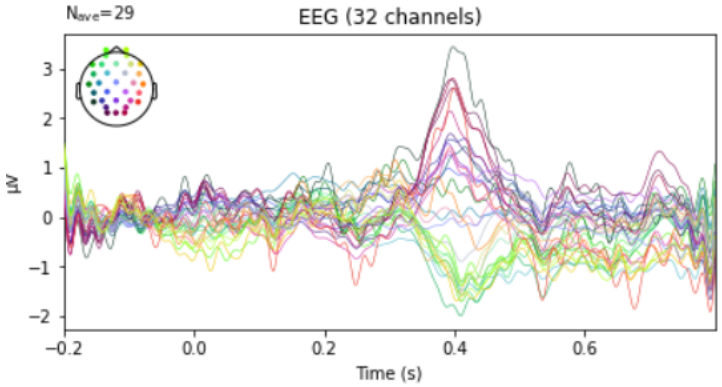


Figure 31: Mismatch GopS of control group



B Mismatch negativity of at risk group for each event

Figure 32: Mismatch GiepM of at risk group

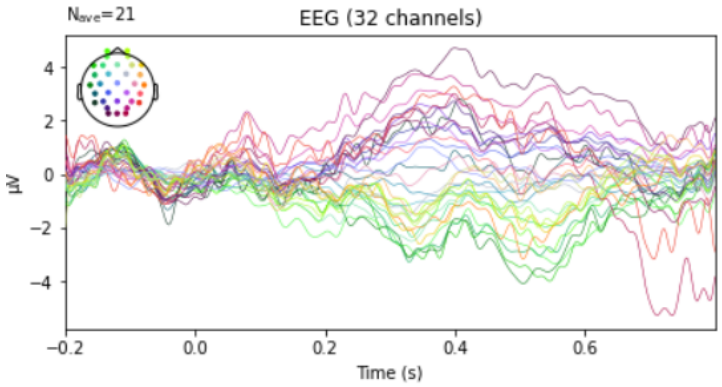


Figure 33: Mismatch GiepS of at risk group

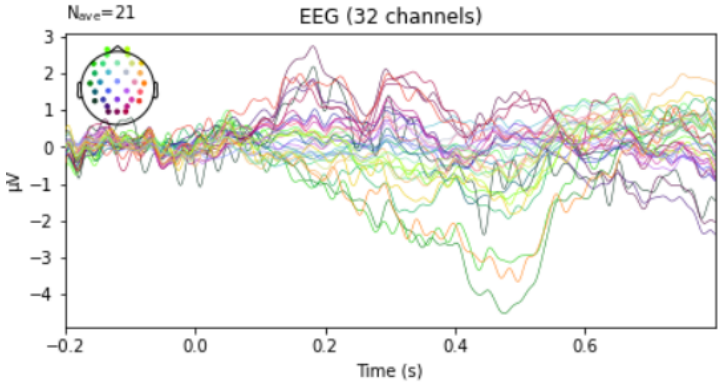


Figure 34: Mismatch GopM of at risk group

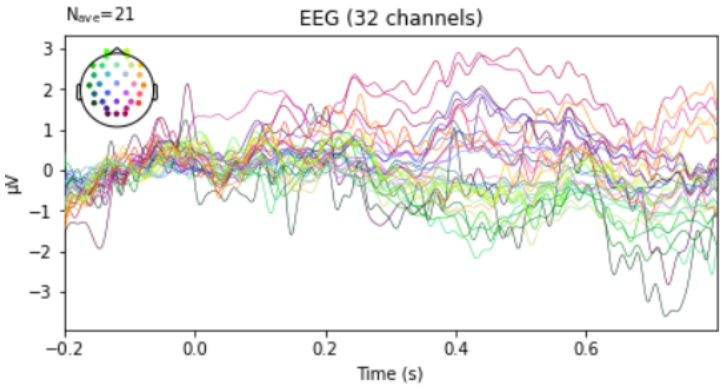
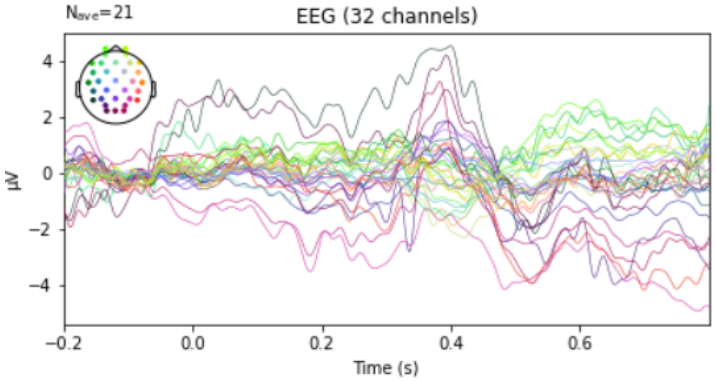


