POLITECNICO
MILANO 1863

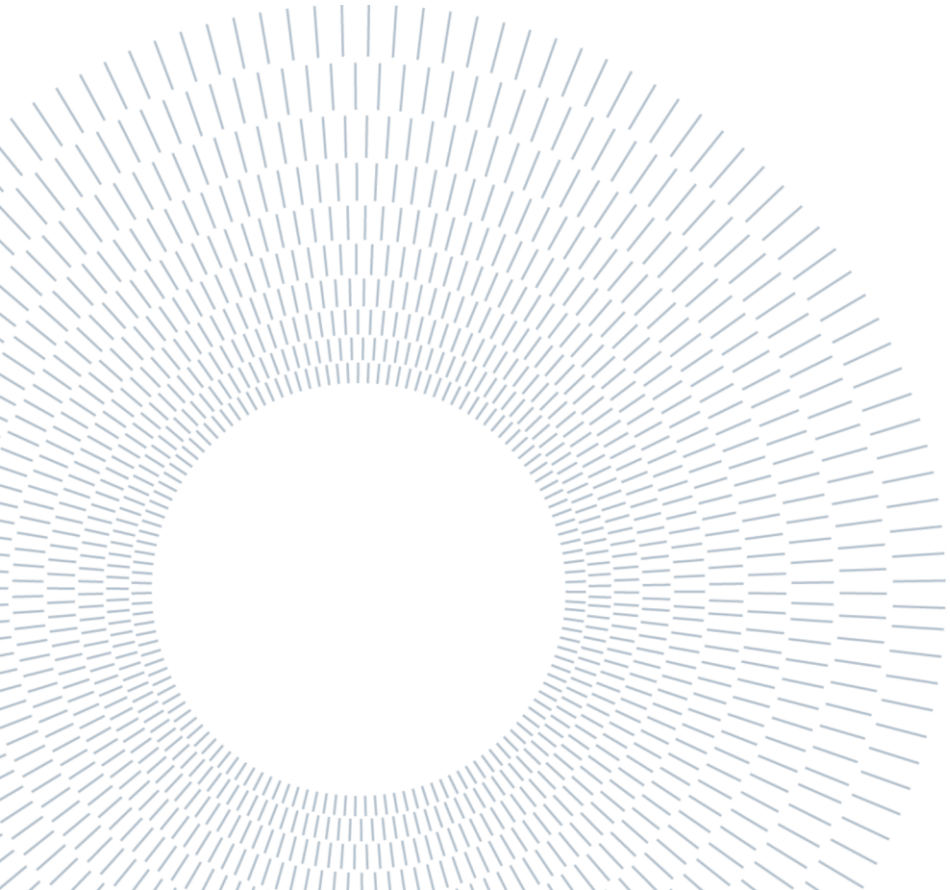
SCUOLA DI INGEGNERIA INDUSTRIALE
E DELL'INFORMAZIONE

# A Multi-center Normative Deep Learning Approach for Automatic Detection of Bipolar Disorder Patients based on Neuroanatomy

TESI DI LAUREA MAGISTRALE IN BIOMEDICAL ENGINEERING-INGEGNERIA BIOMEDICA

Author: **Inês Won Sampaio**

| | |
|---|---|
| Student ID: | 970353 |
| Advisor: | Prof. Eleonora Maggioni |
| Co-advisor: | Prof. Paolo Brambilla, Emma Tassi |
| Academic Year: | 2021-2022 |



# Abstract

Bipolar disorder (BD) is a chronic and disabling mood disorder, characterized by a heterogeneous clinical and symptomatological presentation and a diagnosis latency of 5 to 10 years. Identifying objective neurobiological markers for BD, such as those based on neuroimaging data, might help improve the diagnosis sensitivity by translating quantitative knowledge to clinical practice. A vast amount of neuroscientific literature has reported neuroanatomical alterations correlating with BD, although evidence is fragmented. Machine Learning (ML), as a multivariate statistical method, has become a widely used approach to investigate biological markers and build predictive models for clinical diagnosis. Nevertheless, limitations have hampered their development and application, such as the large number of cases required for the training process and the lack of domain relevance of most models, being characterized as "black box", providing no insight into disease pathophysiology mechanisms. In the present study, for the first time, we experiment an alternative approach for the automatic detection of BD, based on structural neuroimaging data, using an Autoencoder-based (AE) normative model, trained solely on healthy controls' (HC) data devoid of confounding factors. We use a multisite 3T structural Magnetic Resonance Imaging (sMRI) dataset composed of 605 HC and 558 BD, from which we extract brain morphological features and design both an internal and external validation framework, to evaluate the model's discriminative power and generalizability. To eliminate confounding effects in the sMRI data, we compare different multisite data harmonization options using the ComBat tool combined with biological covariates correction. We conclude that estimating ComBat center effects solely in the training set, via a CV framework, leads to an effective harmonization of training, test, and external set. After being trained and tested on HC data, the AE model is employed in an anomaly detection framework on BD data, using the reconstruction error to spot deviating samples, achieving an AUC of 0.51 for BD discrimination, using all brain features. With the proposed model, we then identify BD neuroanatomical deviating features and assess if they help increase the discriminatory power, achieving an AUC of 0.61 in the external set, higher than the AUC obtained in a traditional SVM approach.

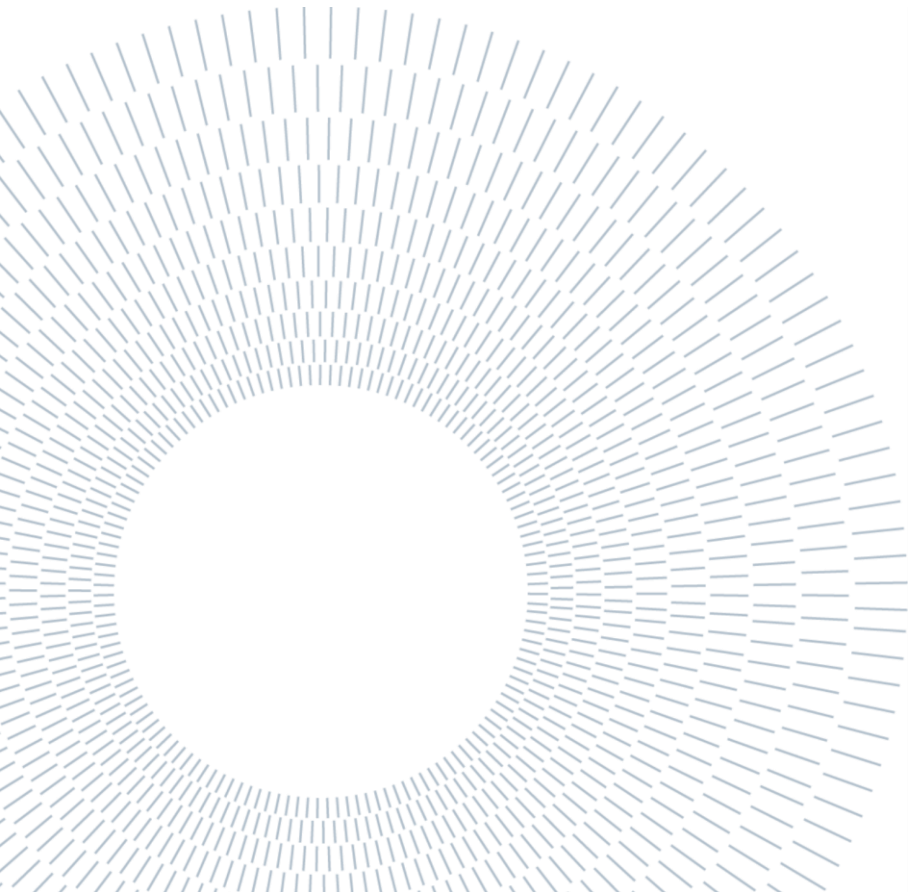
**Key-words:** Normative Model, Anomaly Detection, Bipolar Disorder, Multisite Data

# Abstract in lingua italiana

Il disturbo bipolare (BD) è un disturbo dell'umore cronico e invalidante, caratterizzato da una presentazione clinica e sintomatologica eterogenea e da una latenza della diagnosi che si stima dai 5 ai 10 anni. L'identificazione di marcatori neurobiologici oggettivi relativi al BD, come quelli basati sui dati di neuroimaging, potrebbero aiutare a migliorare la qualità della diagnosi traducendo le conoscenze quantitative estratte nella pratica clinica. Una vasta quantità di letteratura neuroscientifica ha riportato alterazioni neuroanatomiche correlate al BD, sebbene le prove siano frammentate. L'utilizzo di tecniche di Machine Learning (ML), come metodo statistico multivariato, è diventato un approccio ampiamente utilizzato per studiare i marcatori biologici e costruire modelli predittivi per la diagnosi clinica di specifichi patologie. Tuttavia, tali tecniche sono associate a specifiche limitazioni proprio relative alla modalità operative di tipo "black-box", a scatola nera, che rende i processi computazioni e i meccanismi di predizione di specifiche patologie non totalmente trasparenti. In questo studio, per la prima volta, verrà sperimentato un approccio alternativo per la detezione automatica di BD, basato su dati di neuroimaging strutturale, utilizzando un modello normativo basato su Autoencoder (AE), addestrato esclusivamente su dati di controlli sani (HC) privi di fattori confondenti. Abbiamo utilizzato un set di dati 3T strutturale di risonanza magnetica (sMRI) composto da 605 HC e 558 BD, da cui abbiamo estratto le caratteristiche morfologiche del cervello e progettiamo un framework di convalida sia interno che esterno, per valutare il potere discriminativo e la generalizzabilità del modello. Per eliminare gli effetti confondenti nei dati sMRI, abbiamo confrontato diverse opzioni di armonizzazione dei dati multicentrici utilizzando il toolbox di ComBat combinato con la correzione di covariate biologiche. Concludiamo che la stima e la rimozione degli effetti centro ottenuta da ComBat esclusivamente nel training set, tramite un framework CV, porta a un'efficace armonizzazione nel training, test set e dataset esterno indipendente. Dopo essere stato addestrato e testato sui dati HC, il modello AE è stato impiegato in un framework di rilevamento delle anomalie sui dati BD, utilizzando l'errore di ricostruzione per individuare i soggetti che deviano dal modello normativo, ottenendo un AUC di 0,51 per la discriminazione del BD, utilizzando tutte le caratteristiche cerebrali. Con il modello proposto, si identificano

quindi le caratteristiche neuroanatomiche dei BD che deviano rispetto al modello normativo (i.e., modellizzato su HC), per poi valutare il relativo potere discriminatorio, raggiungendo un'AUC di 0,61 nel dataset esterno indipendente, superiore all'AUC ottenuta con un approccio SVM tradizionale.

**Parole chiave:** Modello Normativo, Identificazione Caratteristiche Anomale, Disturbo Bipolare, Dati multicentrici.



# Contents

# 1. Introduction

The argument of this thesis aligns with current concern in tackling the underdevelopment of Psychiatry methods for objective diagnostics and treatment. Precision medicine is a reality in almost all fields of medical specializations, but Psychiatry and Neuro-related specializations have experienced a significant delay in transitioning to this framework, due to an intrinsic disadvantage: the complexity of the brain. How biology and physiology affect behavior and personality is a complex field of study. There are entangled relationships between genetics and environmental factors such as nutrition, physical activity, and family environment, through epigenetics, which come to play to shape our personality, resilience to stress, and adaptation capacity. Brain disorders can be a combination of underlying etiologies that are very complex to disentangle, so researchers and doctors cannot pinpoint which of the individual systems are contributing to the pathogenesis process. Moreover, psychiatric disorders have severe consequences on those who suffer from them, on the health system, and on the economic system, yet, history shows us that they were not paid the necessary attention. There is an inability to provide a fast and accurate diagnosis which leads to an immense delay in defining a proper treatment plan. Currently, the practice of psychiatry uses a diagnosis tool book, *Diagnostic and Statistical Manual of Mental Disorders* (*DSM–5*), or *ICD-10-CM*, which guides psychiatrists in the diagnosis of their patients by categorizing disorders with a set of symptoms so that criteria checkpoints can be met. However, the knowledge gathered so far by clinicians and researchers has shown that many disorders categorize in the DSM have similar symptoms which leads to the conception that there is a continuum spectrum that blurs the frontiers between several disorders. These overlapping symptoms make it hard for a psychiatrist to nail the correct diagnosis and therapy. It is in this gap that lies the need for evidence-based approaches to help increase the accuracy and the speed of psychiatry disorders diagnosis. To bring precision to psychiatry is to bring intelligent resources that will support psychiatrists in their practice. For these purposes, there is a necessity in finding clinical biomarkers associated with the disorder pathogenesis that can be the basis for a precision-oriented diagnosis. The idea behind the work of this thesis is to contribute to a

clinical decision support systems (CDSS), that may assist psychiatrists in the diagnosis of Bipolar Disorder (BD).

## 1.1 Motivation

Bipolar Disorder (BD) is a mood [affective] disorder which causes cyclic mood swings, also known as maniac depression. It is a long-lasting and life-threatening disorder with a lifetime prevalence of around 2-5%[1]. Its patients go from one extreme mood, depression, to another, mania/hypomania, passing through euthymia (state without mood disturbances), with varying length periods, or suffer from mixed states, although, it's possible that people experience maniac episodes and never depressive ones. In fact, to be diagnosed with BD a person must have experienced at least one episode of mania or hypomania. Nevertheless, around 60% of BD patients suffered from depression as their first-lifetime affective episode [2], and these Major Depressive Episodes (MDE) seem to last longer than the maniac ones, at least 2 consecutive weeks but can last years, tending to dominate the course of the illness. Consequentially, and due to the categorical and symptom-based approach to diagnosis, it's hard to distinguish BD depression from Unipolar Depression (UD), which leads to many patients being left undiagnosed. Indeed, the latency for BD diagnosis is around 5-10 years, and many are misdiagnosed, particularly with Unipolar Depression, comprising 60% of the false diagnosis for BD[3]. The fact that BD patients are misdiagnosed with UD or Major Depression Disorder (MDD) brings severe consequences for their health and very poor prognosis, due to inadequate treatment -BD depressive patients tend to be resistant to antidepressants-, which can lead to exacerbation of symptoms.

Concluding, the diagnosis of BD is rather complex and early detection of the disease is very difficult. The latter issue is exacerbated by the heterogeneity of BD which has prevented so far the identification of specific neuroanatomical markers for an objective and precise diagnosis of the disease. Our work goes in that direction in the sense that we try to provide knowledge of the neuroanatomical bases that might allow for accurate automatic detection of BD, based on the study of BD and Healthy Control (HC) differences. Thus, this work aims to contribute to a future evidence-based clinical support system that might support clinicians in making a differential diagnosis of BD. It was focused on the use of imaging data, specifically structural MRI (sMRI), with the main purpose of discriminating BD patients from (HC).

## 1.2 Bipolar Disorder: what do we know so far?

Even though the biological basis of the disease is unknown, some progress has been made in identifying some biological modifications and risk factors for BD. There are several areas of study that investigate alterations in BD patients such as genomics, physiology, and neurosciences. These new findings bring a clearer picture of the biological etiology of BD, however very complex to integrate. There is evidence of alterations in brain connectivity levels, oxidative stress, mitochondrial function, inflammation, circadian rhythms, and dopamine levels. Another very established knowledge is that of the high heritability of BD, estimated at 85% in twin studies [4]. The study of BD etiology has been progressing from different perspectives.

Starting with genomics, there has been an effort to discover and characterize BD phenotype by searching for polymorphism in the genome of the population. It is known that BD inheritance comes in a form of several gene variants with small effects, being a polygenic disorder. These genes and loci have been identified with genome-wide association studies and have led to the discovery of several affected pathways, such as glutamate and calcium signaling. Some Voltage-gate calcium channels (VGCC) genes seem to play a role in BD but there is also evidence of their influence on schizophrenia and major depression [5]. The key might be identifying patterns of alterations rather than individual biomarkers.

In neuroimaging research, several alterations have been found both in brain Grey Matter (GM) and White Matter (WM), yet the available knowledge is fragmented[6], [7]. At the WM level, there have been findings reporting an increased rate of deep WM hyperintensities and reduced WM volumes in BD. Robust results point to alterations specifically in the cingulum, corpus callosum, frontal areas, parahippocampal areas, and tracts such as uncinate fasciculus and fornix[8].

Widespread GM alterations have been found to characterize progressive cognitive deterioration, mainly loss of GM volume. Although, as BD is a highly heterogenous disease it is characterized by variable degrees of cognitive impairment [6]. Besides, there is a lack of clear knowledge on the GM correlates of BD, as there also have been reports of GM alterations unrelated to cognitive impairment [9].

Particularly, in BD patients, there are consistent reports of alterations in the Limbic Network (LN), which is involved in stress-related modulation of homeostasis and neurotransmitter signaling, specifically an increased volume of the amygdala along with volume reductions in the hippocampus. These are paired with state-dependent metabolic changes, such as increased metabolism in the amygdala and hyperactivity of the hypothalamus-pituitary-adrenal (HPA) axis [8]. At the functional neurochemical level, there have been findings of alterations for some

neurotransmitters, mainly, elevated levels of glutamate and glutamine in BD patients, and some growth factors that promote neurogenesis and neuroplasticity called neurotrophic factors. The latter lead to impairment in plasticity and resilience in brain cells in BD patients.

At endocrine and immune system levels, many alterations have been found, either associated with states like depression, mania, or maintenance, as also with illness phase and progression. BD has been characterized by neuroinflammation with increased pro-inflammatory markers. There has been evidence for comorbidity of autoimmune diseases and BD, elevated levels of pro-inflammatory circulating cytokines (some evidence points to a causal link to maniac and depressive symptoms), C-protein, increased cortisol levels (dysfunction in stress pathways with HPA axis hyperactivity), in both mania and depression phases with complete normalization during euthymia, which leads to conclude a strong implication between BD and immune/endocrine dysfunction.

Additionally, circadian rhythms disruption has been consistently determined in BD patients, regardless of disease state, comprising lower levels of melatonin compared to healthy controls. There is also evidence for oxidative stress implications in BD, with lowered antioxidant defenses and increased oxidative and nitrosative stress which seems to lead to mitochondrial damage and dysfunction. Robust evidence suggests energetic metabolic impairment in BD brains found with neuroimaging studies and supported by some genetic findings [4]. Many of the above-mentioned systems, related to the pathophysiology of BD are interdependent. A polygenic disease, with changes at the glial and neuronal level, chronobiological alterations, immune system compromised, and mitochondrial dysfunction.

It is to determine though whether some of these factors are of etiological basis or part of disease progression, i.e., some of the described alterations may be common effects of the true etiological factors of mood disorders. It is known for example that HPA axis abnormalities are related to environmental risk factors since no variants of HPA-axis related genes were found to be associated with increased risk of BD or HPA-axis malfunction [4]. Moreover, the allostatic load could potentially explain some of these alterations: they can be thought of as the result of chronic exposure to the disease itself, after all, there are physiological consequences of exposure to chronic stress. In fact, as reported previously, GM alterations have not been found in early-stage BD or first episodes but rather have been associated with illness progression [8].

Regarding the etiology of BD, data suggests that the disorder is characterized by an immune-mediated WM damage, especially in the Limbic Network. Stress response and inflammation could induce changes in neurotransmitter availability which then could pathologically affect the functional brain activity in BD patients, which goes to

relate to structural alterations of the LN [8]. To understand how each one of these alterations is connected to the others, integrating all the information into one unified biological hypothesis for the etiology and progression of BD, is extremely complex. In [8] the authors attempt to propose a unified model of the pathophysiology of BD, Figure 1.1. The hypothesis is that a core dysregulation of the immune system leads to an LN damage which further alters neurotransmitter signaling. The model proposes that a chronic pro-inflammatory profile in BD, triggered by a susceptible polygenic background, leads to damage in the WM, due to the migration of effector T cells into brain tissue and consequent cytotoxic activity. Stress-related limbic overactivity, together with pro-inflammatory mediators may divert cerebral blood-flow to the hyperactive regions resulting in structural alterations in that LN and consequently destabilizing neurotransmitter signaling which leads to an increased susceptibility to perturbations by several stressors, either internal or environmental. The phasic mood states of BD are explained by the changes in neurotransmitter signaling. The model proposes phasic reconfigurations of intrinsic brain activity, mediated by neurotransmitter unavailability which leads to functional disconnection of the neurotransmitter-related nuclei, clinically manifesting into manic-depressive states. A stressor can trigger a stress-response that is intrinsically hyperactive in BD, causing a prolonged increase in pro-inflammatory factors which can lead to a reduction of 5HT availability (serotonin receptors) or DA availability (dopamine signaling). The latter results in a cascade of triggers, leading to either an over-tuned intrinsic brain activity, associated with maniac state, or de-tuned associated with depression, respectively. Finally, the authors hypothesized that BD subgroups are characterized by the presence of further neurodegenerative factors which lead to GM loss, hence GM alterations could be related to progressive cognitive deterioration. However, there is also evidence of widespread GM alternations unrelated to cognitive impairment [9], thus the latter correlation needs to be further investigated.

In the future, other unified models might be proposed, using new findings or taking different perspectives and hypotheses to integrate the knowledge gathered so far. For now, we accept what has been consistently reported in the literature and take advantage of the biological alterations that can be measured and used to evaluate disease progression, target treatment to improve long-term outcomes, and used for diagnostic purposes, even if a true etiological model has not been accepted and defined.

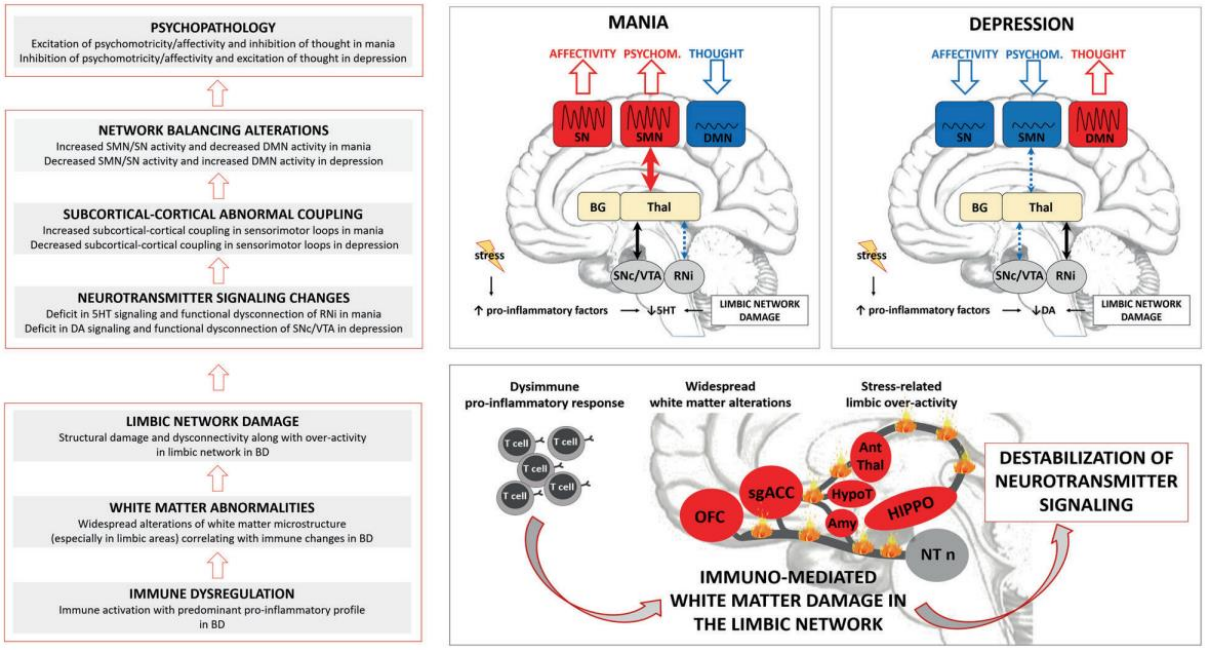



Figure 1.1: A unified model of the pathophysiology of BD [8].

## 1.3 Magnetic Resonance Imaging[10], [11]

Magnetic Resonance Imaging is a non-invasive imaging modality that uses non-ionizing electromagnetic radiation to create 3D images of the internal structures of the body. It is based on the physical phenomena of Nuclear Magnetic Resonance which consists of a spinning nucleus inducing a local magnetic field. An image is created by taking advantage of the different magnetic properties of different types of tissues, which will translate to different contrasts and shades. The spin is the central quantum property that creates the phenomenon. The spin is the intrinsic property of the angular momentum of a particle, being a fundamental property just like mass and charge, but it's quantized. Protons and electrons have a spin number of ½, and an atom that possesses an unpair proton or neutron will have a nonzero net spin which gives rise to a magnetic moment associated with the proton, and a local magnetic field is generated. Hydrogen is an example of an atom with a solitary proton, which acts as a magnetic dipole.

In paramagnetic materials, the magnetic dipoles are "relaxed" at thermal equilibrium, which means they are randomly oriented and so their cumulative or net magnetization is zero. Whenever are submitted to an external static magnetic field, $B_0$, they will re-orient themselves parallel or anti-parallel to this external field, and the number of parallel orientations – lower energy state- is higher, which will result in a global magnetization vector M, in the longitudinal direction $M_z$, with a

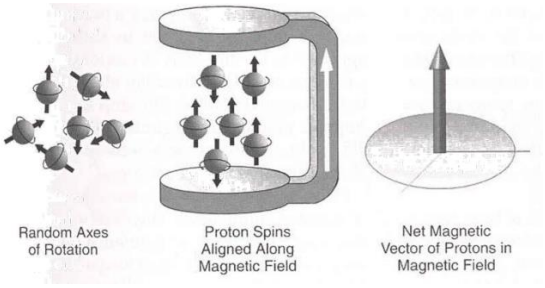
macroscopic magnetization magnitude $M_0$, Figure 1.2a). Although their axes are oriented in the direction of the field, there is a tilting from this position which is called precession, characterized by a frequency called *lamor frequency*, which is proportional to the external magnetic field, $B_0$. Because the macroscopic magnetization cannot be measured in the longitudinal direction, it is necessary to resort to further means that will allow for absorption or emission of measurable energy. The energy to force the nuclei into an energy level transition is supplied with a Radiofrequency (RF) magnetic field $B_1$. When this rotating magnetic field, $B_1$, which rotates in the transversal plane *xy*, is applied to the static external magnetic field, a transversal magnetization component appears due to the rotation of vector $M_0$ of an angle $\alpha$, resulting when $\alpha = 90º$, in an $M_z=0$, and a $M_{xy}=M_1$, as reported in Figure 1.2b). In this process, protons start precessing in phase around $M_1$. Once the RF is turned off, the system comes back to thermal equilibrium, orienting itself with $B_0$. During the relaxation process, the nuclei are undergoing transitions between energy levels, which means absorbing or releasing energy, where the energy levels correspond to the orientations parallel and anti-parallel of the nuclear axis.

To summarize, first, an external magnetic field is applied so that the nuclei axis of $^1H$ are oriented parallelly and anti-parallelly to the magnetic field $B_0$. When the energy of RF is directed, the protons which were aligned with $B_0$, and that possess a *lamor frequency* matching the RF, will absorb that energy, which is the so-called resonance, and shift away from the $B_0$ direction, with a flip angle $\alpha$ (90º or 180º). This RF is usually emitted in small pulses which will lead to an absorption-emission sequence from the nuclei that is detected in a suitable coil.

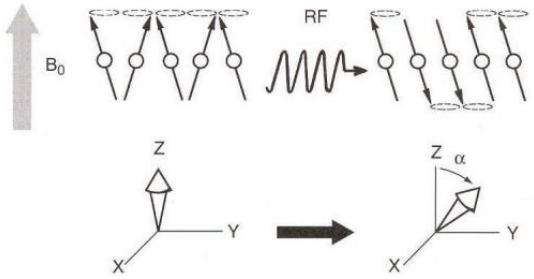
There are two types of relaxation, T1, and T2. T1 is the spin-lattice relaxation where the nuclei realign with the external magnetic field,i.e, goes back to a lower energy state – parallel orientation- by transferring energy from it (spin) to the surrounding molecules (lattice). Also called the longitudinal relaxation, where the Mz component recovers the equilibrium value of $M_0$. T1 measures the time taken for the system to return 63% towards that thermal equilibrium after RF pulse offset. The transversal magnetization $M_{xy}$ will thus decay to a null value with the T2 time constant, which is the spin-spin relaxation. This means the protons that had started precessing in phase will begin to diphase out of the *lamor frequency* in the transverse plane. The energy loss in the transverse direction, due to the change in the magnetic moment of the net magnetization is detected by the RF coil and is called Free Induction Decay (FID).

To connect the FID with a specific position in the body is then necessary to be able to discriminate the FID signal contribution for each voxel. For this purpose a spatially variant intensity magnetic field is added to $B_0$, i.e, a gradient, which encodes the spatial information, as seen in Figure 1.2 c). Because the *lamor frequency* is

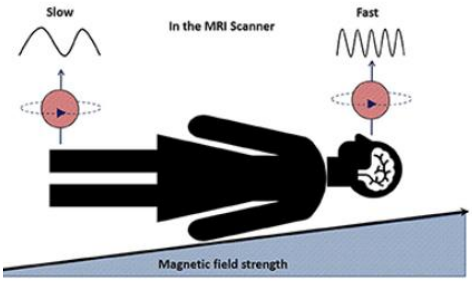
proportional to the magnetic field strength, with a gradient magnetic field, the precession frequency of protons will linearly depend on their position in space. This allows for slice selection and the production of images in the $x,y,z$ components, respectively, sagittal, coronal, and axial. If one is interested in imaging only the brain then it is just necessary to match the RF pulse with the frequency of the precessing protons for that magnetic field strength.



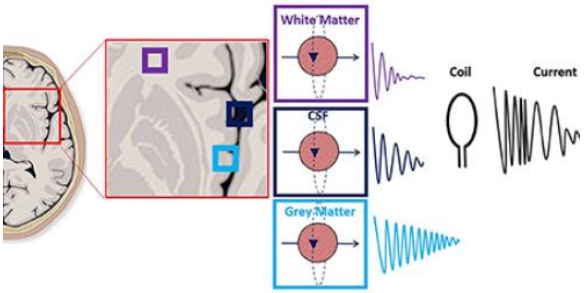

a) Net Magnetization $M0_z$.




b) $M_{xy}$ component of net magnetization.




c) Gradient Magnetic Field [12].




d) Brain T1-Weighted MRI scan [12].

Figure 1.2: MRI Principles.

The information contained in the detected frequency signals is transformed into gray values through Fourier Transformation. The signal that is detected will then depend on the presence of $^1H$, the bonding degree within a molecule and the differentiation of tissues of interest will depend on the T1 and T2 relaxation times. Because the human body is composed mainly of water, the hydrogen properties are used to produce an image. Bone tissue is characterized by firmly bound $^1H$ atoms and as a consequence, their nucleus does not produce a useful signal. In soft tissues and liquids, the present $^1H$ is more loosely bound which enables the production of a measurable signal [10]. T1-weighted images are suitable for fat tissues because fat has the shortest T1 relaxation time which produces a bright image and so good contrast is possible with other tissues. T2-weighted images are also called water images because water has the shortest T2 relaxation times. For brain imaging, one

can differentiate white matter, grey matter, bone, and Cerebral Spinal Fluid (CSF) exactly because they give off different amounts of energy when relaxation happens, as exemplified in Figure 1.2d).

### 1.3.1 MRI for BD diagnosis

Within neuroscience studies, neuroimaging data has been widely used to investigate both functional and structural brain alterations in psychiatry disorders. Brain features extracted from MRI have demonstrated reasonable power in discriminating BD patients while leaving many open questions on the brain underpinnings of this heterogenous illness. Up until now, univariate analyses have been most widely used to investigate the relationship between selected brain characteristics and disease conditions. Univariate analyses are statistical analysis methods where there is one dependent variable and one or more independent variables. Many studies investigating associations between brain morphology or functional activations and disease conditions have been performed using univariate analyses, leading to novelty findings on voxel-based or ROI-based brain alterations that need to be integrated into a whole-brain framework.

From raw sMRI T1-weighted scans many interesting brain measures can be extracted, which may further add to our knowledge of BD biomarkers. Voxel or ROI-Based Morphometry enables the estimation of local volume-based or surface-based brain morphological properties including cortical thickness, cortical and subcortical volume, cortical surface area, cortical curvature, and cortical gyrification.

As reported in section 1.2, several brain structure alterations have been found in individuals affected by BD through the employment of sMRI. In section 1.2, the focus was mainly on brain structure alterations that could be linked to the etiology of the disease, such as WM damage which seem consistently identifiable in early-stage BD or first episodes. Nevertheless, because BD may be diagnosed at later stages of progression, structural brain changes that are a consequence of exposure to the disease itself can also be useful and aid in diagnosing BD.

Besides the findings described in section 1.2, it has been shown that several areas of the brain suffer GM volume and thickness reduction, consistently in the anterior limbic regions. There have been reports specifically of grey matter volume reductions in the prefrontal cortex, left rostral anterior cingulate cortex, mean hippocampus, and thalamus accompanied by greater grey matter volumes for lateral ventricles, and grey matter thickness reductions in the right fronto-insular cortex [13]. In general, a pattern of cortical thinning has been found, mainly in frontal, temporal, and parietal

regions. Robust findings show that frontal-subcortical and prefrontal-limbic circuits are compromised in BD patients [13][4].

Moreover, patients with a longer history of BD diagnosis have been found to have a more substantially reduced cortical thickness and these changes are more pronounced in patients which have experienced multiple maniac episodes[14]. It also seems that both volume and cortical reduction in BD increase with some specific medication, therefore some brain structure alterations may be exacerbated by such environmental factors. The latter may act as confounders when investigating brain alterations that should be linked exclusively to disease or disease progression.

The identification of brain morphological alterations being consistently associated with BD over heterogeneous cognitive and clinical features may help clinicians perform early, accurate, and objective BD diagnoses, thus making the use of structural brain features to have a very relevant clinical applicability.

## 1.4 Machine Learning for BD diagnosis

At the current time, the scientific community has made available robust quantitative measures from clinical and biological data that have been integrated to provide findings on differences between BD and HC. Therefore, there have been many attempts to develop clinical support systems, mainly based on Machine Learning algorithms, that aim to diagnose BD patients. Different types of data can be used to try to make BD diagnoses, such as neuroimaging data, genetic data, blood biomarkers, neuropsychological data, etc., as well as integrating all of them. Even though univariate analysis has been very useful to pinpoint alterations in several clinical feature expressions, a strong drawback is that variables under investigation are considered independent of each other, therefore, multi-voxel patterns or feature patterns of structural and functional alterations across conditions cannot be studied. To overcome this limitation, multivariate analysis can be used to study possible correlated variables, considering the effects of all variables in the condition of interest. Machine Learning, which is the study of algorithms and multivariate statistical methods that allow machines to learn without human intervention, has emerged as the most used method for pattern recognition tasks and prediction tasks. In a decoding setting, it is used to make predictions about variables of interest, based on the joint analysis of multiple features. In a recent review study, 33 articles from 2016 to 2021 that used ML for BD diagnosis were analyzed [15]. It was found that the most commonly used data was clinical, with MRI being the most widely used datatype type while genomic data was the least one. In terms of classification ML models, Support Vector Machines (SVM) were the most commonly used models, followed by Artificial Neural Networks (ANN) and Random Forest. Deep learning-

based models belong to the least commonly used. Most researchers have chosen accuracy as the preferred metric to evaluate model performance. The majority of the studies used a limited number of samples to develop ML models,in fact, only 2 used datasets above 2000 subjects. Besides binary classification between BD and HC, many models have also been proposed to differentiate BD and UD. A meta-analysis conducted in 65 studies analyzed the ML performance in classifying BD patients using various biomarkers against HC and other psychiatric disorders. The single study accuracy (ACC) ranged from 46.4% to 100% [16]. The authors concluded that biomarkers that enable a good classification of BD were: global alteration in functional and structural connectivity, cognitive deficits in attentive and reward-seeking domains, and genes and peripheral biomarkers related to immune-inflammatory response. The highest classification accuracy was found to be associated with Logistic Regression and ANN, although is reported that the ML algorithm should be chosen according to the specific type of marker. Overall classification accuracy for BD in this meta-analysis was 0.77.

### 1.4.1 ML for BD diagnosis using MRI-based brain features

From the same meta-analysis, 24 studies used sMRI, reaching a classification accuracy of BD ranging from 54.8%-100%. As well, some of the most discriminative features found were grey matter and white matter alterations in the cortico-limbic network, reporting an ACC discriminating BD vs HC ranging from 59% to 78% [16].

From all the studies that have reported results in terms of ML with sMRI neuroimaging data, the two leading to the most robust evidence were based on big samples resulting from the integration of samples across multiple sites. Among them, one study was conducted with 853 BD patients and 2167 HC from 13 different sites, reporting an AUC of 0.71 and ACC of 65.23% [17]. Furthermore, an ACC of 66% was reported by the other multicentric study, employing more than 1000 subjects [18].

# 2. Machine Learning Overview

Machine Learning (ML) is the study and development of algorithms that allow machines to learn from data some specific rules and patterns, without human intervention, to complete an assigned task. Several tasks can be solved with Machine Learning. From Classification, where the computer learns to label observations to specific classes, either binary or multi-class labeling, to Regression, where the computer learns to estimate a value for a continuum numerical outcome of interest given some set of input variables by finding the function that best describes that relationship, to Anomaly Detection, where the computer is asked to learn to spot and flag abnormal events. Most commonly, the ultimate purpose of a given task is prediction, either of a label for categorical variables, or of a continuum value in a regression form, but can also be mainly interpretation. The models used in ML can be of empirical bases, like Classification Trees, or of multivariate statistics bases, like Regression and Bayesian classifiers.

Machine Learning Algorithms can be divided into different categories according to the type of learning they employ. There are algorithms based on supervised (including self-supervised), semi-supervised and unsupervised learning. Supervised Learning is when the true outcome of a set of explanatory variables is given to the algorithm, allowing it to learn based on all data available and known outcomes. The goal is to accurately predict unseen future observations. When this label is not provided, meaning there are no predefined classes, the learning scheme is called Unsupervised. The latter is used to learn properties of the structure of the data, meant to retrieve similarity/homogeneity rules for example, either through clustering of observations, association rules, etc. Semi-supervised learning combines both by having a small amount of labeled data in the training set that is used to label a large amount of unlabeled data, which is then re-used in a new training process. Within ML, there is a differentiation between the classical models and Deep Artificial Neural Networks. The latter is called Deep Learning, considered to be more powerful, although needing more training data, it can overcome many ML limitations, such as the need for feature engineering, and the ability to learn abstract representations of data leading to better generalization performances, and the ability to learn any

function according to the Universal Approximation Theorem (UAT). The advantages of Deep Learning will be discussed later on in section 2.4.

## 2.1 Building a Machine Learning Algorithm: An Overview [19]

### 2.1.1 Learning Process and Optimization Algorithms

To develop a machine learning algorithm there must be an optimization algorithm that supports the learning process. The learning implies that the model should incrementally improve its performance on a given task, given a set of observations from which to draw some experience. In supervised learning, the learning process develops through the minimization of an error, called loss, which leads to the update of model weights -parameters- in the direction that will minimize that error. The most basic notion of the error can be the deviation of the estimated prediction from its true value. This learning process is called the training phase. Once the model has been realized, a test phase takes place to assess the performance of the model to unseen data samples. In this process, an estimation of the generalization error is retrieved, i.e., the expected performance of the model on future data observations.

A validation phase is also needed when model optimization is performed. Using a disjoint set of data for the validation set, one can evaluate the performance of several models within a model class, i.e., by setting the model with different hyperparameters combinations and retrieving the one that yields better validation performance. This process is called hyperparameter tunning, where hyperparameter stands for parameters that are used to control the learning process. Hyperparameters are divided into model and algorithm hyperparameters. The model hyperparameters are those that cannot be derived via the training process, as opposed to model weights, which are indeed the learned model parameters. The algorithm hyperparameters are those that will influence the speed and efficiency of the learning process.

The learning process is an iterative process, where, an $n$ number of observations are fed to the model from the training set and in several repetitions. The amount of observation that the model is allowed to see each time is called batch size and the number of repetitions, called epochs, is the number of times the model sees the entire training set. The batch size can go from a single observation to the entire training set size. To feed the model with one single example for each new weight update of the model can give rise to a very erratic training process. Conversely, feeding the entire training set can be unviable to fit in memory, if the training set is too big. Finding an

in-between reasonable batch size can be crucial for learning process stability. Usually, small batch size is preferred as it can offer a form of regularization, i.e, constraining model complexity, mainly because the batch will tend to be noisier thus contributing to prevent overfitting to the training samples. The number of epochs and batch size are two examples of algorithm hyperparameters that can be tuned.

During the training phase, the model weights are updated, gradually, and with the experience that the repetitions and exposure to examples allow. This weight updating scheme is done by minimizing an error function, called Cost Function, through an optimization algorithm. The cost function is the cumulative error calculated by averaging the loss function value for each data observation. There are many loss functions to choose from, and the criterium depends on which model is being used and for which purpose. For linear regression, Equation ( 2.1 ), Least Square Error (LSE), Equation ( 2.2 ), is commonly used as loss function, whereas the cost function would be the Mean Square Error (MSE), Equation( 2.3 ).

$$\hat{y} = \beta_0 x_0 + \beta_1 x_1 + \cdots + \beta_j x_j \qquad (2.1)$$

$$j: number\ of\ features$$

$$LSE = (\hat{y}^{(i)} - y^{(i)})^2 \qquad (2.2)$$

$$MSE = \frac{1}{m}\sum_{i}^{m}(\hat{y}^{(i)} - y^{(i)})^2, \qquad (2.3)$$

$$where\ m: number\ of\ observations\ , i: ith\ observation$$

The objective function, the function to minimize in the training set, is, therefore, the MSE. The process is to search for the weights, β coefficients in this case, which will lead to a reduction in that error. This can be done by solving for where the gradient of the MSE is 0. In fact, the LSE leads to a convex optimization problem, most appropriated to find the global minimum through the gradient. This optimization algorithm is the Gradient Descent Algorithm (GDA). There are many types of Optimization Algorithms depending on which types of models are used, and the goal is to find the set of parameters that result in a minimum function evaluation. Optimization Algorithms that used derivatives are appropriate for differentiable objective functions. The gradient is just the first-order derivative of a multivariate continuous function, Equation ( 2.4 ), and gives information about the rate of inclination of a slope, which tells us how to change the model weights to improve the

error value. The minimum point on the objective function will yield a null derivative value. If in the parameters search space, we are far away from that point and the calculated gradient is a positive value, we know then that there must be a decrease in the parameter value, -moving it in the opposite sign direction of the derivative- Equation ( 2.5 ), to lead the cost function to its minimum point as seen in Figure 2.1 with the red vector. This algorithm however can pose problems with local minima and saddle points.

$$\frac{\partial MSE(\beta)}{\partial \beta_j} = \frac{2}{m}\sum_{i}^{m}(\hat{y}^{(i)} - y^{(i)})x_j^i \quad (2.4)$$

$$\beta_j^{new} = \beta_j - \eta\frac{1}{m}\sum_{i}^{m}(\hat{y}^{(i)} - y^{(i)})x_j^i \quad (2.5)$$

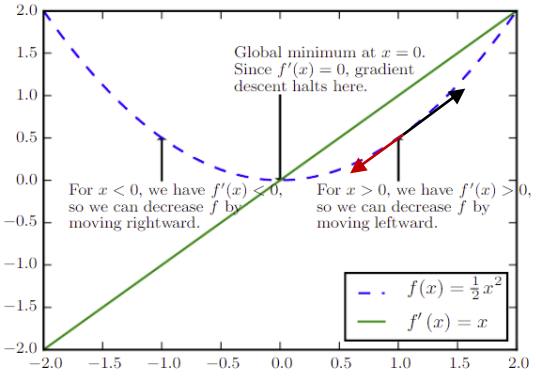



Figure 2.1: Gradient Descent Algorithm.[19]

Equation ( 2.5 ) brings a new element to the discussion, the η coefficient, called the learning rate. The learning rate is a scaling parameter, going from 0 to 1, that determines the size of each step by scaling the gradient vector. An update step with an inappropriate -too big- gradient scaling vector may cause the algorithm to miss the minimum value altogether, leading it to non-convergency, therefore, the learning rate is another algorithm hyperparameter that needs to be tunned since it influences the algorithm speed and ability to converge.