Figure 35: Mismatch GopS of at risk group



C Decision Trees

Figure 36: Decision Tree on the baseline data

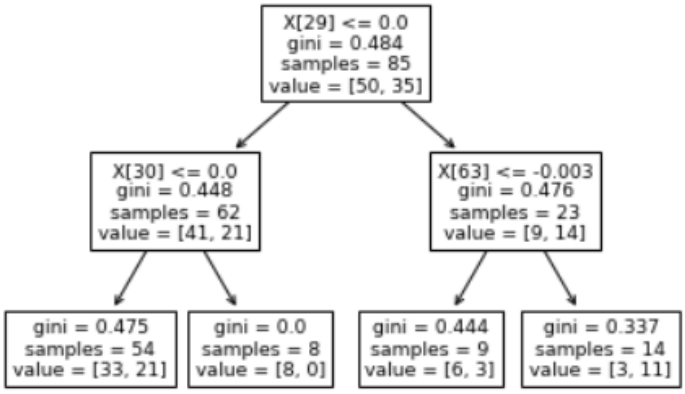


Figure 37: Decision Tree on the literature data

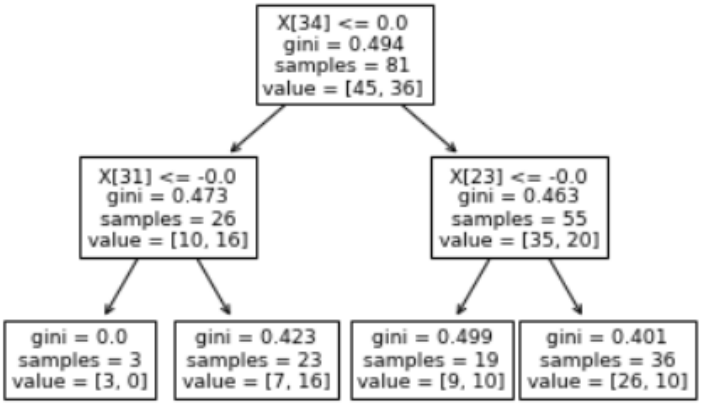


Figure 38: Decision Tree on the t-test data

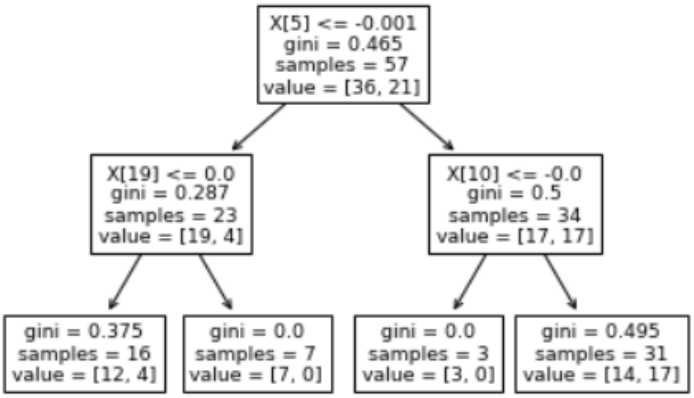


Figure 39: Decision Tree on the connectivity data

D CNN performance on baseline

Figure 40: CNN accuracy on baseline input fold 1

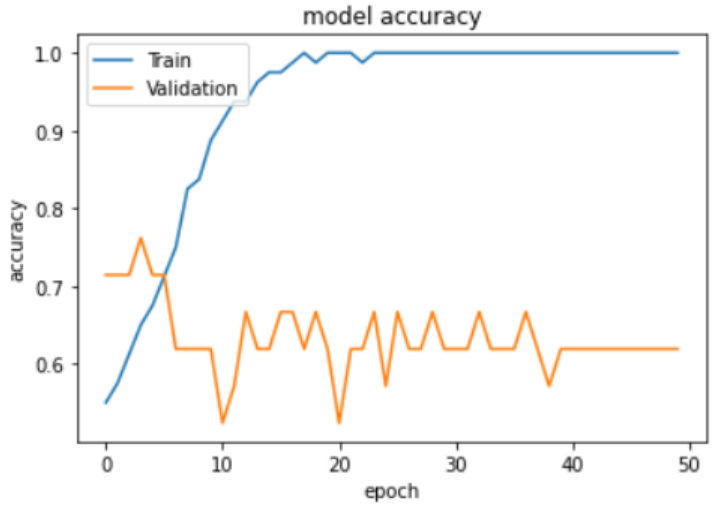


Figure 41: CNN loss on baseline input fold 1

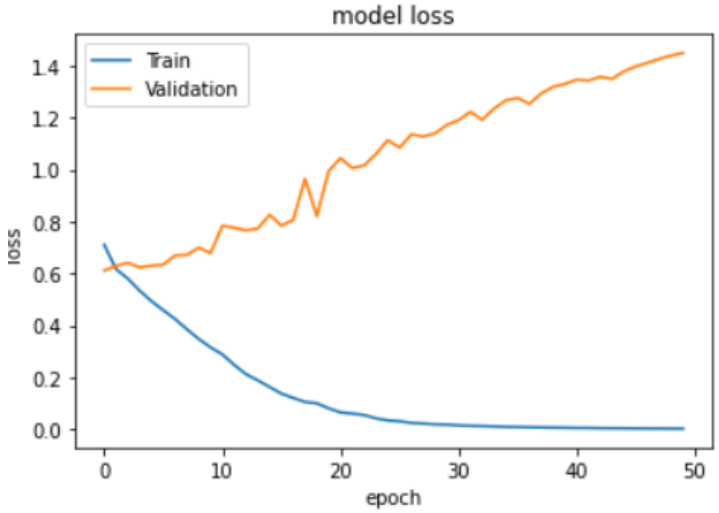


Figure 42: CNN accuracy on baseline input fold 2