Besides GD, there are other kinds of optimization algorithms, such as second-order optimization algorithms that are based on second-order derivatives or other different

kinds. Specifically, decision trees models have cost functions that contain flat regions. Therefore, the optimization problem is formulated in a completely different setting. They use an inductive greedy algorithm, through a recursive procedure where a split function is optimized one node at a time, using a splitting criterion. The objective function to be optimized is local, the splitting criteria, is usually information gain or Gini index, which is maximized by selecting predictors that yield the highest-scoring split.

### 2.1.2 Regularization Techniques

Regularization techniques are a very important part of the optimization process as they allow to modify the learning algorithm according to a set of preferences. It is intended mainly to reduce generalization error and not training error. For that purpose, simpler solutions are preferred, regarding model complexity.

One way to do this might be to restrict the hypothesis space, for example, by decreasing the polynomial degree allowed for a regression problem. However, this does not stand for a regularization technique because it does not modify the learning algorithm.

Another way to apply regularization techniques is to consider the weight's absolute value. Each feature of our data will have a weight attributed in the model, so the weights can be thought of as the strength or importance of that specific feature. If we constrain the weights to be small this leads to solutions, map function between inputs and outcome, that have smaller slopes. The idea is that the simplest solution is f=0, a function that attributes 0 to all inputs, and so one can measure complexity from the distance to 0. This is called weight decay and is a commonly used regularization technique. Adding this criterion to the optimization process is a form of controlling for overfitting, expressing a preference for simpler solutions that still fit well to the training set. It is done by adding a penalty term to the cost function with a controlling parameter $\lambda$, Equation ( 2.6 ). This controlling parameter forces the weights to be smaller when it's set to be large during the minimization process.

The two most used regularization techniques are the L1 norm, or Lasso Regression, corresponding to the absolute value of the magnitude of the weight, and the L2 norm, or Ridge Regression, corresponding to the square magnitude. The L2 norm can have the advantage of penalizing more large components of the weight vector while L1 norm penalties can be a form of feature selection because end up assigning many feature weights to zero.

$$Objective\_Function\ (w) = Cost\_Function(MSE) + \lambda\, w^T w \qquad (2.6)$$

### 2.1.3 Evaluation Metrics

Finally, the model choice decision will have to be backed up by a quantitative measure, a measure indicating how well the model performs. There are two evaluation metrics to consider, the training loss metric and the generalization evaluation metric. The first tells us whether the model fits well in the training set. The second, is whether the model performs well on unseen data samples, the test set. These metrics can be a measure of error, as the MSE, suitable for regression, or, accuracy, for classification models. To have a clearer picture of model performance one can resort to decision tables like confusion matrix, which give more complete and comprehensive information on different types of errors committed by the model, such as sensitivity, the true positive rate, and specificity, the true negative rate. Another commonly used evaluation method is the Receiver operating characteristic (ROC) curve,Figure 2.2, and the corresponding Area Under the Curve (AUC). This curve chart allows for evaluation accuracy, assuming a probabilistic classification output, without settling to a specific threshold. For binary classification, the probability of belonging to one of the classes is between 0 and 1. For a given sample within the test set, a probability of belonging to a class will be attributed, and, iteratively, different thresholds for that boundary decision will be tried out, defining whether that observation would fall in class 0 or 1. For a given threshold, the resulting true positives rate (tp), Equation ( 2.7 ), and false positives rate (fp), Equation ( 2.8 ), are calculated and a point corresponding to that pair (tp,fp) is assigned in the chart. After testing for a range of boundary decision thresholds, which are not explicitly seen in the chart, the ROC curve is complete and its underneath area is measured [20]. The best possible scenario, the ideal ROC, is when the true positive rate is 1 and the false positive rate is 0, meaning is perfectly capable of separating the two classes. This gives an area of exactly 1, an 100% chance that the model can distinguish between the 2 classes, the blue line in Figure 2.2. The worst-case scenario is when the curve is settled on the chance level, so an AUC of approximately 0.5, red dot line in Figure 2.2. That would mean the classes completely overlap and that the model assigns samples to a class in a completely random manner. When the AUC is under 0.5 and tends to 0, the model is reciprocating the classes.

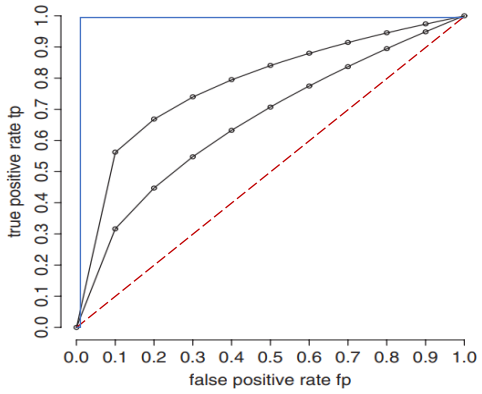



Figure 2.2: ROC curve [20].

$$tp = \frac{Correct\ Postives}{Incorrect\ Negatives + Correct\ Positives} \quad (2.7)$$

$$fp = \frac{Incorrect\ Positives}{Correct\ Negatives + Incorrect\ Positives} \quad (2.8)$$

## 2.2 Important Trade-offs

### 2.2.1 Bias-Variance Trade-off

The idea behind any ML model for classification is to learn to generalize well to new unseen observations. The challenge posed is to learn sufficiently well, that it does not overfit the data from which it has learned, and still achieve a good performance on new observations. Overfitting implies a large variance in the predictions, where variance stands for how much the prediction would vary for a specific data point in different realizations of the model. It also implies low bias, where bias stands for the error between the true data-generating function and the optimal model estimate of that function, indicated in Equation( 2.9 ). The expected test error, for a square error loss, is then decomposed into bias, variance, and irreducible noise term contributions reported in Equation( 2.10 ) [21].

$$Bias\left(\hat{f}(x_0)\right) = E\left[\hat{f}(x_0)\right] - f(x_0) \quad (2.9)$$

$$E(y_0 - \hat{f}(x_0))^2 = Var\left(\hat{f}(x_0)\right) + [Bias\left(\hat{f}(x_0)\right)]^2 + Var(\varepsilon) \quad (2.10)$$

The bias-variance tradeoff is intrinsically related to model complexity, sample size, and prediction error. Model complexity can be thought of as the flexibility of the

estimated function that maps the intended inputs to the outcome. It is related to the non-linearity of a model and the number of its parameters, but not only. A used concept and definition of model complexity is given by the Vapnik-Chervonenkis dimension which quantifies how many data points a function can shatter (or separate). The variance rises with model complexity while bias declines. Thus, complex models yield higher variance and low bias which can lead to overfitting, while simpler models are represented by low variance and high bias which can lead to model underfitting. The optimal model should balance the bias and variance to achieve the minimum prediction error which implies the choice of model complexity based on average test error, amounting to the so-called bias-variance trade-off. The sample size will influence positively the trade-off because large sample sizes allow for having more complex models without necessarily leading them to overfit.

### 2.2.2 Complexity-Interpretability Tradeoff

Another challenging trade-off is that of model complexity and interpretability. Indeed, as model complexity increases, interpretability ability decreases. Interpretability of an ML model stands for the ability to retrieve meaningful rules and identify regular patterns in data, through a final model analysis, that can be understood by experts.[20] The relationships derived should increase the level of knowledge and understanding of a specific system. There are plenty of models, each suited for different complexity requirements. The models that allow achieving higher levels of complexity are also those which interpretability is harder to achieve. Not always a model that was developed with a prediction purpose will yield meaningful interpretations. Indeed, this trade-off translates into a sometimes-mutual exclusive relation between prediction performance and model interpretability. A model yielding high accuracy prediction might not yield meaningful interpretations while a model with a moderate above-chance accuracy level can yield a lot of increment utility when it comes to understanding a certain system. Therefore, the evaluation metric and method used to draw conclusions on model performance depends on the defined goal for that specific model. If it's prediction, accuracy solely can be an appropriate metric, if it is interpretation, other metrics and analysis should be employed.

## 2.3 Support Vector Machines (SVM)

As mentioned previously, SVM models are the preferred models to use with neuroimaging data. These models have a powerful mathematical foundation and excel in many diverse applications. They have outperformed other classical ML models and for this reason, have become the most successful and preferred

classifiers. SVMs are a class of models for supervised learning, either classification or regression. What they do is find the hyperplane that best separates two classes in the feature space.

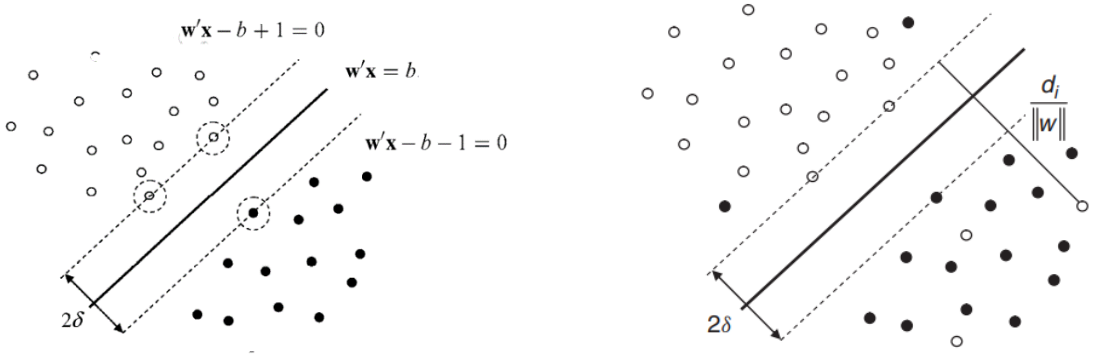
The hyperplane is built and supported by specific examples from the input dataset, called support vectors, that define the position of this separating surface. In a sense, the support vectors can be understood as the most representative observations for each class. The goal of SVM is to find the maximal separation margin $\delta$, defined in Equation ( 2.11 ), i.e, the hyperplane furthest away from examples of the 2 classes, allowing for a bigger gap between them. The optimization problem is formulated as the minimization of the reciprocal of the margin of separation. For the linear separable problem, it just constructs an optimal linear decision boundary, as seen in Figure 2.3 a). However, the optimal linear separation can lead to poor generalization capability to new unseen observations due to overfitting to the current data points and not allowing for a sufficiently reasonable wide margin where the new data points could fall in. Besides, in most cases, data is not linearly separable, therefore, to handle the last two mentioned limitations, the SVM can 1) soften the concept of separation - soft margin - and 2) map data with a kernel function into a higher dimensional space to achieve the needed linear separability, this is called the kernel trick.

The first approach defines a soft margin by allowing some misclassifications errors to occur. For this purpose a loss function on the violation of the linearly separable constraints is introduced, the hinge loss, described in Equation( 2.12 ), where $(w'x_i + b)$ , also denoted by, $\widehat{y_i}$ , corresponds to prediction (either 1 or -1) and $y_i$ to the true classification of the $i^{th}$ data point. The hinge loss ignores correct classifications, allows data points to be misclassified so as to have a linear solution to the optimization problem but penalizes this violation proportionally to its severity, i.e., all points falling on the wrong side of the separation hyperplane will have a positive hinge loss that increases linearly with the point distance to the correct margin side. For example, a point belonging to class 1 predicted as 0.3, will have a hinge loss of 0.7.

To represent the hinge loss in the optimization problem a slack variable is introduced. The slack variable measures the distance from the data point to the corresponding class margin, denoted by $di$ in Figure 2.3(b), and is added to the objective function with a regulating parameter C. This measure incorporates then the severity of misclassification, the furthest away a data point is from the right side of the class margin, the higher will be the measured distance $d_i$, and so the worst the penalty will be. C is a model hyperparameter that allows controlling the importance we give to misclassifications. If C is small, misclassifications are given less

importance and as a consequence, a higher rate is accepted and a wider margin separation is achieved.

The final objective function and the corresponding optimization problem are defined in Equation ( 2.13 ) incorporating the penalty term, corresponding to a Constrained Optimization Problem. [22]




a) Canonical Hyperplane. b) Soft Margin.

Figure 2.3: Soft Margin Definition.[20]

$$\delta = \frac{2}{\|w\|}, \quad where \quad \|w\| = \sum_j w_j^2 \tag{2.11}$$

$$hinge\ loss: L = \max(0, 1 - y_i(w' x_i + b)) \tag{2.12}$$

$$min_{w,d} \frac{1}{2}\|w\|^2 + C\sum_i^m d_i \tag{2.13}$$
$$s.to\ y_i(w' x_i + b) \geq 1 - d_i, \quad i = 1, \dots, m$$
$$d_i \geq 0, i = 1, \dots, m$$

Although the arguments presented previously are a very efficacy formulation to circumvent the nonlinear separable problem, not always the soft margin formulation will be enough to overcome that limitation. Indeed, for certain tasks, no matter which

C value is chosen, the model will always underperform because the data is intrinsically characterized by nonlinear patterns so no linear separation can be found to be satisfactory.

This brings us to the kernel trick, which constitutes a way of enlarging the feature space through some transformations and achieving a linear separation in that transformed space. This results in a nonlinear decision boundary in the original space, but the advantage is that the problem is formulated in the transformed space and so it allows us to search for a nonlinear separating function while solving a linear optimization problem.

A kernel function is a generalized function that measures similarity between 2 vectors by outputting a respective score, Equation( 2.14 ), where φ denotes a given transformation and the dot product, <φ,φ> , a the kernel K. It can be shown that a linear function can be re-written with a kernel function, simply because the function can be written in terms of dot products between examples, Equation ( 2.15 ) [19]. As we can see, the relationship between f(x) and K and $\alpha$ is linear, even though is nonlinear with respect to x. An example of the utility of nonlinear SVM can be seen when data is structured in a somehow circular pattern in the 2D space, and the decision boundary that is best suited for class separation is therefore radial. The way nonlinear SVM would solve this problem is by starting to map the input data to a higher-dimensional space. The coordinates are projected to a new reference system that separates observations in a way that a linear hyperplane fits in as a decision boundary, Figure 2.4.

$$K(x) = \varphi(x).\varphi(x) \tag{2.14}$$

$$f(x) = b + \sum_{i}^{m} \alpha_i K(x, x^i) \tag{2.15}$$

Regarding the optimization algorithm of SVM, we are faced with a Constrained Optimization Problem, and to overcome the difficulty this imposes, the method of Lagrangian multipliers is used to convert the problem into an unconstrained optimization problem by including the constraints into the objective function. By using the Lagrangian Method the dual optimization problem is derived, and it is to this objective function that an optimization algorithm is ultimately applied.

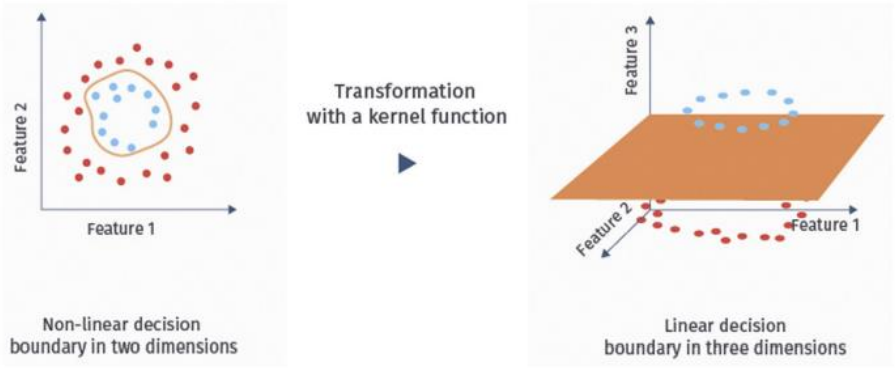



Figure 2.4: Kernel Transformation [23].

Many optimization algorithms can be applied, including the ones belonging to the family of Gradient Descent Algorithms. By solving the optimization problem, we will be able to identify the support vectors and classify new instances of data by using the resulting decision function. For the nonlinear SVM optimization process, the difference resides in simply substituting x with $\varphi(x)$, the kernel transformation, in the Lagrangian formulation.

The key difficulty of nonlinear SVM is the requirement of meaningful kernel functions that enable an efficient transformation. The major drawback is the high computational cost of training when the dataset is large. Indeed, the quadratic form of kernel matrix requires memory that grows as well with the square of the number of data samples, being the computational cost $O(m^2)$ [19].

## 2.4 Deep Learning

Deep learning is a form of representation learning which in turn is a form of machine learning and all are within the Artificial Intelligence (AI) field. It was designed to overcome kernel machines' limitations, respectively 1) Lack of meaningful kernels and 2) Computational Cost on large datasets. The main problem that classical ML models were facing was the difficulty to generalize well when working with high-dimensional data, also known as the curse of dimensionality.

Usually, in a low-dimensional space, a finite number of training samples will be able to represent more or less each and every possible combination for a given variable, then, when generalizing to an unseen observation it simply needs to inspect the training examples that are highly representative of the new sample. In high-dimensional spaces, because the number of possible configurations is huge, the training data is insufficient to represent well every possible combination, which leads

to the problem of having to predict for a new observation for which a variable configuration has never been seen [19]. This problem is also observed in the difficulty that many ML applications suffer regarding the ability to extract high-level abstract features that allow generalizing well for the assigned task. For example, being able to recognize an object should be independent of its position, orientation, illumination, color, etc.

The reason why DL can overcome the formerly mentioned limitations is that compared to kernel algorithms, which use a generic pre-defined mapping function, the kernel function, the strategy of deep learning is to learn the mapping function itself $\phi(x)$ from a broader class of functions [19]. Also, its hierarchical nature, makes it well adapted to learning hierarchies of knowledge by expressing complex representations in terms of simpler ones. For example, it can learn to recognize an object by learning simple concepts like corners and contours to then combine them into edges and so on arriving at a high-level representation that has a much higher generalizability performance compared to what other models are capable of. The ability to learn the mapping function is related to the Universal Approximation Theorem (UAT) which will be explained in more detail in the next subsection.

Therefore, a Deep Learning model is capable of extracting features from raw data, different from the feature engineering perspective where a feature is designed manually in a preprocessing stage. We talk about Deep learning when an Artificial Neural Network (ANN) model has a network depth of more than 3 layers (including input and output layers). The basic instance of an ANN, a single-layer network, is called perceptron, is constituted by an input and output layer, and is considered a classical ML model. The power is unleashed by combining many of these units together by increasing the depth of the network.

### 2.4.1 Artificial Neural Networks

Artificial Neural Networks emulate a biological neural network by simulating neuron interactions through the electrical activity that gives rise to the processing of information. Biological neurons are connected through synapses, gap regions that exist between the axon of one and the dendrite of another. The artificial neuron model is an abstraction of the biological neuron. It is a computational unit that maps the inputs into one output signal, also called its state. Can be seen as a non-linear transformation unit that takes a weighted summation of inputs, called the Action Potential denoted by $z$, and transforms it through an Activation Function (AF), denoted as $\vartheta(z)$, which can be linear or non-linear, Equation ( 2.16 ) [24]. The weights, denoted by $w$, simulate the synaptic strength of a connection between neurons and

the learning process is realized by the changing of this weight through the presence of an external stimulus, emulating the plasticity of brain neural networks.

The simplest ANN architecture is a single-layer network, also called perceptron, Figure 2.5, composed of the input layer and output layer. In multi-layer neural networks, these instances are stacked together and arranged in a layered architecture, called a Feed-Forward Network (FNN). Not only, in an FNN the neurons are fully connected, and, as the name already hints, the connections are in the forward direction only, with no feedback connections. All layers within the input and output layers are called hidden layers. Considering the perceptron for a binary classification task, to determine the 0 or 1 class output, the weighted sum has to overcome a given threshold value so that the Activation Function can map it to 1, otherwise, the output is 0. The described process is simulating the threshold that needs to be overcome in a biological neuron membrane so that an Action Potential can be propagated. The threshold, called bias $b$, can be added to the neuron as if it is a weighted input itself. The trick is just considering a positive unit input and bias $b$ as its weight. In this way, the threshold to overcome becomes 0 but the mathematical description becomes simpler, $z \geq 0$.

The AF representing this transformation, $\vartheta$, could be the Heaviside step function. It is appropriated because it bounds the input in the saturation domain, allowing to stabilize the Action Potential signals. The problem with the simplistic Heaviside step function is that a small change in the weights can lead to a big change in the output, i.e., the output completely flips undesirably. Besides, this step function is non-differentiable which can lead to several limitations during the learning process, regarding the optimization algorithms. The sigmoid function can be seen as a differentiable function which is approximating the step function. Its smoother shape enables the necessary behavior: that a small change in the weights concomitantly causes a small change in the output [25]. Indeed a balance needs to be found between the change in the weights and a change in the error of the model. Ultimately, AFs are chosen according to the strengths of their intrinsic properties.

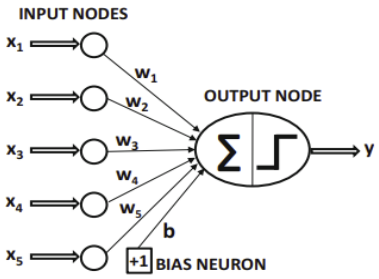



Figure 2.5: Perceptron[26].

$$y = \vartheta(z)$$

$$z = \sum_k w_k x_k + b \tag{2.16}$$

As discussed in previous chapters, the learning process will consist in learning the weights so that an objective function is minimized. This objective is a cost function, denoted by J, a proxy for the output error given by the model. The GDA is a suitable algorithm to figure out how to change the weights so to improve model performance. The gradient of the cost function must hint at which direction to update the values of the weights. An effective learning process happens when the AF changes its output whenever we have a meaningful update in weights. If the AF is insensitive to a change in weights it becomes hard to realize in which direction to evolve the weights, leading to a stagnation in the number of misclassified observations, thus obstructing the improvement of model performance. A weight update is given by the form of Equation ( 2.17 ), where $\nabla J$ is the gradient of the cost function, the partial derivative with respect to weights, Equation ( 2.18 ). To solve the optimization problem for any ANN for binary classification task one must decide on the AF and the loss function. For a simple example on a single-layer ANN, one can consider that the cost function is simply the MSE, $J = \frac{1}{2}(\hat{y}_i - y_i)^2$, as previously in Chapter 2.1 and the AF the sigmoid function. The derivation of the cost function gradient, ∇J, for the gradient descent algorithm, can be seen in Equation ( 2.19 ), where delta $\delta$ denotes $\frac{\partial J}{\partial y}\frac{\partial y_i}{\partial w_{ik}}$, which integrates the Equation ( 2.17 ), called delta rule.

For each training epoch, the gradient descent algorithm needs to calculate the gradient of the cost function, which means calculating the gradient of the loss function for each i$^{th}$ observation and averaging the $m$ total samples. For a large training size, it can be unfeasible to use all samples to take a single updating step, given the time that it would take to compute this average considering all m observations at once. An improvement on this batch GD is the mini-batch Stochastic Gradient Descent algorithm (SGD). The mini-batch SGD provides a modification by considering a randomly picked small batch from the training set which is used to estimate the true gradient of the cost function. It can be shown that for a large sample size $m$, the estimate is a good approximation of the true overall cost function gradient [25]. Thus, for each epoch, several steps on the weight update rule will be made, each based on a randomly picked small batch of training data until all the training samples have been exhausted.

$$w_k^{new} = w_k + \Delta w \qquad (2.17)$$
$$where, \ \Delta w = -\eta \nabla J$$

$$\nabla J = \left(\frac{\partial J}{\partial w_1}, \dots, \frac{\partial J}{\partial w_k}\right) = \frac{\partial J}{\partial y}\frac{\partial y}{\partial w_{1,..,k}} \qquad (2.18)$$

$$\frac{\partial J}{\partial w_{ik}} = \frac{\partial\left(\frac{1}{2}(\hat{y}_i - y_i)^2\right)}{\partial y_i}\frac{\partial y_i}{\partial w_{ik}} \rightarrow$$
$$\rightarrow -(\hat{y}_i - y_i)\frac{\partial y_i}{\partial z_i}\frac{\partial z_i}{\partial w_{ik}} \rightarrow$$
$$\rightarrow -(\hat{y}_i - y_i)\vartheta'(z_i)\, x_i$$
$$\Delta w = \eta\, \delta\, x \qquad (2.19)$$

The Universal Approximation Theorem states that any continuous function on a closed and bounded subset of $IR^n$ can be approximated by a neural network, featuring a nonlinear squashing hidden unit and a linear output layer [19]. This is a very powerful statement, affirming that an ANN will be able to represent any function that it's trying to learn, even if it's not able to find it in practice due to limitations on the training algorithm. Besides, theoretically, for a single-layer ANN, the degree of error to which we can approximate the true function can be defined at any nonzero amount and is only constrained by the number of hidden units. To achieve the desired accuracy, however, the number of necessary hidden units can be prohibitively large and also lead to an overfitting problem. Increasing the depth of the network, i.e., the number of hidden layers, the number of hidden units required to represent the given function will be reduced, thus the great power that comes with Deep Learning.

### 2.4.2 Learning Process for DL: Backpropagation

There is however one thing missing to complete the learning process overview for a deep ANN, i.e., with some hidden layer(s). The former explanation suits well for a single-layer ANN, Figure 2.5, given that we are in a supervision framework, thus $y_i$, the true label of an observation, is used in the loss function to calculate misclassifications/errors and further guide the learning process to change the weights to correct for these misclassifications. In the case of Deep ANN, the hidden neuron's weights also need updating, however, there are no explicit supervision labels to directly apply the delta rule, Equation( 2.17 ). Hence, the backpropagation concept is introduced, which will address the formerly mentioned gap in the weight updating rule.

Backpropagation is a method developed to efficiently calculate the weights which later are used in the gradient descent optimizer (or other). The weights will be denoted from now on as $w_{kj}^l$ and the activation value of a neuron as $a_j^l$ , i.e., a hidden neuron output, where l denotes the $l^{th}$ layer, k the $k^{th}$ neuron in the $(l-1)^{th}$ layer and j the $j^{th}$ neuron in the current $l^{th}$ layer. Regardless of the network depth, the rationale

behind the computation of backpropagation can be visualized in Figure 2.6, in a simplified toy ANN.

If the input layer is layer l, then the first hidden layer will be l+1 and so on, until the output layer which is denoted by L. The weight update is based on the gradient descent method.

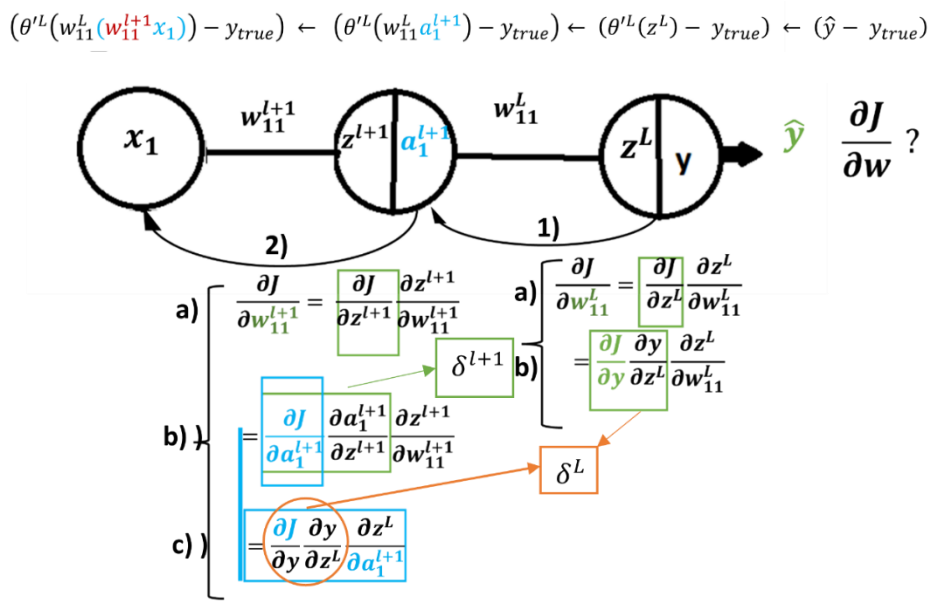



Figure 2.6: Chain rule in Backpropagation.

$$\boldsymbol{\delta^{l+1}x_1} = \boldsymbol{\delta^L}\, \boldsymbol{w^L}\, \boldsymbol{\theta'^{l+1}(z^{l+1})x_1} = \frac{\partial J}{\partial y}\frac{\partial y}{\partial z^L}\frac{\partial z^L}{\partial a_1^{l+1}}\frac{\partial a_1^{l+1}}{\partial z^{l+1}}\frac{\partial z^{l+1}}{\partial w_{11}^{l+1}} \quad (2.20)$$

Since a known true label doesn't exist for activation $a_1^{l+1}$, The cost $J = \frac{1}{2}(\hat{a}_1^{l+1} - a_1^{l+1})^2$ cannot be explicitly calculated nor its partial derivative for the GD algorithm. In fact, we can see that the cost function dependency on $a_1^{l+1}$ is not direct, which is represented in the equality on the top of Figure 2.6. This already suggests that a further derivation will be necessary to understand how the weight on hidden layer l+1 will be updated.

The learning process will evolve in the following way: firstly, there is a forward step, where weights are initialized according to a chosen criteria and the training set is passed layer by layer until we get the output predictions for each observation, $\hat{y}_i$. From the output layer, the cost function can be calculated based on the misclassifications which result in an error value. The last layer weight, $w_{11}^{L}$ , can be easily updated by applying the delta rule since we do have an explicit label, $y_{true}$ , which corresponds to the derivations on points 1) a),b) in Figure 2.6, similarly to what was shown previously in Equations ( 2.17 ), ( 2.18 ) and ( 2.19 ). From this point on, a backward step initializes where the error will be propagated back to the input layer and the weights will be iteratively updated. In Step 2) the backpropagation starts and it must serve to calculate the partial derivative of the cost function with respect to $w_{11}^{l+1}$, which is shown in point 2) c). While point 2) a) and b) follow the same dynamics as 1) a),b), with the gradient of the Cost function being calculated with respect to a different layer weight, l+1 instead of L, a further step arises in sub-step c), where it is necessary to further apply the chain rule to $\frac{\partial J}{\partial a}$ for the reasons already stated above (there is no explicit computation of cost on the output of $a_1^{l+1}$ hidden layer). This leads us to Equation ( 2.20 ) , the complete rule to backpropagate the error back to layer l+1 in Figure 2.6. It is noticeable that there is a repeating pattern, and in Equation( 2.21 ), the general cost function derivation on a hidden neuron output, o, can be seen. The general rule for weight update, $\Delta w_{kj}^{l}$ , in a layer l<L is stated in Equation ( 2.22 ), where we can see that $\delta^{l}$ unfolds to the product between weights and $\delta$ delta of the next layer, l+1. The chain rule is recursively applied until input layer.

$$\frac{\partial J}{\partial o_j^l} = \frac{\partial J}{\partial z_j^{l+1}} \frac{\partial z_j^{l+1}}{\partial o_j^l} = \frac{\partial J}{\partial o^{l+1}} \frac{\partial o^{l+1}}{\partial z_j^{l+1}} \frac{\partial z_j^{l+1}}{\partial o^l} \tag{2.21}$$

$$\Delta w_{kj}^{l} = \eta \, \delta_k^{\,l} x_j^{l-1} \tag{2.22}$$

$$\delta_k^{\,l} = \left( \sum_r^R \left( w_{r,k}^{l+1} \right) \delta_r^{l+1} \right) \vartheta'\left( z_k^l \right), \quad for\ l < L$$

$$where\ R\ is\ the\ number\ of\ hidden\ units$$

### 2.4.3 Optimizing the Learning Process

For the sake of optimizing the learning process, several modifications or improvements can be made at different levels in our model. From the optimizer in use to the choice of Activation Function, the possibilities are endless.

#### *2.4.3.1 Choice of Activation Function (AF)*

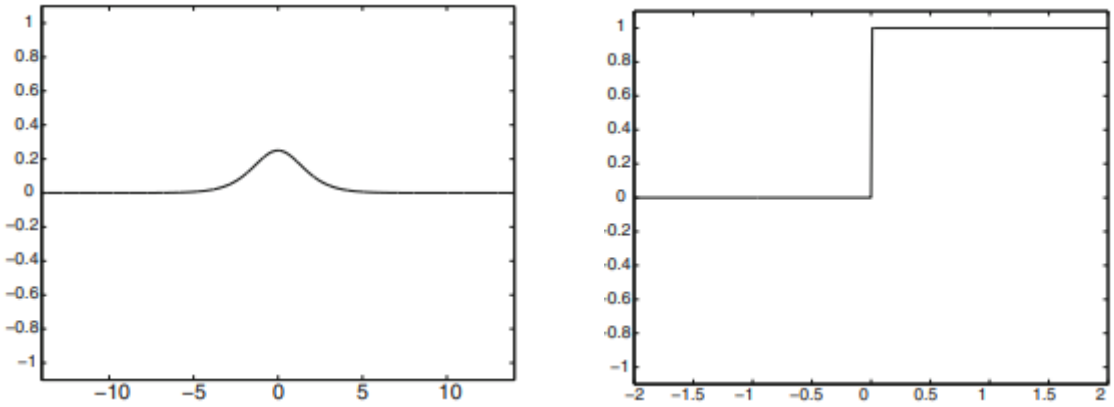
The choice of AF plays an important role in the efficiency of the training since there are some known limitations characterizing each of them. An overview will be made of the Sigmoid function, Rectified Linear Unit (ReLU), and Scaled Exponential Linear Unit (SELU).

One problem that can arise in training an ANN is the vanishing gradient problem (as well as the exploding gradient problem). This problem is intensified with the depth of the ANN and it can be traced back to the way gradients backpropagate. This problem can be identified when using sigmoid AF to train deep ANN.

The sigmoid function saturates mostly for very high and low values of input $z$, and is most sensitive when z is near 0, although its derivative is less than 0.25 in the entire domain, $\vartheta'^{(z)}$, Figure 2.7 a). When the calculated gradient is small in beginning, the later gradients will be increasingly small, due to the chain-like product computation in backpropagation. Consequentially, earlier layers learn much slower (or nothing at all) than late ones. Layers learning at different speeds is undesirable since the goal is to optimize all layers, therefore, if the gradient is vanishing, the gradient-based learning becomes very difficult and the weights result stagnated. On the contrary, if the AF has too large gradients, the opposite problem can arise, the exploding gradient problem.

The ReLU AF is defined as *max(0,z)*, being 0 for negative values of input and the identity function for positive values. It overcomes the vanishing gradient problem and is usually accepted as a default choice for an ANN. It is noticeable that it contains a discontinuity at z=0 however this does not invalidate the use of gradient-descent learning. There are several AF that are non-differentiable but only at some points, and because it is not expected that the ANN training algorithm arrives at a minimum point then this point can also have an undefined gradient. It is enough to ensure that the cost function value is reduced substantially [19]. The derivative of ReLU returns 0 when z<0 and returns 1 when z>1, Figure 2.7b), and because of this behavior we encounter another problem, called the "dead" neuron problem. When z<0, the neuron becomes inactive because the gradient is zero and so the update will also be zero. ReLU cannot learn via gradient-descent methods for samples whose activation is zero.

Various generalizations of ReLU AF were introduced which improve the limitations mentioned before. For example, SELU AF avoids vanishing or exploding gradient problems and the dead neuron problem[27].




a) Derivative of Sigmoid Function.

b) Derivative of ReLU Function.

Figure 2.7: Derivatives of AF [26].

### *2.4.3.2 Choice of Loss Function*

The choice of loss function influences the efficiency of the optimization algorithm. Some loss functions cannot be minimized, as in the case of 0-1 loss. For this reason, loss functions were developed to act as proxies for the one we care about, this is called a surrogate loss function. Besides, as was mentioned previously, the optimization of an ANN is not a pure optimization since we do not care about reaching a minimum point in the training set search space, but to decrease a generalization error. In fact, when training a model, the training is stopped when it has converged to a solution that is decreasing an evaluation metric, and right after overfitting starts to happen, even though the gradient might still be high. This evaluation metric acts as a preview of the generalization error we expect in the unseen test set.

Several loss functions have advantages compared with the simple quadratic cost or 0-1 loss. Some problems already discussed, like the vanishing gradient, can be fixed by choosing an improved cost function, with better gradient characteristics, $\frac{\partial J}{\partial wj}$, which improves the efficiency of the gradient descent method.

For example, the gradient of the cross-entropy cost function has a form that does not pose the problem caused by the quadratic cost, of slowing down learning when using a sigmoid AF. This is because the term $\vartheta'(z)$, responsible for the vanishing gradient problem due to small derivatives with the sigmoid function, is canceled out during the derivation of the final expression of the gradient using cross-entropy cost, Equation ( 2.23 ) [25].

$$\frac{\partial J}{\partial w_j} = \frac{1}{n}\sum_i x_{ij}(\vartheta(z) - y_i) \quad (2.23)$$

### 2.4.3.3 Choice of Optimizer

SGD is a very commonly used optimizer, however, it can be slow for solving certain problems, and because of this, several other optimizers have been proposed that try to improve SGD limitations.

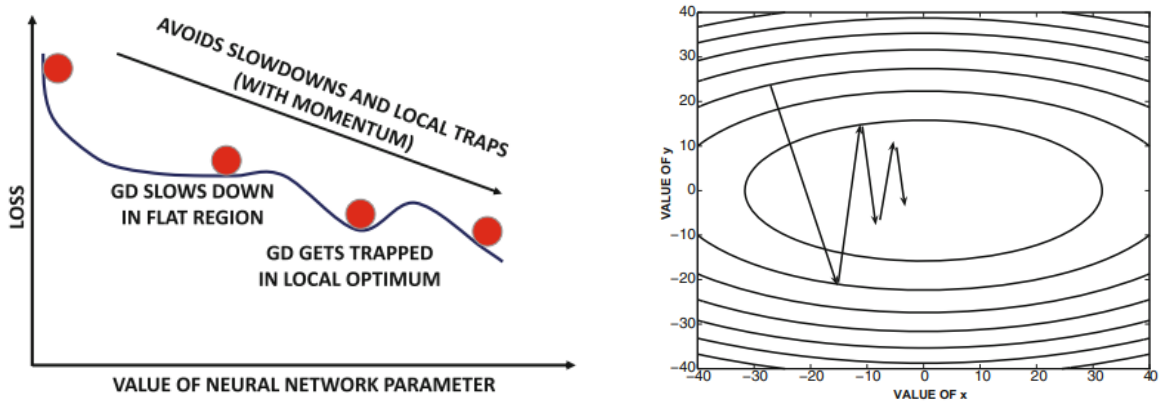
Momentum is a concept of physics that was introduced in the optimizers' methods to accelerate learning. In fact, momentum can be defined as the increase in the rate of development of a process which is translated in the optimizing framework as a model's ability in increasing its learning velocity, the more it learns, the faster it also learns. It introduces a hyperparameter $\alpha$, bound between 0 and 1, that determines how quickly the contributions of previous gradients exponentially decay. The higher the $\alpha$ is with respect to learning rate η, the more previous gradients affect the current direction. If learning is affected more by previous gradients, then, the step size taken during gradient descent becomes larger when many successive observations follow a certain somehow similar pattern (gradient points following the exact same direction). One way to accelerate learning is to combine SGD with momentum, Figure 2.8 a).

Another approach is to adapt the learning rate during training. This can be done through learning rate schedules, which adapt the learning rate throughout the training course. This method does not modify the optimizer. Learning rate influences significantly the model performance and it can be advantageous if it decreases during training so that when reaching a minimum value it takes smaller updating steps. For this, several types of schedules can be used, such as the Exponential Learning Rate Schedule, which exponentially decays the learning rate during training. Of course, this introduces more hyperparameters in the algorithm, such as the ones concerning the schedule itself.

Another perspective is introduced by optimizers that adapt the learning rate for each parameter, such as AdaGrad. In fact, in the search space, the cost can be rather insensitive to some parameters directions, exemplified in Figure 2.8b) and by introducing an individual learning rate for each, the issue can be mitigated and more degrees of freedom are introduced. The parameters with the largest partial derivatives of the loss will have a higher decrease in the learning rate. RMSProp is another adaptative learning optimizer that was introduced as an improvement to AdaGrad. It can be applied to nonconvex functions where AdaGrad fails, by decreasing the learning rate fast only when it finds a convex region [19].

Lastly, Adam optimizer, which is derived from "adaptative moments" is a combination of RMSProp with momentum and it has been proved to perform fairly

better with respect to others [28]. There is no standard method to define which is the right optimization algorithm for a given optimization problem. Adaptative learning rate algorithms seem to perform better but, within that family of algorithms, the choice is more empirically and subjective to the user's experience.




a) GD with momentum.

b) Elliptic Loss Function- more insensitive to x-direction.

Figure 2.8: Optimizers' Methods [26].