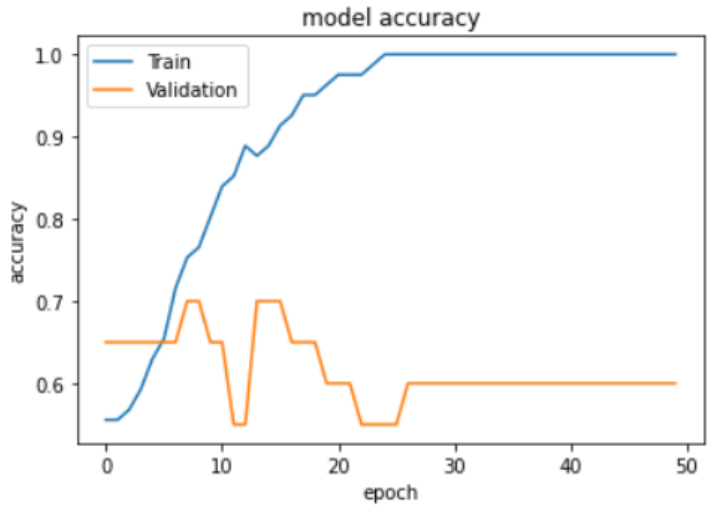


Figure 43: CNN loss on baseline input fold 2

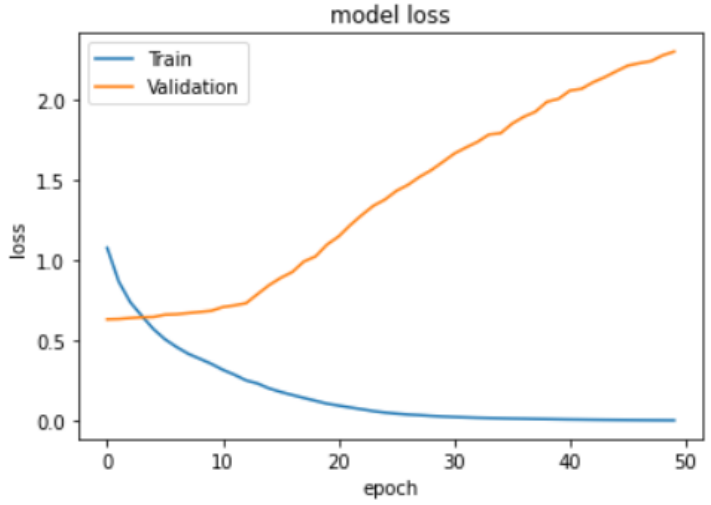


Figure 44: CNN accuracy on baseline input fold 3

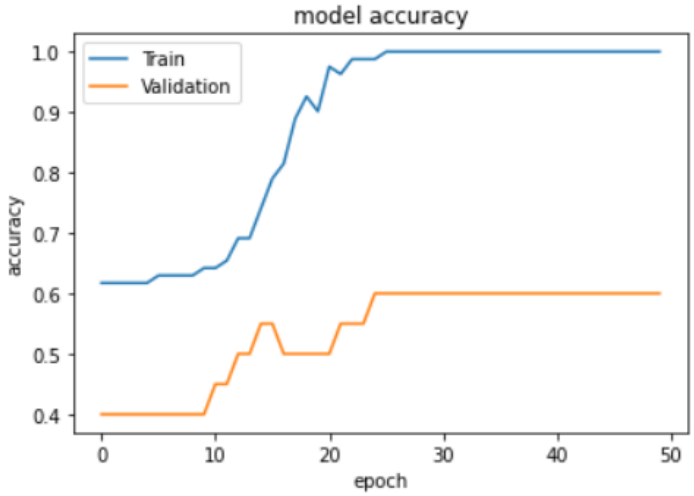


Figure 45: CNN loss on baseline input fold 3

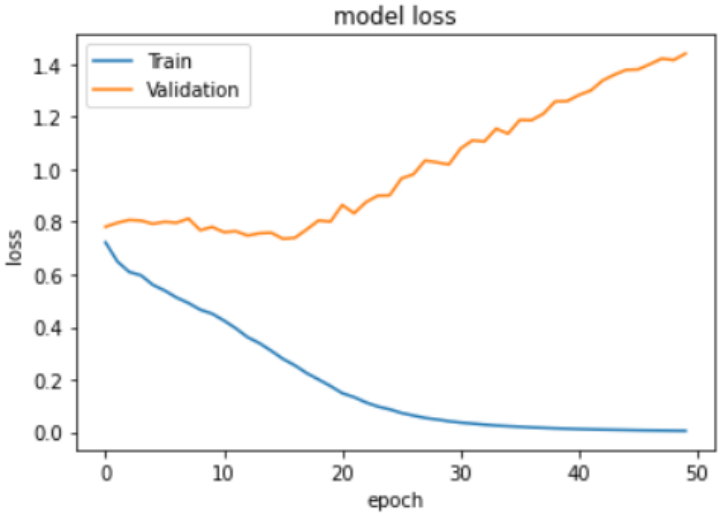


Figure 46: CNN accuracy on baseline input fold 4

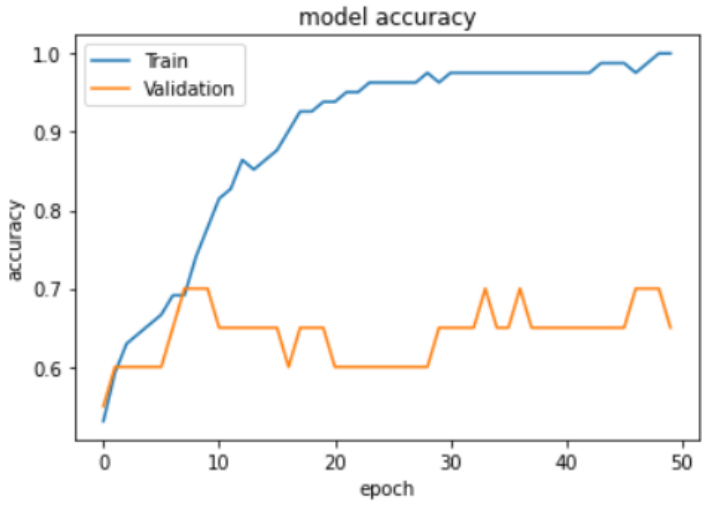


Figure 47: CNN loss on baseline input fold 4

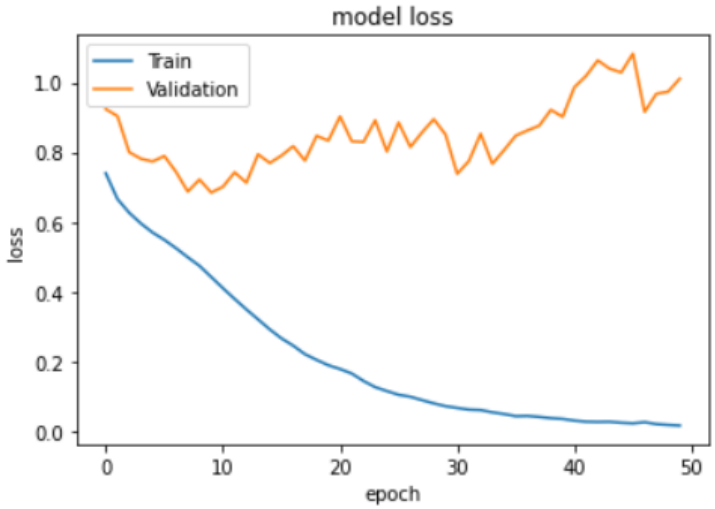


Figure 48: CNN accuracy on baseline input fold 5

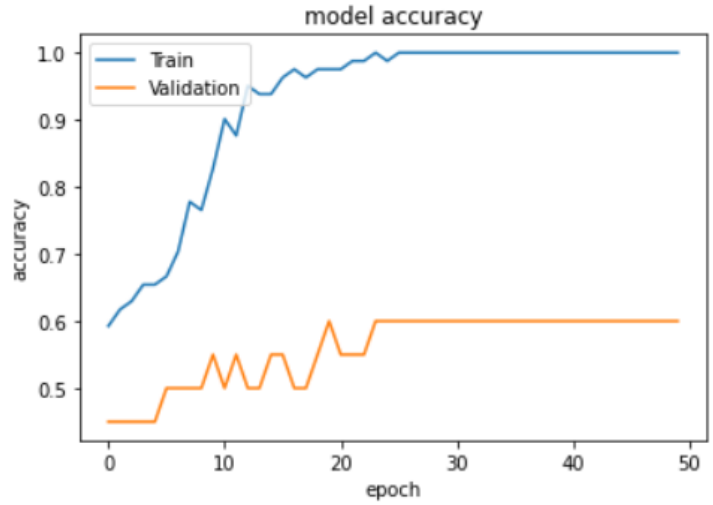
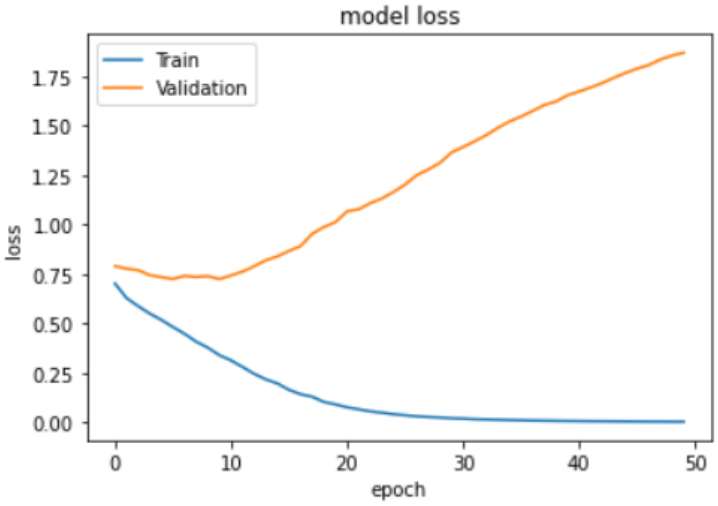