#### *2.4.3.4 Parameter Initializer*

Training algorithms are usually iterative which implies the definition of a starting point. This starting point can then influence whether or not an algorithm will converge to a solution. For this reason, several strategies were developed for the initialization of model parameters. One of these strategies is to randomly initialize all weights in the model drawing from a Gaussian or Uniform distribution. Furthermore, the scale of this initialization also seems to be relevant. Initializing weights with too small a value can lead to losing signal during backpropagation and too large weights can lead to the exploding of weights problems or causing AF to saturate causing the vanishing gradient problem. Adding to this, some AF functions require specific parameter initializers such as SELU AF which requires LecunNormal initialization.

#### *2.4.3.5 Dropout Regularization*

Regularization techniques have already been discussed in Chapter 2.1.2. These regularizers can be equally added to the optimization process for DL.

A specific form of regularization for ANN models is the dropout method. Dropout does not modify the cost function, it modifies the network by forcing sparsity. With dropout, some hidden units of the network are shut down during each weight

update in the backpropagation process. This means that a small group of neurons activate to a set of inputs which force these neurons to learn more robust features. Another take on the dropout technique is that dropping out a different set of neurons is like training different ANN at the same time.

## 2.5 Autoencoders

An Autoencoder is a type of ANN used for feature learning, is a type of FFNN designed with a symmetric architecture. The goal of an Autoencoder is to learn the underlying structure of data. It learns how to reconstruct its input from a compressed version of it. This implies that the information present in the compressed version, or code, must be very relevant and representative of the input.

It's composed of two parts, the encoder, starting from the input layer in which layers decrease gradually in breath until the minimum dimension, the so-called bottleneck, where the code for representing the input is presented, also denoted by latent space or latent variables, and the decoder, composed by layers of hidden units mirroring the encoder part, increasing in size leading to the output, which has the same size as the input and should be a successful reproduction of it, Figure 2.9 a).

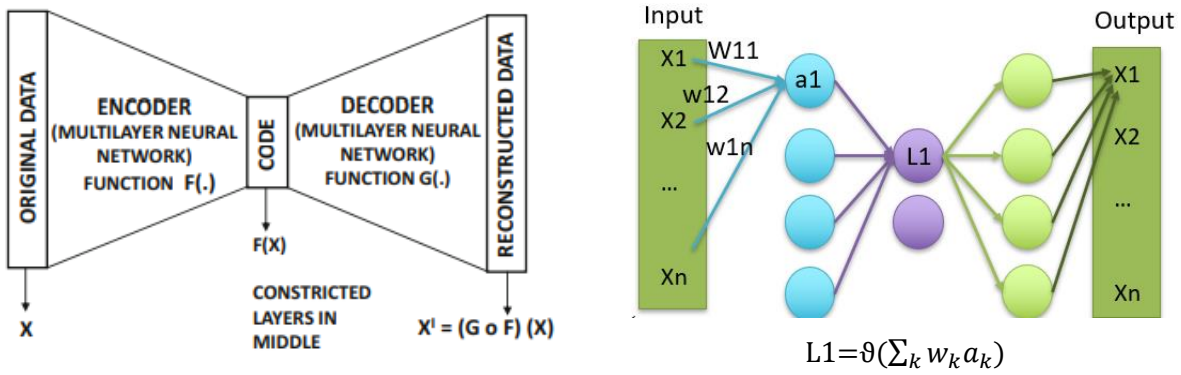
The challenge is to learn to represent the input without copying it, i.e., learning an approximation to the identity function but not the identity itself. This is more or less ensured if the architecture of the AE is designed in the formerly described way, i.e., shrinking the number of hidden units in each layer, called an undercomplete AE, so to force a low dimensionality representation of the input in the bottleneck. However, restraint must be taken in the capacity (complexity) that is allowed to the network. Indeed, an undercomplete AE where the hidden units are allowed too much capacity can still be able to overfit to the training data, for this reason, adding regularization layers might be useful to further ensure the model does not learn to copy the input.

The learning framework of an AE is similar to any other NN within the supervised training, it learns how to reconstruct the input by a self-supervised framework, where the input itself serves as the target variable. The training goal is to minimize the generalization error on the test set between the reconstruction and the original sample.

In terms of the dimensionality reduction ability, there are some counterparts to PCA. An undercomplete AE in which the decoder is composed of only linear units and the loss function is the mean squared error learns the principal subspace of training data, i.e., the same subspace as PCA would [19]. When nonlinear units are chosen instead, AE can learn a more powerful nonlinear generalization of PCA. Indeed, AE can map

an input to an even lower dimensionality than PCA because AE manages to learn the structure of the data manifold.

The basis of manifold learning is the idea that data follows a certain low-dimensional manifold or set of manifolds. A manifold is a connected region, i.e., a set of points associated with a neighborhood around each point [19], which can be understood as a hidden data structure. Many ML algorithms try to learn a function that behaves correctly in the manifold but fail to generalize if an observation is of the manifold.




a) Autoencoder [26].

b) Latent Variables.

Figure 2.9: Autoencoder.

The AE should then be able to recover the manifold structure, which PCA fails to do because manifolds are intrinsically nonlinear- they are curved in the original space because data may be distributed in entangled spirals and complex shapes- and Deep nonlinear AEs are suitable for disentangling such forms. Of course, this capacity also comes with the drawback that an AE may be able to memorize the input, learning a representation that is not useful, therefore it is recommended to make use of architecture constraints and regularization techniques, as mentioned previously.

Besides data compression, the AE can be used to tackle other tasks, such as denoising data, integrating different data types, as a generative network, and anomaly detection. In data compression one is interested in preserving the code, i.e., the latent variables of the input. Thus, once the AE is trained the decoder part can be lost.

The latent variables, however, can be hard to interpret since one does not know exactly what they represent. They are derived as any other hidden unit activation, by applying the AF to the input which represents the output of a neuron of a previous layer, Figure 2.9 b).

In the anomaly detection task, the AE learns how to represent the input, being this input considered the "normal" or within a normative range. When a sample is

deviating from this normative range, presenting some abnormality, the AE fails to reconstruct it well. This is closely related to dimensionality reduction, in fact, anomaly detection is a consequence of an AE which has effectively learned how to represent an input through a compressed representation. A compressed representation of data represents only regularities in data, i.e., all unusual and noisy variations are lost in the dimensionality reduction process. Since an outlier point is mainly characterized by irregularities, the AE will be unable to encode it without losing information [26]. The strategy is therefore to quantify the reconstruction error so to have a score within a sample is considered an inlier and above a certain threshold an outlier.

## 2.6 ML Best Practices

Aiming to build a robust and replicable ML model it is important to respect and follow certain standards and guidelines. From data collection, and data processing to ML pipelines, there are a lot of factors to be considered to avoid overestimating results and misleading conclusions.

In the area of psychiatry and brain disorders, many ML models for diagnosis have been proposed, however, the reported results are rather polarized and ambiguous, giving rise to concerns about the validity of certain findings. Focusing on neuroimaging data for the diagnosis of BD, reported performance results are extremely discrepant, going from 50% to 100% accuracy (ACC) [18]. Taking this into account it is important to overview ML best practices and recommendations in order to develop a model grounded in a robust pipeline.

From the data point of view, some issues might be a source of systematic overestimations, concerning size and sample heterogeneity. There has been a consistent bias where small sample studies report better performances than large sample studies. In fact, an inverse relationship between sample size and balanced accuracy has been reported, a rather counter-intuitive finding since one would expect accuracy to increase with sample size [29]. This finding hints at the fact that subject inter and intra-variability, and population heterogeneity could be influencing lowering model performances. If one is to study BD through a sample of N subjects, within this sample, patients might belong to different subtypes of the disorder, be at different statuses or disease progression, present a variety of different symptoms between each other, and have been submitted to different neuroimaging collection protocols, use of different scanners, etc. Logically, all these factors should contribute to lower model performance since it is harder to learn on a broad heterogeneous data sample. However, a small sample may lack the heterogenicity that is intrinsically presented in the population under study, making it easier for the model to learn and

predict for that sample but failing to generalize to the whole population of interest. Whereas higher sample sizes yield increased statistical power, making heterogeneity a lesser problem and resulting in more replicable findings. Heterogenicity may be represented by demographic variability, site variability, phenotypic variability, and clinical status variability. Since the biological etiology of most brain disorders is unknown, it is not possible to subtype patients by creating well-defined clusters that yield phenotypic homogeneity. Moreover, a collection of data taken place at the same recruiting center will be rather homogeneous whereas a multisite data collection would yield a significantly higher variability.

Apart from sample heterogeneity, size can be by itself a limiting factor, regarding its relationship with model complexity. Sample size shifts the trade-off between bias and variance allowing for the retrieval of complex models with low bias and relatively low variance, Figure 2.10. This means that large sample sizes allow for the realization of complex models without compromising the prediction error [21]. This is why, Deep Learning models require more data to be trained in order to outperform classical ML models, as they have much more parameters to be learned yielding more complex models. With increased model complexity, a small sample size easily leads to overfitting models. It is hard to effectively learn from a few data samples when the learning task is very complex, thus, models end up memorizing rather than learning. Although it is important to keep this in mind, Deep learning models should not be excluded on this basis as some methods and adaptations can be done to try to circumvent this issue. Firstly, using a priori knowledge of methods that facilitate training can reduce the number of to-be-trained parameters and the large dataset size demands. This can include sticking to more shallow network architectures (small number of hidden layers), choosing efficient activation functions (facilitating the training process), modifications to loss functions for example with regularizations techniques that can efficiently decrease model complexity hence optimizing the training process in specific ways, employing an efficient initialization process which can improve training convergency, choosing efficient training schemes that adapt well to smaller sample sizes, using data augmentation techniques, using pre-trained networks and making use of transfer learning techniques, etc. Just to mention a few [21]. The idea is that size requirements to successfully train a Deep ANN model depends on many hyperparameters and design/architecture choices, and that these can be tunned to smaller to medium size datasets.

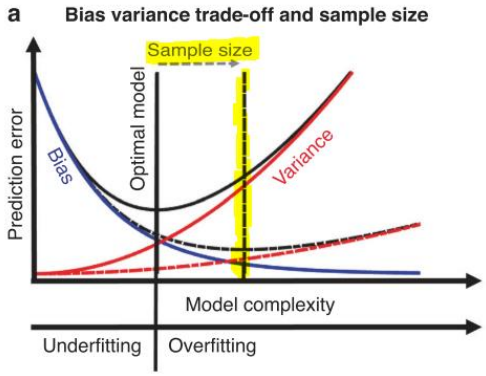



Figure 2.10: Bias variance trade-off and sample size shifting [21].

Another source of systematic overestimation can happen when evaluating model performance. When optimizing a model, it is recurrent to perform feature selection and hyperparameter tuning. If we perform these operations and evaluate the error in the same set, then this will optimistically bias the model. To choose the best parameters based on the same dataset in which we evaluate performance is to choose the parameters that best fit that specific set, which consists in a form of circular bias. The generalization error (performance of model) is ought to be reported in a set of data "unseen" by the model – the test set. For this reason, a Cross-Validation (CV) framework is the best way to conduct any of the former operations.

CV is a broad model validation strategy that allows estimating the generalizability of a model to an independent set. It is a resampling method, using different portions of the data to train and test, in several iterations. A common CV technique is a k-fold CV which divides data into k parts, using each one of them one time as a test set and all the others as the training set, therefore, it implies performing k iterations or runs. The Leave-one-out-CV (LOO-CV) implies living 1 sample for testing, using the rest of the data as training, thus requiring N-1 iterations, where N is the number of observations/examples in the dataset. Lastly, Leave-one-site-out-CV (LOSO-CV) is a good strategy to use for multisite data, in which a model is validated by leaving out in each iteration data from one site to test it, thus achieving a more robust and average performance of how a model whole generalize to an external set. Therefore, as a re-sampling procedure allows us to effectively evaluate the model, especially in case of limited data size. However, the CV disadvantages are associated with the increasing time complexity by increasing the number of evaluations performed, especially when high K folds are used.

In CV framework for hyperparameter tunning (thus model optimization), all hyperparameter combinations are applied and evaluated based solely on the training set, holding out a complete unseen set – the test set- to later evaluate the

generalization error. The recommended procedure is k-fold nested cross-validation. It consists of an outer loop where data is divided into training and test set folds. Then, an inner loop where the former training set is further divided into train and validation set folds. This is recommended when both hyperparameter tunning and estimation of generalizing error are performed. In the inner loop data transformations and hyperparameter tuning are performed. A certain model is trained, with hyperparameters combination A, and then tested on a validation set, within the inner loop. This nested loop is used to set the hyperparameters, whereas the outer loop, with a varying test set within each fold, is used to measure prediction performance. The performance measured in the inner loop tells us which model parameters fit best to our data, whereas the outer loop performance tells us an estimation of how the model would respond to unseen data, Figure 2.11 . The recommended number of folds, and most used K, is 10 folds, as it has been shown that it best balances bias-variance trade-off [29].

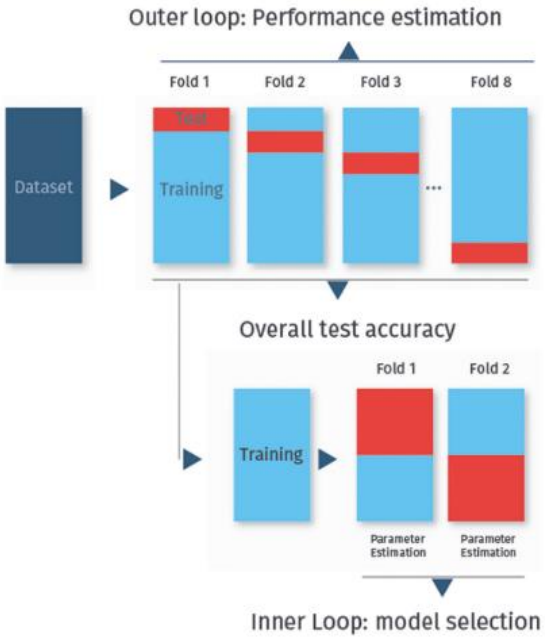



Figure 2.11: Nested K-Fold Cross-Validation [23].

In the best-case scenario, when there is a large enough sample available, it is recurrent to divide the data into train/validation/test, called Holdout method, where a single run is performed. With this method, the model is chosen by analyzing validation set performance and the generalization error is evaluated in the test set.

The next level, regarding model validation, is to use an external validation framework. For multisite data, this would mean using a new unseen independent site. For psychiatry disorders, it can also be useful to evaluate model specificity, i.e.,

test models on different psychiatry disorders since many disorders somehow overlap in some of the etiologic factors and symptoms. Performing carefully all the above-mentioned steps will lead to more robust conclusions regarding model generalization performance results, making the model one step closer to clinical applicability.

Another practice that can lead to the overestimation of results is called data leakage. Data leakage happens when statistical dependencies are created between train/validation/test sets. A common example would be to perform feature selection before train/validation/test set partition. If features are selected based on a global overview of the entire dataset, then, the test set is contributing to informing which features might be relevant, losing completely the validity of being a set of "unseen" data samples. Hence, later model evaluation on the test set would yield optimistic results because the model would have been shaped according, somehow, to test set feature distributions. Indeed, any data transformations and feature selection must be performed after dataset partition or within the cross-validation folds, exclusively on the training set and later applied to the test set [29].

Although the standard procedure would be model optimization, it has been reported, in an ML review of the kind, that 73% of studies employed only one ML model, and even for that model hyperparameter search was not performed [29]. Proceeding in this way might avoid data leakage and optimistic biased results, however, it is not very informative since any other model could outperform the latter, according to the *No Free Lunch Theorem*.

Still, many limitations are commonly found in Machine Learning studies for brain disorders. A lot of studies are performed on one site, lacking the external validity step for an independent site sample. This leads to relevant site-dependent conclusions but questionable generalization capability. Concluding, in order to have a Machine Learning Model which might yield clinical applicability, it is recommended to consider several check-points[29]:

1) Generalization:
    1.1) Nested- CV for model evaluation and hyperparameter tunning;
    1.2) Large independent test set;
    1.3) Large external test set;
2) Model-scope:

    2.1) Representative sample of target population: heterogeneous samples are recommended if it represents the real-life scenario for the population of interest. (e.g: excluding patients with certain comorbidities might be

suboptimal if the aim is to have a clinical decision support system for this group of patients) ;

3) Incremental utility:

   3.1) Reporting whether results outperform the current state of the art;

4) Model Interpretability:

   4.1) Excluding/Treating confounding variables;

# 3. Data Processing

Neuroimaging data can consist of raw images, processed images, or features extracted from MRI scans. The data processing protocols will be reviewed in this section with a particular perspective of how these protocols can then be included in a ML pipeline.

## 3.1 Brain Morphological Feature Extraction

Once the raw MR images are obtained, they can be processed to extract relevant measurements. One could always use raw image data and input it to a Deep Learning Model, however, such high dimensionality data would yield the necessity of extremely high computational resources, such to store data, optimizing the model and training the model. The alternative method is to proceed with some pre-processing to reduce the dimensionality of this data, such as with Regions-Of-Interest (ROIs) feature extraction. To extract ROIs features a Statistical Parametric Mapping (SPM) Tool [30] is used, which is also the standard procedure to perform a Voxel-based Morphometry (VBM).

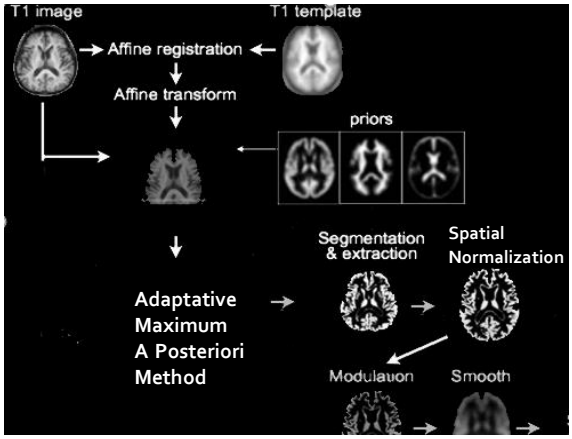
VBM is the study of local brain regions' sizes and shapes compared at a voxel-wise level for a given population. To do this, brain images need to be processed and tissues segmented, as the raw ones present noise, intensity-inhomogeneities, vessels, etc., all details that are not reliable or interested to study. Besides, to perform a group analysis or comparison one wants to eliminate all variations that are external and unrelated to group differences. The idea is to extract volumetric and cortical thickness measures from ROIs, as well as the Total Intracranial Volume (TIV), from each brain scan. Due to anatomy inter-subject variability, brain image scans contain unwanted spatial variations, but scans need to match spatially between each other so that the location of each region in one scan can correspond to the same location in another scan so that a direct comparison can be performed. Therefore, scans must be aligned in the same space, and shot to the same brain template, so that the remaining

differences account for group differences and nothing else. The spatial matching is called spatial normalization.

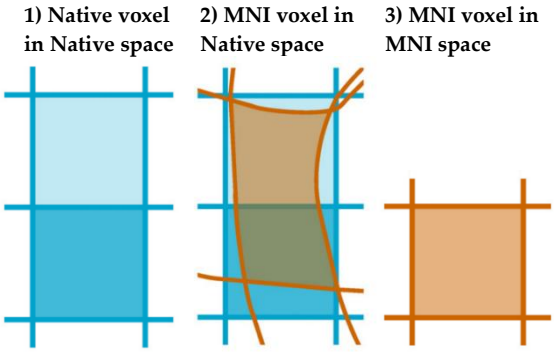
The protocol is usually implemented with SPM12 Software [30] or Computational Anatomy Toolbox (CAT12), an SPM12 add-on [31]. The default protocol follows the steps of Tissue Segmentation, into Grey Matter (GM), White matter (WM) and Cerebrospinal fluid (CSF), Spatial Normalization, making brain scans matching into a common space via registration to a standard stereotactic atlas( a default template usually given by the software) to ensure voxel-wise correspondence across different brains, an optional Modulation step which aims at correcting for local volumes deformations, and finally, a Spatial Smoothing step, which helps compensate some imprecision of the spatial normalization and increase Signal-to-Noise Ratio (SNR) [32]. When these steps are performed, ROIs features can be automatically extracted, through anatomical automatic labeling [33], using probabilistic atlases to extract volumetric and cortical thickness measures, since the brain scans will be matching to a reference template brain.

The main differences between the CAT12 protocol and SPM12 are in the method used to classify brain tissues. SPM12 uses Tissues Probability Maps (TPM), i.e., using images of six tissue priors representing their best guess on that tissue type to classify a voxel, which consists of a hypothesis-based approach. In this method, the image histogram is modeled as a mixture of 3 Gaussians, representing GM, WM, and CSF tissues, and the classification of each voxel tissue is done by estimating the contribution of each gaussian to that single voxel. CAT12 uses an adaptive maximum a posteriori (AMAP) method to classify voxel units, making the final tissue classification independent from priors, a hypothesis-free approach [32], [34].

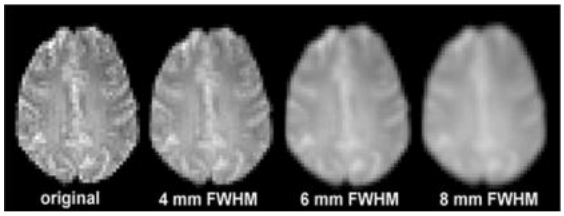
CAT12 basic voxel-based protocol called “Voxel-based processing” includes a module for tissue segmentation, spatial normalization, and ROIs generation and ROIs measurements extraction. The protocol can be separated into three main data processing steps, the initial voxel-based processing, then refined voxel-based processing, and finally the surface-based processing (optional). The initial voxel-based processing corresponds to the initial SPM12 pipeline, and includes an affine registration step and initialization of tissue segmentation with TPM priors, through the SPM Unified Segmentation[35]. The initial spatial normalization ensures the registration of individual MR scans to an MNI space template. The step includes a 12-parameter affine registration, it determines the optimum transformations among the following 12 degrees of freedom: 3 translations, 3 rotations, 3 shears, and 3 zooms, therefore linear transformations. The normalization, or warping, is performed to the default ICBM space template – European Brains.




a) CAT12 flowchart.




b) Modulation [36].




c) Spatial Smoothing. Adapted from [37].

Figure 3.1: CAT12 pre-processing.

The refined consequent step starts the brain parcellation and uses the AMAP approach to derivate the final segmented tissues and, as the last step, tissue segments are further spatially normalized, this time including non-linear deformations, to the template from (http://www.brain-development.org) of the IXI-dataset, using Diffeomorphic Anatomical Registration Through Exponentiated Lie algebra (DARTEL) algorithm [38].

After spatial normalization, an optional step called modulation can be performed. Modulation serves to correct for volumes changes, and its logic follows the reasoning that because brain regions can be artificially enlarged or shrink during normalization so to match the MNI space, the value of its voxels should be proportionally reduced or enlarged to guarantee that the original overall volume of that region is preserved in the normalized scan [36]. To correct for these volume alterations, the Jacobian determinants are used, which capture the amount of expansion or contraction of a voxel. If a Jacobian Determinant is 0.5, it means the native voxels were shrunk by this amount, and we can calculate 1/0.5 = 2, giving 1 voxel in the MNI space corresponding to 2 voxels in the native space. Therefore, if one multiplies the voxel

gray matter values in the MNI space by quantity 2, it would restore the original grey volume value, Figure 3.1b).

Regardless of performing the modulation step, the normalized or modulated images are spatially smoothed in the final step of the pipeline. At this last processing point, the voxels are convoluted with a gaussian function, with an optional Full-Width Half Maximum (FWHM), depending on the level of blurring one wants to achieve. This step aims at suppressing noise by ensuring each voxel contains an averaging of the voxel's values in its neighborhood, defined exactly by the FWHM choice. It guarantees also that data are more normally distributed, an important factor when considering final analysis through statistical testing. Lastly, it also helps compensate for imprecision in spatial normalization[32]. For balanced designs an FWHM of 4 mm was found to be enough to attenuate the nonnormality in data, while non-balanced designs were less robust to violations of normality, needing a smoothing with FWHM above 8mm[39].

To proceed with an automated ROI analysis, probabilistic atlases are given within the CAT software which can be used to estimate the volumes and thickness of different regions. The probabilistic atlases are created based on the labeling of MRI scans of multiple healthy individuals which are then registered to MNI space. The result is an average template that accounts for inter-subject variations, resulting in a labeled brain atlas, with the brain parcellated into known regions, covering the whole cortex, subcortical structures, grey and white matter structures, etc., depending on the atlases[40]. Commonly used atlases are the Hammers(c) Copyright Imperial College of Science, Technology and Medicine 2007. All rights reserved., CoBra[41], Neuromorphometrics Inc. [42], and Desikan-Killiany Atlases[43]. CAT12 then enables the estimation of mean tissue volumes, cortical thickness, and gyrification index of surfaces, for different ROIs, according to the atlases chosen as reference.

## 3.2 Brain Morphological Feature Processing

After pre-processing MR raw images, one can extract volumetric measures and cortical thickness measures for several brain regions. This allows for 1) repeatability of studies, since the regions correspond to an automatic labeling procedure using reference probabilistic atlases and 2) for reducing data dimensionality while still being able to study features provided from the subject's MRIs. Nevertheless, a further processing step should be considered regarding covariates of no interest coded in the data that can act as confounder variables to our model.

Dealing with confounders is an important data processing step, which might influence positively the final ML model interpretability. A confounding variable has several definitions in the literature, depending on the context it is applied. In disease etiological studies, confounding variables have a blurring effect that interferes with casual inferences by hiding the true causal effect [44]. In this kind of setup, a confounding variable is defined as a variable that has an association with disease, it must be a risk factor that is unequally distributed between exposure groups (HC vs not HC).

In the context of ML analysis, a confounding variable has a different definition, defined as a variable that confounds a certain predictive model. When using neuroimaging data, the confounder variable it's not constricted to be a risk factor, but rather a variable that covaries with the target (disease), that affects neuroimaging data, but which the ML model shouldn't take into account as useful information. It can be defined as a variable that affects our data but its association with a target variable is not representative of the population of interest. Hence, data in this setup is said to be biased by the confounding variable with respect to the population of interest [45]. This concept is illustrated in Figure 3.2, where TIV covaries with gender, with males having larger brain volumes, and in the population of interest the decreased brain sizes are associated with an increase in y scores (target variable). However, gender has no association with y target variable, as it is evenly distributed. In a biased sample, there is a correlation between gender and y that is not representative of the population of interest, with males tending to have higher values of y than females, thus gender is acting as a confounder.

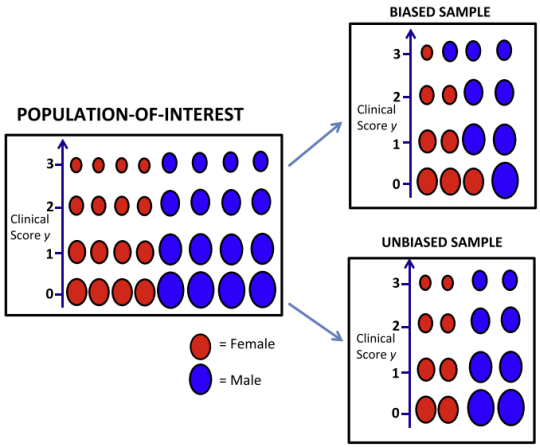



Figure 3.2: Confounders: A Biased Sample [47].

Besides, in an ML decoding analysis, confounders can be thought of as giving rise to a source ambiguity problem [46]. Usually, there is an interest in interpreting which features have contributed the most to an increase in the accuracy or performance of an ML model prediction. In this process, usually, it is (implicitly) assumed that the model is using information encoded in data that is uniquely related to the outcome of interest, the target variable. However, when the data encodes for information that is correlated with the outcome but that is of no interest to the study, there is a scenario of multiple unwanted sources of information which violates the aforementioned assumption, giving rise to a source ambiguity problem. This problem leads to the inability to know if the model used information that is uniquely associated with a disease or also information associated with a covariate of no interest, thus leading to confusion when trying to interpret a link between brain structure alterations and disease.

Acquiring brain imaging is costly, however, the hypothesis is that neuroimaging data will increase the ability of an ML model to identify a disease, and it's worth gaining knowledge about neuroimaging features that are relevant to identifying it. Therefore, it's important to prevent an ML prediction from being guided by other cheap-acquisitional clinical variables. Thus, in a setting which we care about interpretability, a variable that is encoded in our data and is not a covariate of interest, becomes a confounder variable for our model.

For example, gender can covary with several diseases (men or women being more often diagnosed with that disease than the other) and also affects grey matter, representing a covariate of no interest. In neurodegenerative diseases, however, age covaries with disease and also affects grey matter, but in this case, age effects are directly associated with changes in disease progression, being a covariate of interest. One might want to eliminate the age effect in neuroimaging data, without eliminating disease-specific age effects, thus removing age effects that are associated with Healthy Controls, leaving only the disease-specific age effects.

The data processing step which allows disentangling multiple sources of information that might be encoded in our data is called controlling for confounding variables. Besides gender, other variables can be considered confounding variables depending on the context of the study, such as age, medication, total intracranial volume (TIV), scanner or site effects. It has been shown that scanner and site effects can significantly bias the ML model and that it could accurately predict data origin based solely on the site effects that are encoded in data. This can lead to an ML model that learns the structure of the site effects on data and further uses it positively biasing its performance.[48], [49] To be sure the model is not capturing these variations in data, and that its performance is guided only by features of interest, it is necessary to

remove its effects from the data. Usually, confounding effects can be reduced by applying a statistical adjustment method. These methods produce an output that has been corrected for the effect of these variables, thus, being free of such confounding effects.

- **Regressing-out biological covariates**

One of the methods to model confounding effects is through regression models, which estimate the influence of one variable on another. The simplest regression model is a linear regression with 2 variables. The regression model can be extended to a multivariate case, reported in, Equation ( 3.1 ), assuming that each confounder has a linear relationship with feature Y and that their joint effect is the sum of their separate effects [45].

For continuous variables, the $\beta$ coefficient will give an indication of the effect of a unit increase in confounding variable X on the variable of interest Y. If we considered X to represent age, having a coefficient $\beta_1$, its meaning would be: 1-year increase in age leads to an average increase or decrease (depending on signal) of $\beta_1$ to Y. For X representing gender, which in the model is transformed into a dummy variable, the interpretation of $\beta$ coefficient is slightly different. Here, the number of dummy variables is always one less than the total number of options - eg: gender: female 1, thus leaving male 0, thus, $\beta_2$ * (1) female. The interpretation is, for a positive $\beta_2$ coefficient, that feature Y value is higher for females than for the reference group males. Supposing Y stands for cortical thickness (in mm) for region j, and $\beta_{j,female}$ is 5, then, females have, on average, a cortical thickness for feature j of 5 mm more than males.

$$Y = \beta_1 age + \beta_2 gender + \beta_3 TIV + \alpha \quad (3.1)$$

$$Y_{j,corr} = Y_j - \hat{\beta}_j X \quad (3.2)$$

$$where\ j\ stands\ for\ feature\ and\ X\ the\ confounding\ variables$$

It is important to notice that the proposed model is assumed, not proved, therefore the coefficients estimated do not yield any information about model rightness [50]. If the chosen model is not suitable, then the assumed association between covariates and outcome differs from the true association, which is called misspecification. There are methods to analyze the regression model validity, such as normality and independence of the regression residuals, the significance of the coefficients, analysis of variance, coefficient of determination $R^2$, multicollinearity of the independent variables, and confidence and prediction limits [51],[20].

1) Normality of Residuals: The residuals represent the errors between each pair of observations (x, y) to the linear regression. It can be seen as the remaining variance that hinds the linear regression to perfectly fit and predict the data observations. For this reason, there should be no source of regularity or pattern left in the residuals, i.e., the residuals should behave as random noise, satisfying the normality assumption. To assess the normality assumption of the residuals one can perform a goodness-of-fit hypothesis test, such as the Kolmogorov-Smirnov test, or plot a QQ plot of the residuals. Another method is to simply visualize the distribution of residuals against the fitted values of estimated y and look for abnormalities or regular trends which might indicate some explanatory factor left in the residuals not included in the model.
2) The significance of the coefficients: a hypothesis test is performed for each independent variable included in the model, for which the null hypothesis is that the beta coefficient contains the zero value in its confidence interval. In fact, if we conclude that a variable X, influences the dependent variable, positively and negatively, thus the confidence interval including the zero value, then this variable is also not given a meaningful contribution to the regression. Thus, we look to the p-value and conclude it is significant if $p<0.05$. For multiple linear regression, if a variable is said to not be meaningful to the regression, it does not mean it will never be meaningful, just that it is not meaningful for that group of independent explanatory features.

| predictor | value | standard error | $t$-value | Pr > $\|t\|$ |
|---|---|---|---|---|
| (intercept) | 1.36411 | 1.48944 | 0.916 | 0.3780 |
| volume | 0.29033 | 0.02423 | 11.980 | < 0.0001 |

Figure 3.3: Significance of coefficients Analysis [20].

3) Analysis of variance: Analyze that the predictive variables explain totally the variance inherent to the dependent variable Y, leaving aside only random noise represented by the residuals. If this is achieved, the sample variance of the residuals needs to be much smaller than the one of the independent variable Y.
4) Coefficient of linear correlation: the closer $R^2$ is to 1, the better the approximation of the distribution of the observations to the straight line.
5) Multicollinearity of independent variables: The independent variables included in the regression model should not be linear correlated. If they are, regression model significance is compromised.

After finding a linear equation that fits the data, we might ask whether it fits the data well. If it fits the data well, the coefficient of multiple determination $R^2$ should be close to 1, which tells us how much of the dependent variable variation is explained by the independent features. Then we can assess whether a particular independent variable contributes significantly to the regression, after controlling for the effect of the others, through the p-value or significance of the t-statistics.

Despite the previously considered problem, linear regression to adjust for covariates is a standard procedure in literature in dealing with confounding variables, called confound regression. After fitting the linear regression, the variance that can be explained by the confounding variable is removed from the data directly, with the estimated $\beta$ coefficients, reported in Equation ( 3.2 ). The $\beta$ coefficients are estimated by solving the Least Squares Problem, through minimization of the residuals or the Moore-Penrose pseudoinverse method.

One additional point we need to have into account, in an ML analysis, regarding the Cross-Validation (CV) framework. As it was discussed previously, whenever model optimization and generalization error estimation are performed employing the same subset of data, it necessarily leads to a biased model. Data transformations must take into account the latter, hence, correcting for confounders must be done within an ML pipeline in a way that does not break the CV validity. For this reason, the $\beta$ coefficients must be estimated only using the training set, and, if using k-fold CV, within each fold that is used for training. Then, the estimated $\beta$ coefficients are applied to the train and test set. This is called Cross-validated confound regression (CVCR) and was studied by L. Snoek et al. [46] yielding plausible model performances, while Whole-Dataset Confound Regression showed pessimistic and below chance level performance results.

Besides the regression model, another commonly used method is Counterbalancing (A Priori or Post Hoc). Apriori Counterbalancing is done by counterbalanced confounding variables in the experimental design. This entails that subjects are chosen (randomly) in a way that there is no correlation between confound and target variable, leading to the definition of rigorous excluding criteria. If this is not possible a priori, the Post Hoc Counterbalancing proposes to extract a subset of samples, from the original complete dataset, in which there is no correlation between target and confounders. This method however has been shown to yield optimistic model performances in an ML framework and has been suggested as an inappropriate method to control for confounders. This can be explained by the fact that restrictive excluding criteria or subsampling rejects samples that are harder to classify or to learn from, but that is representative of the population of interest, inducing substantial bias in the model [46].

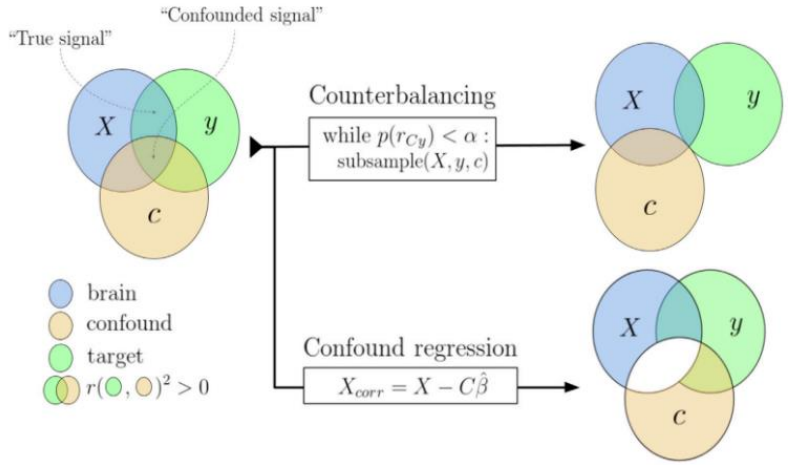



Figure 3.4: Dealing with Confounding Variables [46].

- **Data Harmonization**

More recently, with the increase of multicentric studies and datasets, a necessity for center harmonization has arisen. A tool specifically to remove batch effects in genomics was introduced in the literature and has been appropriated for the correction of site or scanner effects in neuroimaging studies. Site effects are non-biological covariates that stand for different acquisition protocols, different data acquisition sites, different scanning equipment, or even parameter configurations. It was then proposed a reproducible and repeatable method to model for these effects: using the same tool that was developed to deal with batch effects, called ComBat (Combating batch effect) [52].

ComBat is a parametric and non-parametric empirical Bayes framework for adjusting data for batch effects. In the context of its development batch effect was defined as systematic non-biological differences that make samples not directly comparable[52]. This concept can be easily generalized and applied to other settings besides genetics, such as the ones considering neuroimaging studies.

The model extends a linear regression in which biological covariates are included so that their effect is separated from the non-biological site effects, making the assumption that scanners or sites have both an additive and multiplicative effect on data. The ComBat model, described in Equation ( 3.3 ), shows a parameter $\alpha_v$ represents the average cortical thickness or matter volume for the reference site for feature $v$, $\gamma_{iv}$ are the coeficients associated with the site $i$ for feature $v$, $\beta_v$ are the coefficients associated with biological covariates X, $\varepsilon_{ijv}$ is the residual term with zero mean and $\delta_{iv}$ describe the multiplicative site effect for the ith site on feature $v$. The site parameters are estimated using Empirical Bayes, described in W. Johnson et al.

[52], and the final harmonize data is calculated like in Equation ( 3.4 ) with the star * signal representing the estimated site parameters. The estimated biological covariates effect, $\widehat{\beta_v}$ , are not removed, they are estimated but added again after correcting for the site effects, similarly for the data standardization procedure. ComBat has proven to be successful in harmonizing neuroimaging data, by removing site effects from data while conserving biological associations in data [53]. However, a further modification needs to be considered to be able to apply this processing step in an ML analysis, within a CV pipeline.

$$y_{ijv} = \alpha_v + X_{ij}^T \beta_v + Z_{ij}^T \theta_v + \delta_{iv} \varepsilon_{ijv} \quad (3.3)$$

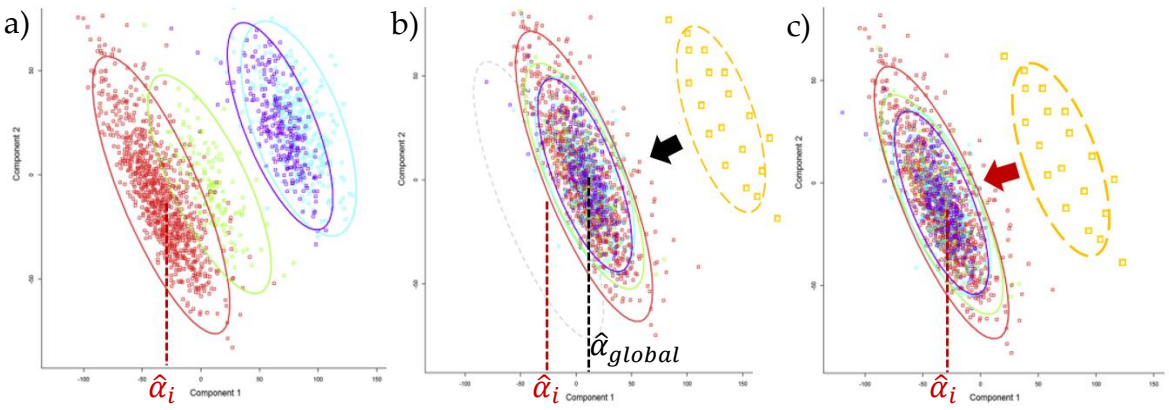
$$y_{ijv}^{ComBat} = \frac{y_{ijv} - \widehat{\alpha_v} - X_{ij}\widehat{\beta_v} - \gamma_{iv}^*}{\delta_{iv}^*} + \widehat{\alpha_v} + X_{ij}\widehat{\beta_v} \quad (3.4)$$

The site harmonization parameters need to be estimated in the training set once again, or within a CV fold, and then be applied to the test set. Several ComBat modifications have been proposed that allow that flexibility, such as in J. Radua et al. [54], where the authors propose a *combat_fit*, suitable to estimate and apply in the training set, and *comBat_apply*, to use only in test sets, made available in R in https://enigma.ini.usc.edu/wp-content/uploads/combat_for_ENIGMA_sMRI/combat_for_ENIGMA_sMRI.R. As well as in J. Fortin et al. [53], the authors that developed the neuroCombat function, provided in https://github.com/Jfortin1/ComBatHarmonization, also available in the form *neuroCombatFromTraining*, both in R and in Python, which enables the application of Combat coefficients separately to a test set. Nevertheless, both suffer from a drawback that limits their use in an external validation pipeline, the constraint that the sites or scanners from where data was acquired in the test set, coincide with those in the training set. This prevents the possibility of further testing an ML model with data pooled from a new, not seen, site. To respect the CV pipeline, the parameters cannot be re-estimated based on test data either, i.e, by re-running ComBat on training data plus a new site set because it would lead to a different harmonization from the one performed using only the training set. Further reasoning has to be made on how to apply ComBat harmonization for external datasets.

A different ComBat modification, proposed in C. Stein et al. [55], called M-ComBat, available in GitHub - SteinCK/M-ComBat in R, was proposed to center data on a location and scale of a pre-determined batch reference. The modification of M-ComBat enables the use of external datasets to validate fixed predictive models by using a reference batch to which new data is shifted. Although M-ComBat was developed for a different context, mainly to shift data to a gold-standard reference

batch, the idea of using a reference batch could be further applied to harmonized external data samples. Adapting to the training/test/external validation pipeline of ML, the independent external data set could be harmonized by adjusting the samples towards the already harmonized training set, which would work as the reference set. The untransformed data from an external validation site is brought to the level of the harmonized data.

In Figure 3.5, 4 different sites are represented – red, green, purple, and blue clusters stand for a training set that is being harmonized, and an external site represented in yellow, being harmonized a posteriori. Two possible application scenarios of reference ComBat are represented. The second figure, b), represents the standard ComBat application to a training set while using M-ComBat to harmonize a posteriori an external set. In c) both training and validation sets are harmonized using M-ComBat, in which, one of the sites included in the training set is chosen as the reference batch. The implementation of the reference-batch option has been further made available on *neuroCombat* and *pycombat* function [56], with the option *ref_batch*.