Figure 49: CNN loss on baseline input fold 5



E CNN performance on literature

Figure 50: CNN accuracy on literature input fold 1

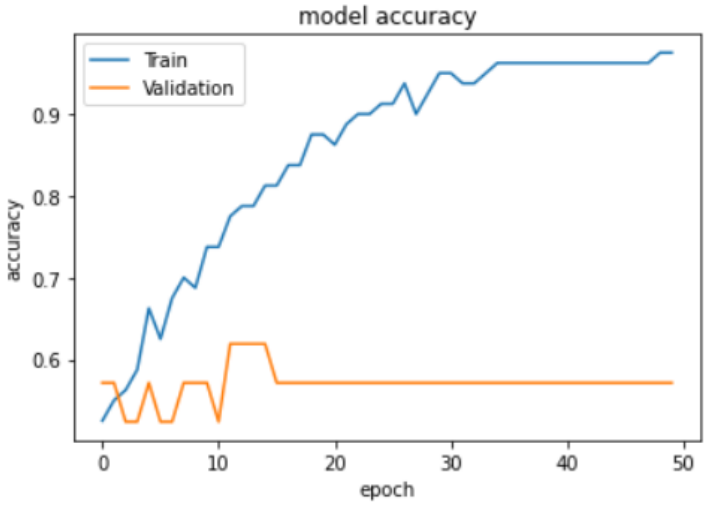


Figure 51: CNN loss on literature input fold 1

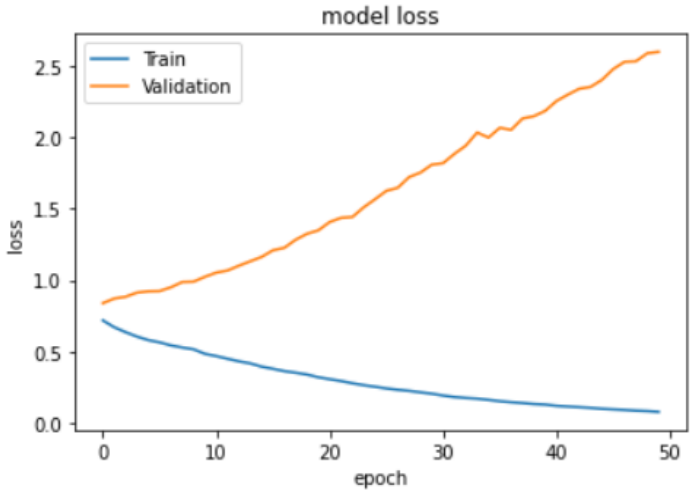


Figure 52: CNN accuracy on literature input fold 2

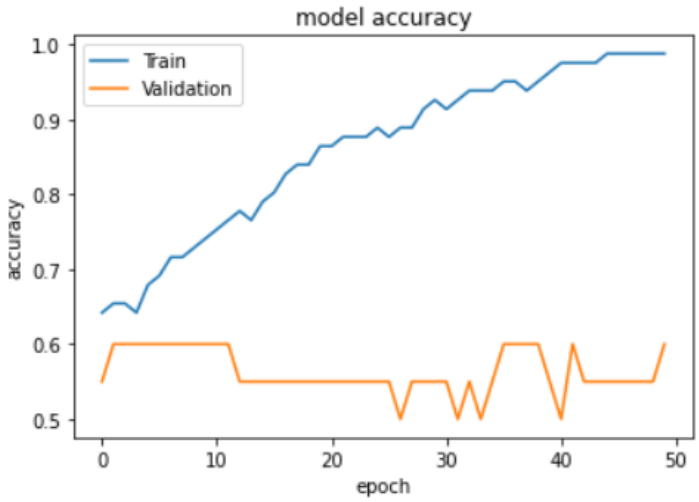


Figure 53: CNN loss on literature input fold 2

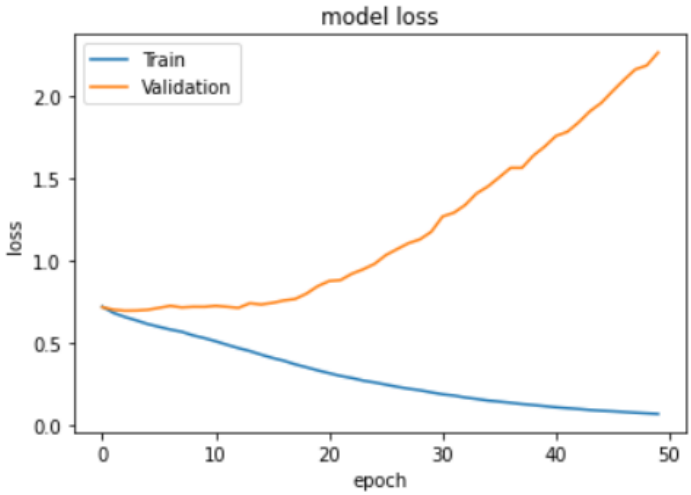


Figure 54: CNN accuracy on literature input fold 3

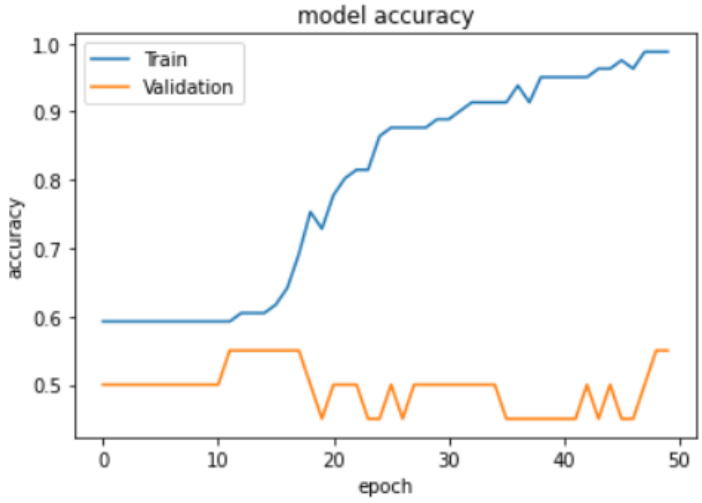


Figure 55: CNN loss on literature input fold 3

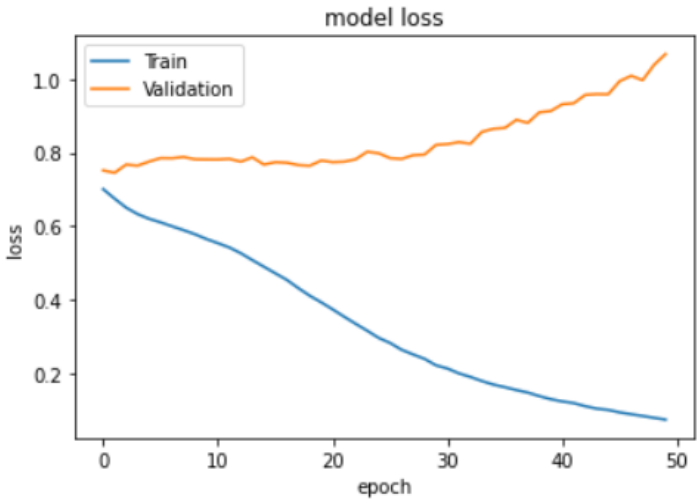


Figure 56: CNN accuracy on literature input fold 4

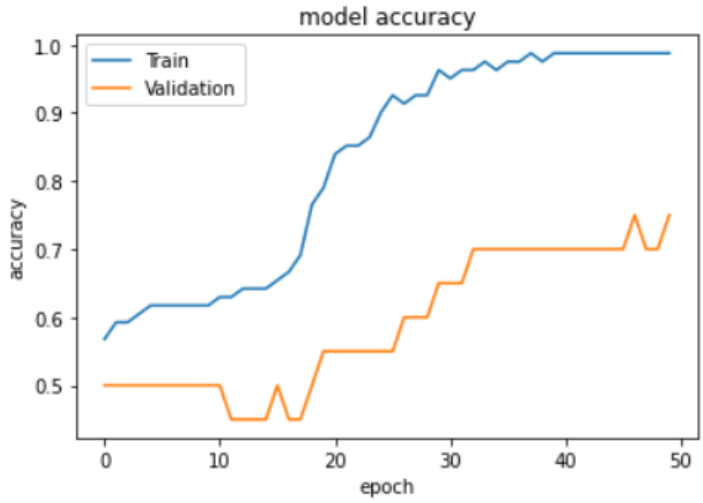


Figure 57: CNN loss on literature input fold 4

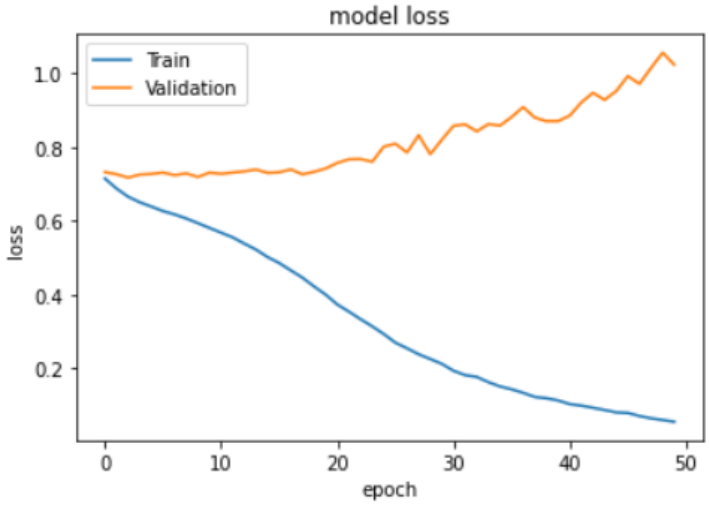
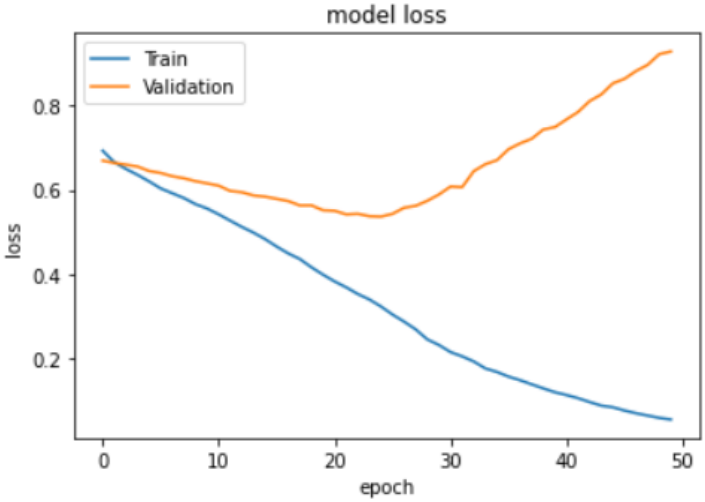