-Data samples from 4 different sites – red, green, purple, blue and an external validation site represented in yellow. a) Untransformed data. b) ComBat: using global grand mean and variance. Black arrow representing possible a posteriori harmonization for external data, shifting new data to grand-mean location, $\hat{\alpha}_{global}$. c) M- ComBat: using reference batch, i, mean, $\hat{\alpha}_i$ and variance. Red arrow representing possible a posteriori harmonization for external data, shifting data to reference-mean location $\hat{\alpha}_i$.

Figure 3.5: PCA visualization on different ComBat methods. Adapted from [55]

# 4. Aim of the Work

The methodological work presented in this thesis employs autoencoders in an anomaly detection framework to discriminate and classify BD subjects from HC. In order to achieve the main proposed goal, several intermediate methodological steps had to be investigated by the mean of secondary analyses, each with a specific secondary goal.

The ultimate goal is to achieve BD subjects' diagnosis through a normative automatic approach based on brain structural characteristics.

Thus, the first aim of this work is to build a healthy brain-features reconstructive model, free of biological and exogenous confounders, to be used as a normative model. The normative approach is trained on a dataset of HC and tested in an independent set composed of HC and BD subjects data for its capability of reconstructing HC samples and discriminating between BD patients and controls. This latter approach is then compared to a classical SVM classifier performance. Finally, a specific BD brain-feature pattern is assessed in order to uniquely classify the BD patient group.

In between this pipeline, mainly in the data processing step, several trials were performed to define a proper processing pipeline that would be robust towards confounding factors, generalizable to unseen datasets, following approaches usually recommended or employed in the scientific literature. Specifically, the comparison and optimization of processing methods for dealing and controlling with confounding variables (i.e., age, sex, and center's effects) were assessed as a secondary aim of this project.

Resuming, the specific aims of this thesis are:

1. Produce a successful normative model to reconstruct healthy brain features;
2. Discriminate BD against HC using the normative model;
3. Extract brain-feature abnormalities characterizing patients within the heterogeneous BD spectrum;

4. Assess if BD can be classified by using the subset of unique relevant brain features (aim 3) instead of all brain features;
5. Assess any improvement in BD classification obtained using the normative-based approach with respect to the classical SVM classifier;
6. Identify the optimal site-effect removal pipeline to be integrated in a ML analysis by comparing different multisite harmonization pipelines combined with biological covariates correction;

For the first point, the question to ask is, "Can the model successfully reconstruct HC brain features?", for which the answer is concluded by developing a normative model on a big training set composed of HC and by measuring, in a HC test set, the reconstruction error. The second goal aims to answer the question "Can BD be discriminated by employing a Normative model approach?" which we try to answer by assessing the discriminative power of the model's BD reconstruction error, through a test set composed of BD patients. The third point would answer to the question "Which brain features deviate the most in BD patients with respect to HC ?". Then, the fourth goal is reached by answering the question "Can we use this subset of abnormal brain features to uniquely identify BD subjects with respect to HC?", and we answer it by evaluating the discriminative performance of this subset of features on an external independent set of data (replication set). Specific goal n. 5 is reached by comparing different multisite harmonization pipelines integrated differently in the ML analysis framework. They are compared based on the normative model results (i.e., reconstruction error and BD anomaly detection metrics), and we try to understand which harmonization pipeline is the most effective in removing site-related confounders while preserving the biological variability of interest and independence between train, test and external sets. Finally, the performances of the autoencoder-based BD classifications are compared to the performance of the SVM classifier, acting as a baseline, to assess the improvements in the automatic BD vs. HC classification produced by the newly proposed anomaly detection approach (specific aim 5).

## 4.1 Organization of the thesis

To reach the above-defined goals, the thesis results and methods will be organized coherently and chronologically to the designed pipeline. This thesis is organized in the following way: first, the MRI data preprocessing step, which includes the usage of CAT12 software to extract brain morphological features, second, the cross-validation framework and splitting of the dataset are defined so that all further data transformations and processing are performed within that framework; third, the processing step, in which parallel sub-pipelines are defined, involving the multisite

harmonization procedure with ComBat function and the biological confounder regression procedure, which later will be compared among each other based on the final results of classification; forth, the design and evaluation of the gold standard SVM classifier; fifth, the design of the normative model and consequent hyperparameter tunning, then the evaluation of the model performance on both unseen HC and BD subjects to determine its discriminative power, and finally the selection of a subset of brain features to uniquely identify BD patients and consequent evaluation performance with an external independent set. The model evaluation is repeated for all harmonization pipelines defined in the third step – the data processing step- and all model results will be compared.

## 4.2 Methodological approach

In this section, it will be explained the foundations that led to the definition of the proposed pipelines and methodologies, since those are not standard approaches to the problem this work is trying to solve.

- **Data processing stage**

In the work presented in this thesis, the methodological approach to correct for biological covariates and to model differences in MRI acquisition protocols and scanners will consist of the use of linear regression to adjust for biological covariates and of the ComBat tool for data harmonization across multiple sites. What led to the definition and comparison of several pipelines regarding data harmonization was the lack of a standardized protocol. Even in the literature, different strategies are used for this purpose, or entirely skip this step of data processing. Besides, including this data processing step within an ML analysis pipeline is rather recent, and not many references are found in the literature, especially for neuroimaging data.

- **Autoencoder Pipeline**

The methodological approach proposed for the normative model was based on the pipeline presented in W. Pinaya et al. [57]. In the former study, the authors designed a deep AE model to detect abnormalities in subjects' brain structures. With this approach, the authors reported an attempt to discriminate between HC subjects and patients affected by Autism (AD) and Schizophrenia (SCZ) both with a classical SVM classifier and with the normative AE-based approach, and extract brain regions that are found to be significantly different between healthy and patients groups. The authors used 104 brain features, composed of 68 cortical thickness measures and 36 neuroanatomical volumetric measures, extracted from the Desikan-Killiany atlas and via whole-brain segmentation procedure, respectively. The AE model was trained with 1113 HC subjects from the public dataset Human Connectome Project, and the

test sets were drawn from NUDAST (40 HC and 35 SCZ) and ABIDE (105 HC and 83 AD) public datasets. The reported results show that the AE model is capable of differentiating HC from SCZ patients with an AUC-ROC=0.707 and HC from AD patients with an AUC-ROC=0.639, whereas the SVM classifier achieved an AUC-ROC=0.637 and AUC-ROC=0.569, respectively. Furthermore, brain regions were identified for each disorder, those which presented significantly deviations.

The described AE model is a versatile one since it is not constricted from the beginning to classify one disease type, but rather can be applied to any and all brain disorders. Furthermore, the reconstruction error is a useful interpretability tool that characterizes this methodology, as one can trace back, easily, the features that have been relevant for the lack of success when reconstructing data from a specific subject.

In conclusion, the overall objective of the former study was to develop a model capable of discriminating HC and non-HC subjects and then use the model to investigate abnormal brain regions in different brain disorders.

In the present thesis, for the first time, we aim to develop an adapted version of the brain structural normative model and to use such a model for the detection of neuroanatomical underpinnings of bipolar disorder. Therefore, the proposed pipeline is to develop an HC discriminative network, an adapted version of the normative approach presented in the previously mentioned study, and then develop a subsequent step to classify HC and BD patients.

An AE model is trained solely on HCs' brain morphological features extracted from sMRI data, with the main goal being that the AE learns the hidden data structure of healthy brains. The hypothesis is that an AE model should have more difficulty in reconstructing data from a subject not belonging to the HC group, and this will be quantified by a reconstruction error (RE) metric, such as the deviation between feature value reconstruction and original, denoted from here and on as Deviation Metric (DM). The DM score is then used to threshold HC from non-HC individuals.

Then, abnormal brain regions in BD vs HC are investigated, by extracting the brain features that are being worst reconstructed in BD patients. A Mann-Whitney U-test is performed to the feature deviation scores comparing the two groups, and the brain regional features that yield statistically significant differences at the DM level between HC and BD are extracted. Theoretically, to specifically classify an individual as a BD patient there must exist a unique disease feature-pattern signature, i.e., the specific disorder needs to be identified through a unique set of brain regions that specifically deviate overall from HC but do not overlap with any other brain disorder. The hypothesis is that, if the abnormal brain regions that are found to belong to the BD patient group are generalizable when taking into account the

network reconstruction of those features in a new independent test set, the classification of the two groups should be well above chance.

The proposed methodological pipeline is reported in Figure 4.1. The first step is to evaluate the model capability in discriminating HC and BD subjects, and the second step is to explore the local brain abnormalities by extracting relevant brain features, i.e., the brain regional morphological characteristics that mostly deviate in BD patient group from the normative model. The additional step, in an attempt to classify the specific patient group, is to use only the abnormal brain regions to perform classification, hypothesizing that the regions that are found are generalizable to any BD subject. This last step must be validated in a new independent test set.

There are important distinctions between the methodological approach described in W. Pinaya et al. [57] and the one presented in this work mainly regarding the modeling for different MRI acquisition protocols, which was required in our study. In the described study the authors did not need to model these differences, since the network was trained with a dataset collected from the same site, the problem of the network learning site effects was not posed. Besides, the authors designed a model in a semi-supervised manner, with age and sex included in the data which the model learns to classify in a supervised framework while using a custom loss function to disentangle the information learned about age and sex from the latent variables encoded by the AE.

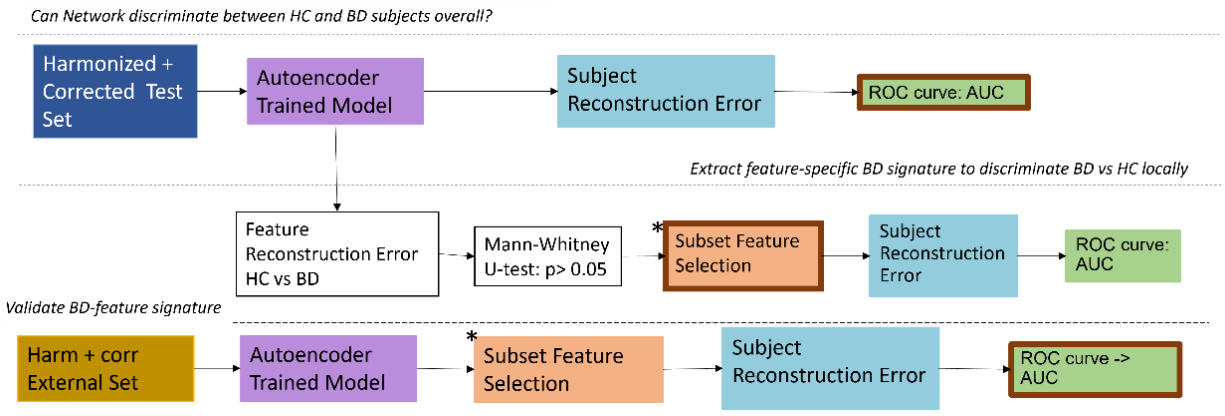



Figure 4.1: Proposed AE Pipeline to classify HC and BD.

# 5. Methods

## 5.1 Project

This thesis was developed at B3Lab (Biosignal-Bioimaging-Bioinformatics) in the Department of Electronics, Informatics and Bioengineering at Politecnico di Milano in collaboration with the MiBrain (Milan Brain Research on Affective and Integrative Neuroscience) Lab coordinated by Prof. Paolo Brambilla in the Psychiatry Unit of the Fondazione IRCCS Ca' Granda Ospedale Maggiore Policlinico and the University of Milan. The methodological work is inserted n the scope of the Moodlearning project, "Classifying unipolar versus bipolar depression: an innovative diagnostic support system based on clinical features and genetic, inflammatory and neuroimaging biomarkers". The Moodlearning project has been granted by the Italian Ministry of Health (GR-2018-12367789) to the Ospedale San Raffaele S.R.L., Milan, (PI: Dott.ssa I. Bollettini) and the Fondazione Policlnico (Operative Unit leader: Prof. E. Maggioni).

The main goal of the Moodlearning project is the development of a clinical decision support system (CDSS) to aid psychiatrists in the diagnosis of Bipolar Disorder, specifically in discriminating between Unipolar and Bipolar Depression.

The Fondazione Policlinico Unit, which is responsible for developing and releasing the CDSS, is implementing data pre-processing and processing pipelines using already available datasets, which were collected in the context of multicentric projects. Specifically, to reach the before mentioned purpose, clinical and neuroimaging data was collected from BD and HC across several hospitals and research centers that participate in the StratiBip network initiative, which is coordinated by Prof. Brambilla and Prof. Maggioni . With this multisource and big sample data, the goal of StratiBip is to:

1) Identify reliable biomarkers to discriminate BD patients from HC and stratify BD based on neurobiological dimensions.
2) Create an international consortium for sharing BD clinical and neuroimaging data.

In the present thesis project, the StratiBip sMRI dataset was used to develop the AE-based normative model and test its performance in discriminating individuals with BD from HC.

In the future, the newly developed tool will be tested on the newly acquired MoodLearning dataset and extended to accommodate multiple patient groups. Specifically, using the identified deviation features as candidate BD biomarkers, the ultimate intention is to integrate this data and develop an ML model to discriminate healthy controls or other disorders (e.g., major depressive disorder (MDD)) against BD patients.

The novelty that is brought by the StratiBip Project is the integration of large multisite and multimodal data, with the aim to achieve higher generalizability of prediction models and higher statistical power. Besides, the numerosity of subjects that can be included to train and test the ML model leads to more robust conclusions and results - which might be relevant for regulatory requirements and CE marking for the CDSS software [58].

The work in this thesis was focused on discriminating between HC and BD subjects. All the Data Processing and Machine Learning Pipelines were built using Google Colaboratory, a free cloud service by Google, and Anaconda3 with Jupyter notebook, with Python 3.7.13.

## 5.2 Participants

The data used in this work was retrieved from 7 centers that have been collected from the Stratibip study, previously introduced. It is composed of 1163 subjects, divided into 605 HC and 558 BD.

All subjects underwent a clinical assessment by professional psychiatrists, which confirmed BD diagnosis according to the Structured Clinical Interview for DMS-IV Axis I disorders (SCID-I)[59], with the exception of the patients recruited in Vancouver Center for which the BD diagnosis was achieved based on a clinical interview and Mini International Neuropsychiatry Interview[60].

The excluding criteria considered in the study were: comorbidities; mental retardation; pregnancy, history of epilepsy, major medical and neurological disorders; neuroleptic treatment in the last 3 months; drug or alcohol abuse in the last 6 months; medical conditions affecting immune system.

All subjects provided written informed consent to the study protocol, which was conducted in accordance with the Declaration of Helsinki and approved by the Ethical Committees of the participating centers.

After recruitment, sociodemographic and clinical data for all subjects was attentively checked for missing data, which resulted in the exclusion of one subject due to missing biological and clinical data (eg: age, sex, diagnosis).

| ID | Center | Abbreviation | Reference Person | HC | BD | Total |
|---|---|---|---|---|---|---|
| **1** | **AOU Verona, Verona, Italy** | AUOV | Marcella Bellani | 93 | 20 | 113 |
| **2** | **Fondazione IRCCS Santa Lucia, Roma, Italy** | FSL_ROME | Gianfranco Spalletta | 250 | 257 | 507 |
| **3** | **University of Jena, Germany** | JUH | Igor Nenadic | 111 | 23 | 134 |
| **4** | **Milano Policlinico, Italy** | MI_POLI_3T_3 | Paolo Brambilla | 26 | 12 | 38 |
| **5** | **Ospedale San Raffaele, Milano, Italy** | OSR | Francesco Benedetti | 67 | 133 | 200 |
| **6** | **University of Pittsburgh, Pittsburgh, US** | PITTS | Amelia Versace | 28 | 58 | 86 |
| **7** | **University of British Columbia, Vancouver, Canada** | UBC | Lakshmi Yatham | 30 | 55 | 85 |
| **Total** | **-** | - | - | 605 | 558 | 1163 |

Table 5.1: Center Participants Information.

## 5.3 Data Acquisition

The sMRI scans were acquired in the 7 centers using T1-weighted sequences on 3T RMN scanners.

- **1-AUOV-Verona**

-*Scanner:* Magnetom Allegra Syngo (Siemens, Erlangen, Germany)

-*Sequence:* T1-MPRAGE

-*Matrix size:* 256x256x160 mm3

-*Voxel Size:* 1.00x1.00.x1.00 mm3

- **2- FSL-Rome**

-*Scanner:* Philips Achieva 3T (Philips, Best, the Netherlands)

-*Sequence:* T13D-MPRAGE

-*Matrix size:* 432x432x190 mm3

-*Voxel Size:* 0.542x 0.524.x 0.900 mm3

- **3-JUH-Jena**

-*Scanner:* Siemens Tim Trio (Siemens, Erlangen, Germany)

-*Sequence:* T1 Magnetization Prepared Rapid Gradient Echo (MP-RAGE)

-*Matrix size:* 256x256x192 mm3

-*Voxel Size:* 1.00x1.00.x1.00 mm3

- **4-MI-Milano Policlinico**

-*Scanner:* Philips Achieva 3T (Philips, Best, the Netherlands)

-*Sequence:* T1-Turbo Field Echo (TFE) 3D

-*Matrix size:* 240x240x165 mm3

-*Voxel Size:* 1.1x1.05x1.05 mm3

- **5-OSR-Ospedale S.Raffaele**

-*Scanner:* Philips Intera (Philips, Best, the Netherlands)

-*Sequence:* T1-Fast Field Echo (FFE)

-*Matrix size:* 256x256x220 mm3

-*Voxel Size:* 0.9x0.9x0.8 mm3

- **6- PITTS-Pittsburgh**

-*Scanner:* 3T Siemens Tim Trio

-*Sequence:* -

-*Matrix size:* 192x256x192 mm3

-*Voxel Size:* 1.00x1.00x1.00 mm3

- **7-UBC- Vancouver**

-*Scanner:* Philips Achieva (Philips, Best, the Netherlands)

-*Sequence:* 3D TFE

-*Matrix size:* 256x256x180 mm3

-*Voxel Size:* 1.00x1.00x1.00 mm3

## 5.4 MRI Preprocessing

The raw MRI scans were processed after acquisition using a gold-standard protocol.

Firstly, images underwent a visual check to assess their quality and were converted from DICOM to NIFTI, the standard format for neuroimaging analysis software. The preprocessing pipeline was performed in Matlab R2018a (The Mathworks, Inc®) environment.

The preprocessing was performed using SPM version 12 Software [30] (http://www.fil.ion.ucl.ac.uk/spm/software/spm12/) and Computational Anatomy Toolbox (CAT12), an SPM12 add-on [31], using the preprocessing module of the "basic VBM analysis" described in chapter 3.1 Brain Morphological Feature Extraction. For all the preprocessing descriptions that will follow, the used parameters were the default ones for the *basic VBM analysis* [61], unless stated otherwise, for which in that case parameters are specified.

The preprocessing pipeline followed the hereunder steps:

1. **First Module: Data Segmentation**
    a. *Writing Options*: Process Volume ROI – Atlases – cobra;
2. **Second Module: Display Slices** – Quality check after segmentation;
3. **Third Module: Estimation of Total Intracranial Volume (TIV)**
4. **Fourth Module: Sample Intensity Homogeneity**
    a. No Nuisances
5. **Fifth Module: Data Smoothing**
    a. *FWHM*: 6mm

The brain ROI measures included in this work regarded only Gray Matter (GM) and were extracted from 52 subcortical regions of CoBra Atlas for volumes and 68 cortical regions of the Desikan-Killiany Atlas for cortical thickness, whose detailed description is reported in Appendix A.

## 5.5 Cross-Validation Framework

An internal and external validation frameworks were designed to evaluate models' performance. Within the internal validation framework, a 10-fold CV was used during the hyperparameter tuning to estimate the best model, while a holdout method was used to estimate the generalization error. First, a holdout method was used, splitting the entire dataset into training set, test set, and an external replication set, which was used in the external validation framework. The training set was then inputted into a 10-fold CV framework for hyperparameter tuning. A simple holdout

method was chosen for generalization error evaluation due to the high computational resources and the amount of time needed to perform a complete 10-fold nested CV, given the DL model properties.

- **Holdout Method: Dataset Splitting**

The data was split into a training set, test set, and external site set in the following manner:

Firstly, from the 7 centers contained in the dataset, one was randomly holdout as an independent site set, specifically all data from the PITTS site. Secondly, data were divided into HC datasets and BD datasets. The HC dataset was split stratifying to site proportions, in 90% training data and 10% test data, using function *train_test_split* from class *model_selection* in the *scikit-learn* python library, with a fixed random-state to ensure the splits were always the same and reproducible. The BD dataset was split also stratifying to site proportions holding 15% for the test set, so to have a balance test set. The training set, composed only of HC, contained 519 HC subjects, whereas the test set was composed of 58 HC and 75 BD subjects. For specific analysis, which will be described in the next sections, we use the whole BD dataset, 500 subjects, instead of the BD test set. The described dataset split will be used for all of the subsequent steps in the general ML pipeline.

- **Hyperparameter Tuning: 10-fold CV**

To perform hyperparameter tunning to the AE model which will be described afterward, a 10-fold CV framework was used. Within this process, only the training set was used, retrieved from the splitting described in the previous point, and which is further divided into 10 folds iteratively used for training. The data transformations that are to be applied in the holdout pipeline (i.e, to the training, test, and external sets), will be applied similarly inside the hyperparameter search process, where 9 folds of the training set are used to train the model with a hyperparameter combination $x$ and 1 fold is used to test the trained model. This process ensures that the best hyperparameter combination is chosen according to its performance on the training set part, without compromising the unseen test set. In this way, once the best model was realized within the 10-fold CV, the whole training set was then used to re-train the best model from the beginning.