Figure 58: CNN accuracy on literature input fold 5

Figure 59: CNN loss on literature input fold 5



F CNN performance on t-test

Figure 60: CNN accuracy on t-test input fold 1

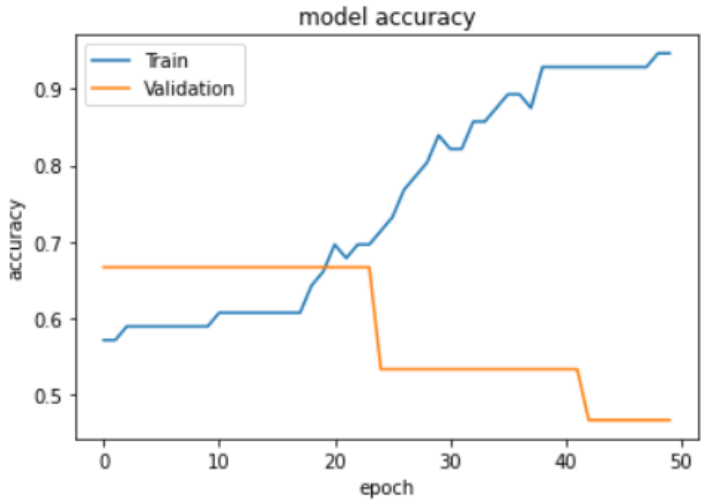


Figure 61: CNN loss on t-test input fold 1

Figure 62: CNN accuracy on t-test input fold 2

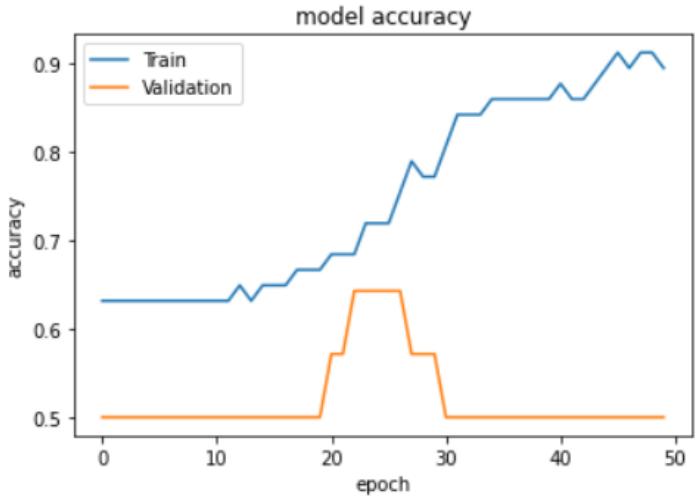


Figure 63: CNN loss on t-test input fold 2

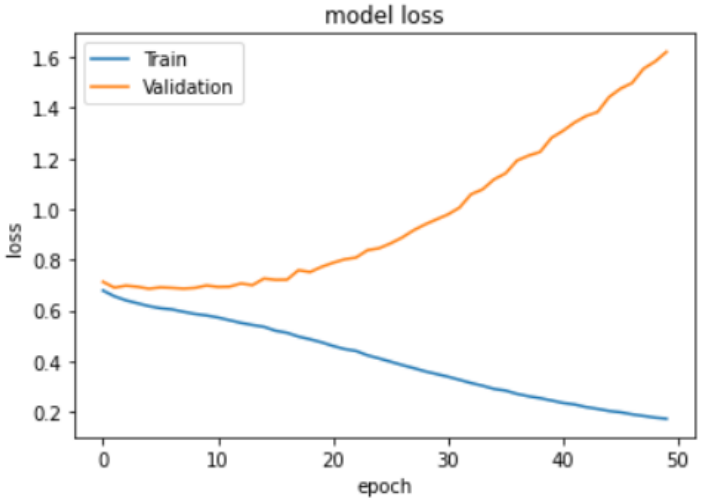


Figure 64: CNN accuracy on t-test input fold 3

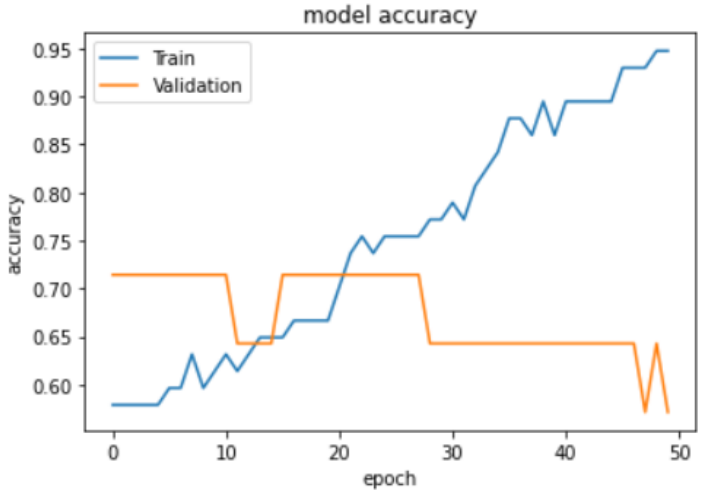


Figure 65: CNN loss on t-test input fold 3

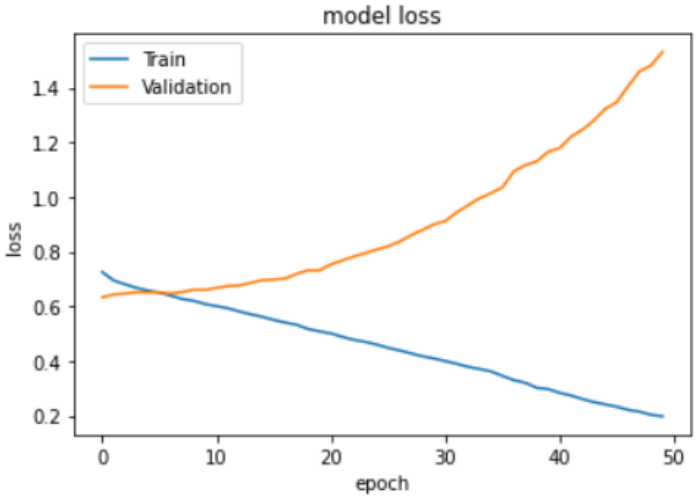


Figure 66: CNN accuracy on t-test input fold 4

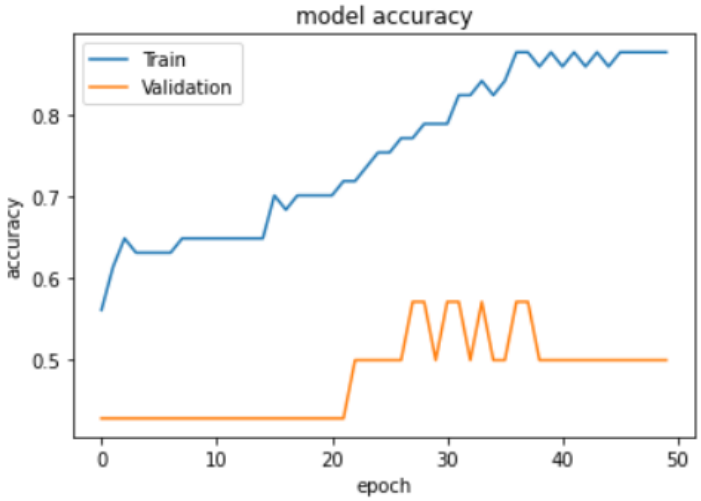


Figure 67: CNN loss on t-test input fold 4

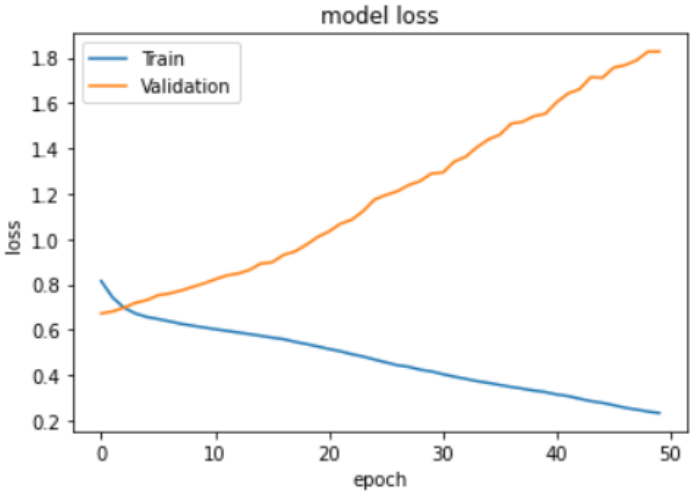


Figure 68: CNN accuracy on t-test input fold 5

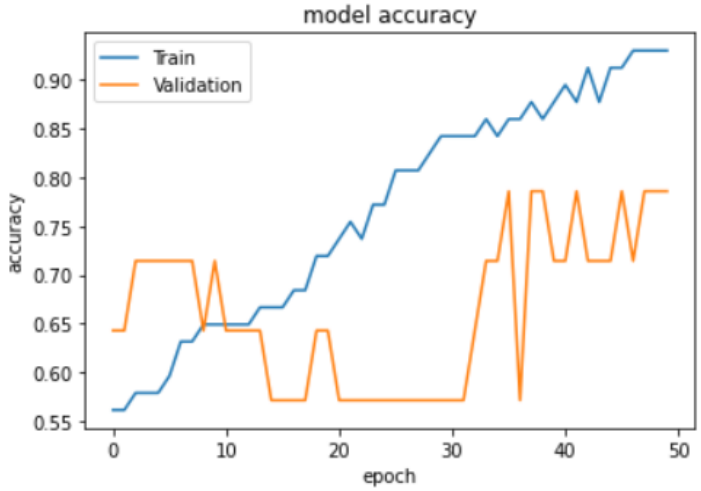


Figure 69: CNN loss on t-test input fold 5

G CNN performance on connectivity

Figure 70: CNN accuracy on connectivity input fold 1

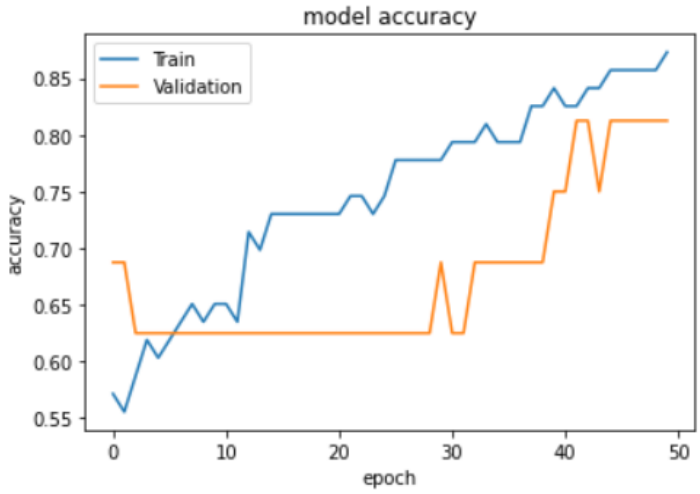


Figure 71: CNN loss on connectivity input fold 1

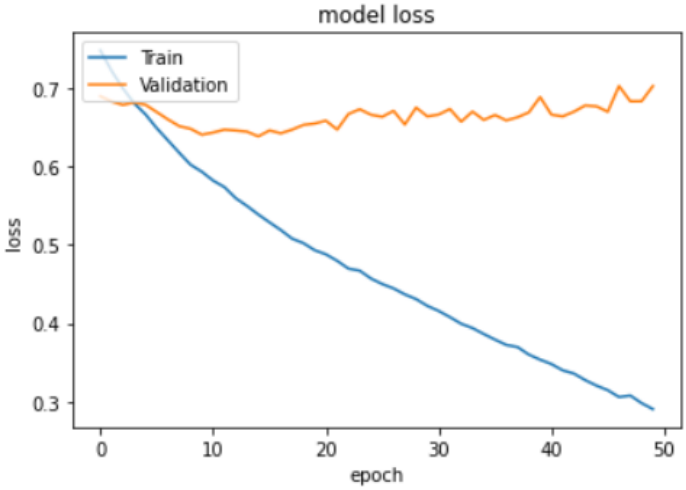


Figure 72: CNN accuracy on connectivity input fold 2

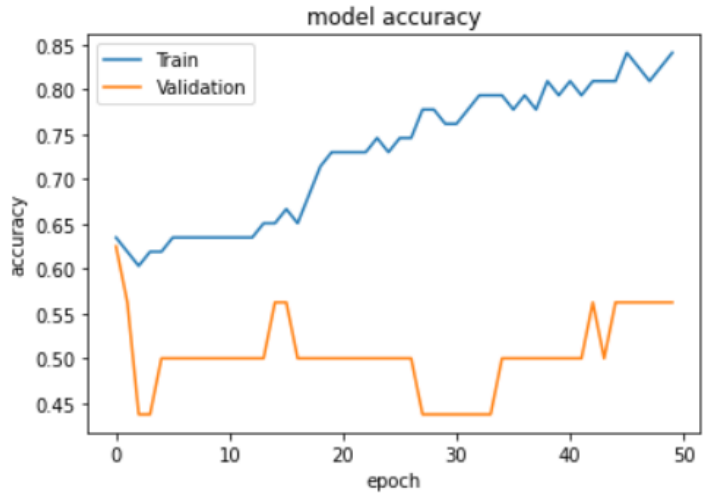


Figure 73: CNN loss on connectivity input fold 2

Figure 74: CNN accuracy on connectivity input fold 3

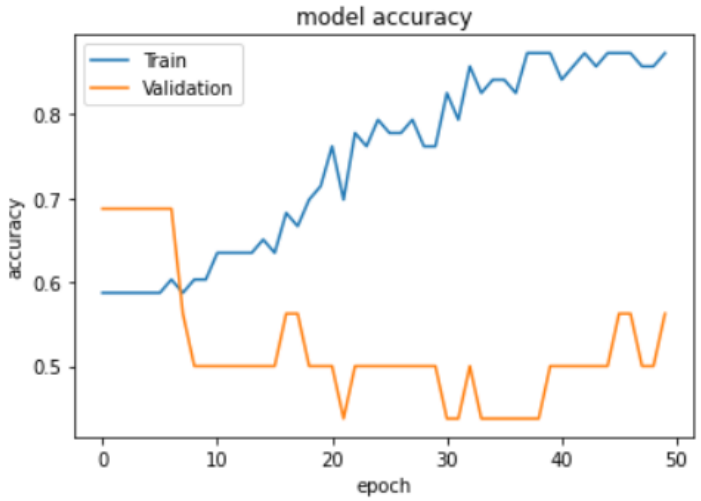


Figure 75: CNN loss on connectivity input fold 3

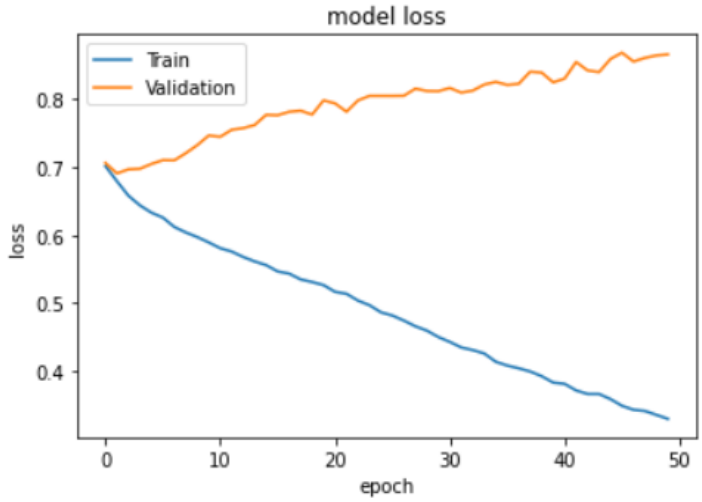


Figure 76: CNN accuracy on connectivity input fold 4

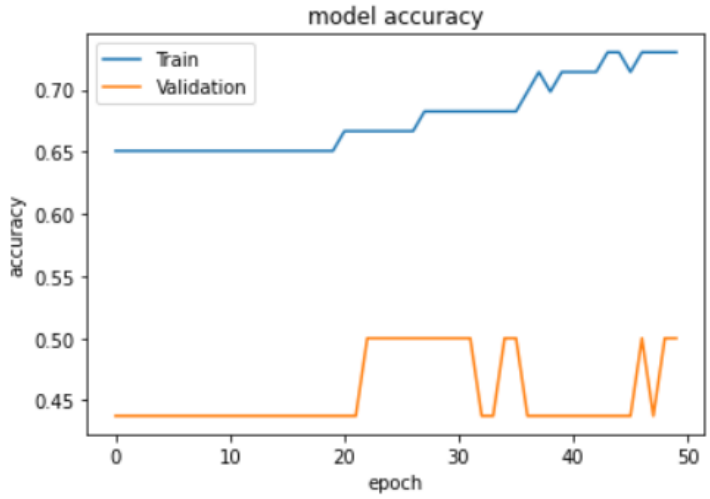


Figure 77: CNN loss on connectivity input fold 4

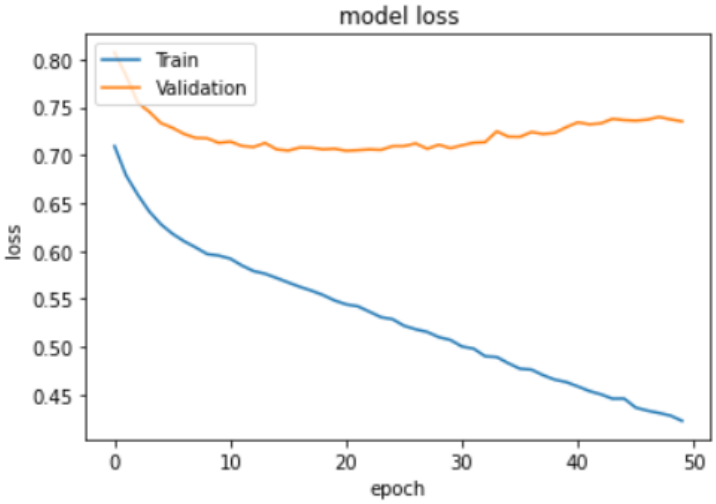


Figure 78: CNN accuracy on connectivity input fold 5

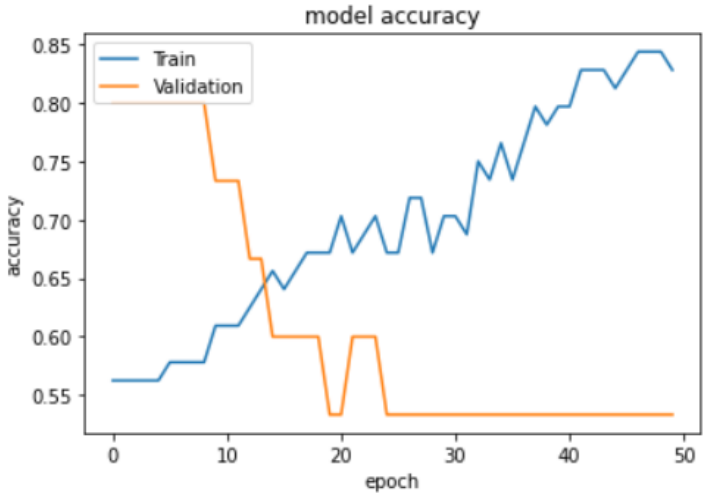


Figure 79: CNN loss on connectivity input fold 5