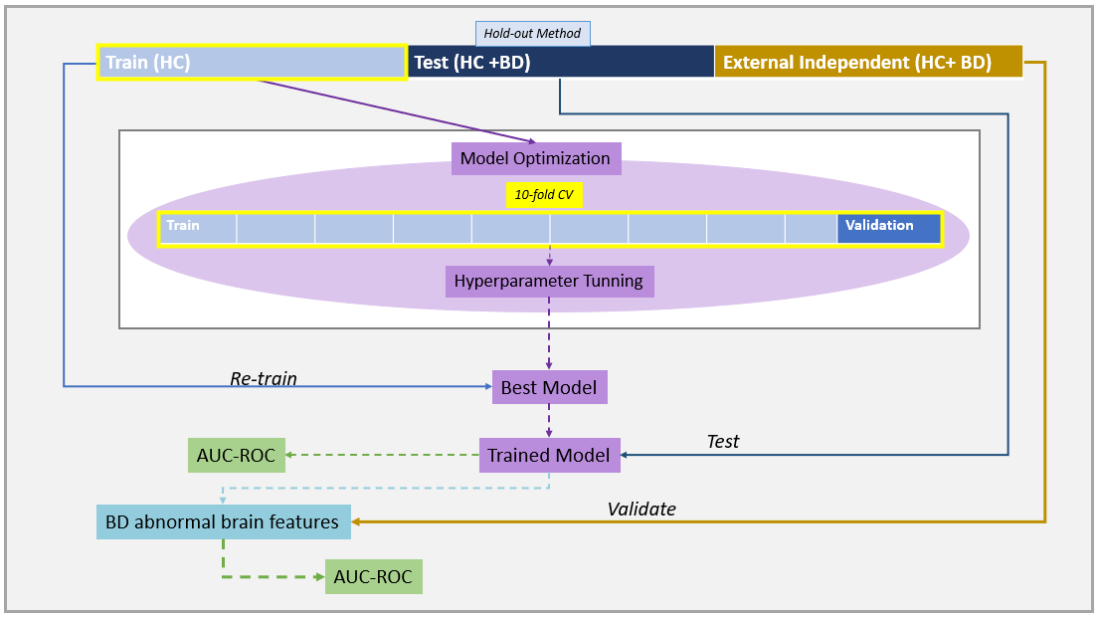



Figure 5.1: Internal and External Validation Framework Description.

## 5.6 Modeling Confounding Variables

In this step, multisite data harmonization and correction for biological covariates are considered. There is not a gold-standard approach to dealing with confounding variables, and so, the effort was to align ourselves to what was more conventional in the scientific literature.

The multisite data harmonization was achieved by performing ComBat with *neurocomBat* function in python, and the correction for biological covariates by a regressing-out approach, through a linear regression fitting, both done considering all brain ROI features included in the data. Several variations of the former methods were investigated.

The assumptions that were considered for applying the former transformations were that age-related changes and inbetween sex differences might be comparable between HC and BD. This assumption allows us to perform estimates on the training set, which is composed of only HC data. However, if there is a correlation between older age and more serious brain damage- due to chronic exposure to disease –, those effects would still be present in the disease group.

In light of these considerations, due to the lack of standard protocols in the data harmonization processing step, it was decided that the best approach would be to create distinct pipelines and compare them. For this reason, different harmonization approaches were considered which were the following:

### A. No Harmonization

Data are not harmonized across centers, which are concatenated as if they were a unique center.

### B. CV-Harmonization

Site-related effects are estimated and corrected by respecting the CV frameworks, both in the holdout method and 10-fold CV during hyperparameter tunning. The center effects are only estimated on the data which is being used as a training set and those estimates are applied afterward to the test set. The test set contains data from centers that belong also to the training set.

### C. Harmonization of an External Set

The external set is a test set that contains data from an independent unseen center, which is not included in the training set. The harmonization process is done a posteriori, as if after the model design in a real clinical application. The ComBat correction coefficients related to the external set are extracted keeping the external set independent from all the others, by using the reference batch method, as explained in section 3.2, avoiding to re-run ComBat including external set examples into the training set which would be a form of data leakage and break the independence of the external set.

### D. Harmonization of the whole dataset

Data is harmonized before entering the ML pipeline, i.e., before dataset splitting.

The effectiveness of the above harmonization pipelines, which will be presented in the results section, is qualitatively assessed by inspecting the main directions of data variance before and after the application of each harmonization approach using PCA.

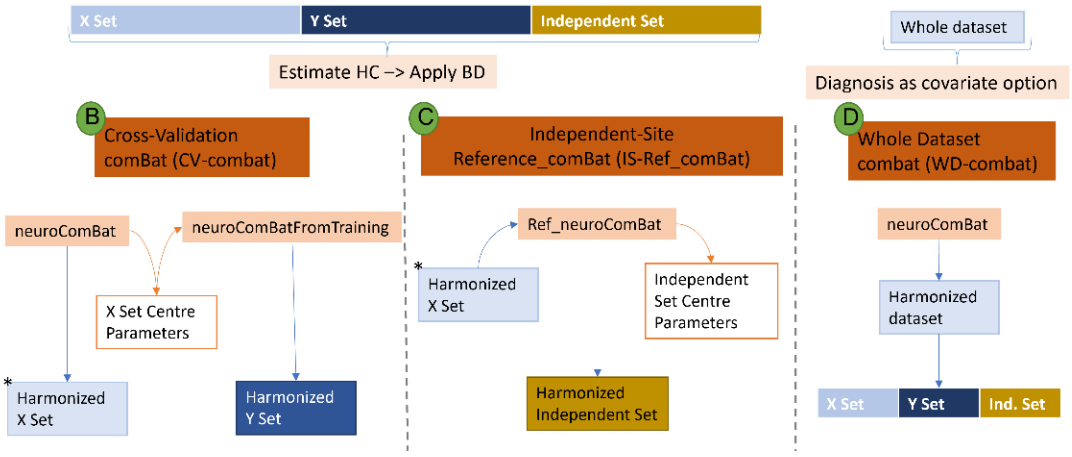



Figure 5.2: ComBat Options.

From the former data harmonization options, A, B and D are alternatives to each other, A serves as a baseline comparison for non-harmonization, B is a consequence of integrating this processing step into an internal validation ML framework (i.e., the holdout/CV framework), and D is the opposition to B, harmonizing data outside the internal validation framework. Whereas option C, the harmonization of an external set, is the variation that allows the integration of this processing step into an external validation framework. Hence, B and C options are combined to design a pipeline fully integrated into the ML validation frameworks, whereas D and C are also combined, but for the objective of comparing against option D alone.

Regarding the adjustment of data for biological covariates, a fixed pipeline was considered, and all variations of the processing pipelines perform this adjustment by respecting the CV framework (as in the above B option).

Thus, the processing pipelines arising from the combination of the different harmonization options that I presented before, together with the correction for biological covariates are the following:

**1) No Data Correction (A) Pipeline**

Including correction for biological covariates:

**2) No Harmonization (A) Pipeline**

**3) Whole Dataset Harmonization (D) Pipeline**

**4) Whole Dataset Harmonization (D) + External Set Harmonization (C) Pipeline**

**5)CV-Harmonization pipeline (B+ C)**

The following sub-sections will explain in a detailed manner the harmonization options integrated into pipelines 3,4,5, (options – B,C,D). A detailed description will be made of how pipeline 5 is integrated inside the hyperparameter tunning, where a 10-fold CV was used, where data processing was performed within each fold of the CV. Finally, a detailed description will be reported of how the correction for biological covariates takes place, which is similar to all processing pipelines including this step (2,3,4,5).

### 5.6.1 Data Harmonization Options Within Processing Pipelines

For all data harmonization options, the data is divided into cortical thickness measures and volumetric measures. The biological covariates considered to harmonize cortical thickness measures are age and sex whereas, for volumetric measures, TIV is also added as a biological covariate.

- **Pipeline 5: CV-Harmonization**

**-Holdout method: Option B+C**

***I. CV-ComBat – Option B***

After dataset splitting, all confounder effect estimations and data transformations must be performed in the training set and then applied to the test set and the external set to respect the CV framework. Thus, the *neurocomBat* function is used firstly on the training set data.

Cortical thickness and volumetric measures are harmonized separately, however, before harmonizing volumetric measures, for which TIV must be included as a biological covariate, TIV must be itself corrected for site effects. TIV is a measure extracted from the MRI scans in the CAT12 preprocessing so it is affected by the same site effects. This preliminary correction step avoids that site effects are preserved in data by preserving the TIV effects that include site confounding effects. To this end, the volumetric regional features extracted from the CoBra atlas and TIV values are concatenated, and the *neurocomBat* function is applied to that data, using age and sex as biological covariates. The harmonized TIV values are extracted and the *neurocomBat* function is applied once again to original ROI volumetric measures, using age, sex, and harmonized TIV as biological covariates.

After this process, comBat center parameters ar extracted to harmonize the test set. The harmonization of the test set is done using the *neurocomBatFromTraining* function, which receives as input all the ComBat center parameters estimated in the training set. The application of the estimates is done in the same fashion as in the training set, i.e., dividing test data into cortical and volumetric measurements, harmonizing separately cortical thickness data, TIV, and volumetric measurements.

***II. Ref_ComBat – Option C***

Finally, there is the external set, which belongs to a center that is not included in the training and test sets, which still has to be harmonized. This choice was aimed to assess the versatility and suitability of the newly developed tool in a real multicenter research project. To not break its independence, data from the external center must not be present in the training set, to simulate data from a new collection site that has arrived after model design. Therefore, there are no estimated parameters for this new center. The strategy to harmonize an independent center is to consider the harmonized training set as a unique reference batch, and the new center will be brought to its level (i.e., overall mean and variance of training set). To ensure consistency of the procedure, the independent center parameters are estimated in the previously described way, using only the HC from the independent center. Then, the center parameters are applied to the BD subjects using the *neurocomBatFromTraining* function.

**-10-fold CV: AE Hyperparameter search**

All experiments presented in the results section, considering the several processing pipelines, were performed using the same best model, which is found through the hyperparameter search using pipeline 5 as the data processing pipeline. Data processing was performed within each fold respecting the CV framework. Within the 10-fold CV, the data was harmonized and corrected using *option B – CV- ComBat*, respecting the dataset splitting within each fold. Here, only the training set is being used, and the estimations are performed to the 9-folds used to train the model, and the estimations are applied to the 1-fold used as the validation set. The external set data will not be seen in this process (nor the test set from the dataset split with the holdout method), thus model hyperparameters will be fitted by training with data that does not include any observation from the external set center acquisition. The best model will be retrieved and used in all processing pipelines for possible comparability of results.

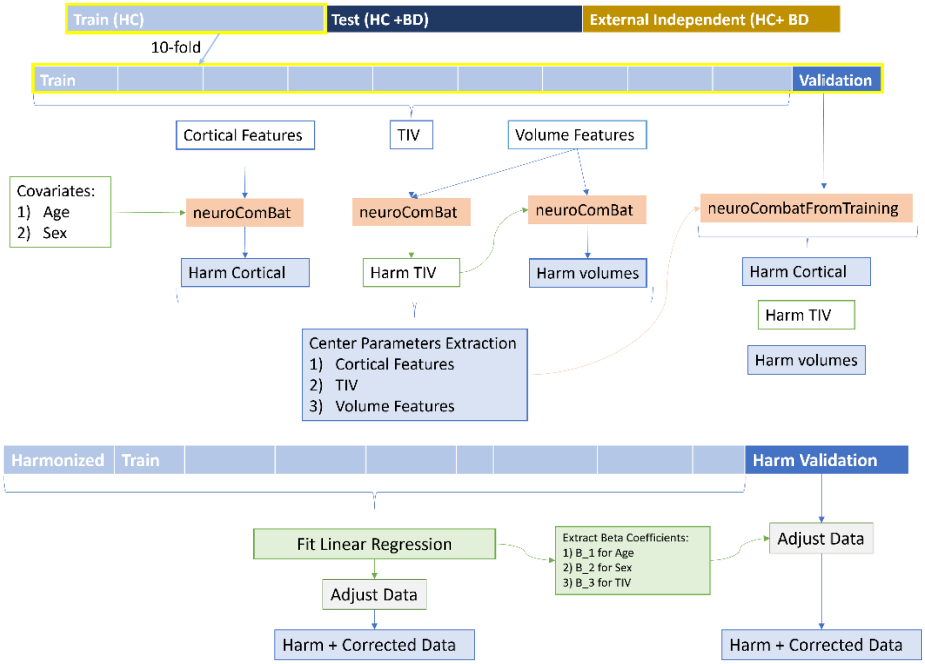



Figure 5.3: Data processing within hyperparameter tunning.

- **Pipeline 3: Whole-Dataset Harmonization ( D )**

Before splitting data into the training set, test set, and external set, multisite data was harmonized all together using *neurocomBat* function, considering the diagnosis variable as a biological covariate. The variable is added to the demographic biological covariates (age, sex and, TIV), to ensure that *ComBat* does not eliminate some biological effects by mistaking them with site effects. After data harmonization,

the resulting harmonized dataset is entered into the validation frameworks. This option is considered to try to model the scenario in which data ideally would come from one unique center, and so the data processing stage would start with data that is somehow ideal. The correction of data for biological covariates is performed within the CV framework, so after the harmonization process takes place and after dataset splitting.

- **Pipeline 4: WD Harmonization (D) + External Set Harmonization (C)**

In this harmonization pipeline, the PITTS external set is holdout before the whole-dataset harmonization. This pipeline is set to be compared especially with the previous one, pipeline 3, in an attempt to understand how model generalization to an external set is influenced by harmonization. Also, compared with pipeline 5, as ComBat harmonization is influenced by the number of examples it is presented with, increasing its performance with the increase of dataset examples, it could be the case that the harmonization process is improved slightly when performed in more numerous data, case of this pipeline 4, and that would lead to an improved model generalization to an external set. Thus, WD-ComBat is performed with *neurocomBat* function and *ref_neurocomBat* is used to harmonize a posteriori the external set. The correction of data for biological covariates is performed within the CV framework, after dataset splitting.

### 5.6.2 Biological covariate correction

Data is adjusted for biological covariates in pipeline 2,3,4,5 in a similar way, although in pipeline 2, No harmonization (A), the input for this step is the raw data, instead of harmonized data. The covariates taken into consideration are age, sex, and harmonized TIV (or raw TIV for pipeline 2). From this point forward the training set, test set, and external set are already harmonized (for pipelines 3,4,5).

Firstly, the training set standardized statistics are estimated, employing the *StandardScaler()* python function from the *preprocessing* class in the *scikit-learn* library, applying the function *fit()*. Then, the training set, the test set, and the external set are standardized by applying the python function *transform()* using the former estimated training set standardization statistics. The effect of the biological covariates is estimated through a linear regression considering each brain ROI feature as the dependent variable. Cortical thickness features are corrected for age and sex effects. The linear regression fit is done using the *OLS()* function from *statsmodels* python module, besides the function *add_constant()* is used to add an intercept to our linear regression. The *OLS()* function receives as input the training data, and the biological covariates matrix, which contains a constant unit column in the case of the intercept

option. Then, employing the *fit()* function, linear regression is fitted to the training data. Afterward, only the volumetric measures are considered, which are corrected for age, sex and (not harmonized for pipeline 2)/ harmonized TIV. The same process as before takes place, the linear regression is fitted to the harmonized volumes of the training set and the beta coefficients are estimated.

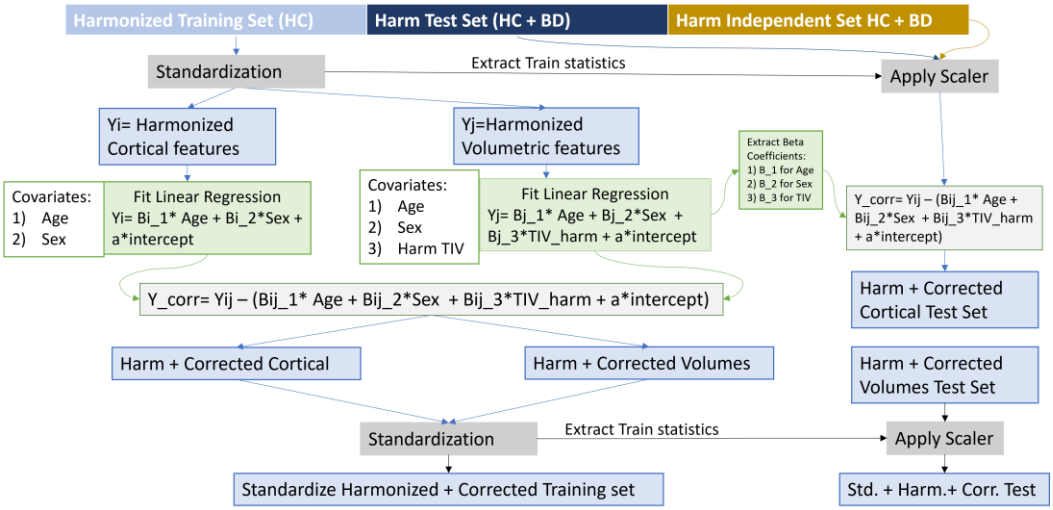



Figure 5.4: Regressing-out biological covariates integrated into pipelines 2,3,4,5 (input is harmonized data for 3,4,5).

The beta coefficients, β, that will result from this process have the dimension of the number of covariates (i) x number of brain features (j), thus 2 (age, sex) x 68 cortical brain features, for $\beta_{i\_cortical}$, and 3 (age, sex, harm/no harm TIV) x 52 volumes, for $\beta_{i\_volumes}$. To adjust data each of the beta coefficients is multiplied by the respective covariate data and subtracted to the respective brain feature. At the end of this process, data is further standardized. Having available the beta coefficients estimates of the training set, they are applied to the test set and external set, as described previously, by directly using them to adjust the test data.

## 5.7 Autoencoder Normative Model

### 5.7.1 The Autoencoder

The Autoencoder model was designed using *tensorflow 2*, an ML open-source python module and *keras* API built on top of *tensorflow 2* for easy deep learning models implementation. A random seed was set at the beginning of the model architecture definition to ensure the replicability of the model.

Using the *keras.layers* API a sequential model was built, making use of the Input layer and Dense layers. The AE is composed of 5 layers, including input and output with 120 hidden units. The hidden layers dimension is to be chosen, as well as the activation function, parameter initializer, and layer regularization.

In the encoder part, the dimension is forced to decrease while in the decoder part is forced to increase mirroring the dimensions of the encoder. The activation function and parameter initialization were chosen a priori and were not included in the hyperparameter search. For the three hidden layers, the Scaled Exponential Linear Unit (SELU) was chosen as the activation function, paired with the *lecun_normal* parameter initializer, enforcement of the AF itself. The output layer is composed of a linear activation function and *gorot_uniform()* parameter initializer. Besides the model architecture design, we chose Adam optimizer as the network optimizer, paired with the MSE loss function and a learning rate that was tuned during hyperparameter tuning. Regarding the remaining algorithm hyperparameters, batch size and the number of epochs was fixed to 30 and 2000, respectively, however, the *EarlyStopping* option from the keras.callbacks API was used, which stopped the training process when overfitting was identified. For the *EarlyStopping* option, the monitorization metric used was the MSE of the validation set, with a patient of 250 epochs. Thus, while training the model, if the MSE of the test set wasn't improving for 250 epochs, the training stops, and the best model weights are restored. The training data was shuffled at the beginning of each epoch to avoid overfitting, by setting the option shuffle=True.

**Hyperparameter tuning**

The hyperparameter tuning was performed to two hidden layer dimensions -the first hidden layer (i.e., second network layer) and the bottleneck- as the third hidden layer is forced to have the same dimension as the first one (decoder part), to layer regularizer, equally used in all layers, and learning rate. A grid search was performed but not using the *GridSearchCV* function from *sklearn.model_selection* due to the data transformations that must be performed within each fold.

The pipeline was the following: the number of splitting iterations was defined as 10 and performed to the training set making use of the *StratifiedKFold* function from *sklearn.model_selection* class, stratifying for center proportions. The *ParameterGrid* was used to define a grid from the hyperparameter space which is defined by creating a dictionary with the hyperparameter options and values. For each hyperparameter combination *i*, a 10-fold splitting iteration initializes, and the model is trained, iteratively, with different 9 folds of the training set, and evaluated in the left-out fold, the validation fold.

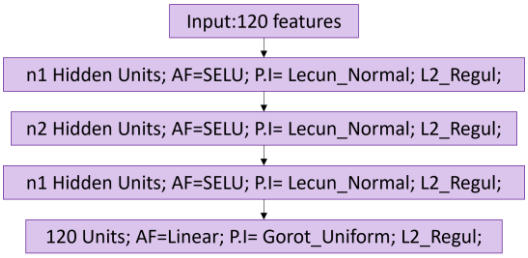



Figure 5.5: Network Architecture.

Once the training was done, the validation MSEs for the 10 iterations were averaged and saved. Finally, after the entire hyperparameter combinations have been tried out, the best hyperparameter combination was chosen according to which had the lower average validation MSE – thus lower reconstruction error. The best model was retrieved and the hyperparameter tuning ends. The best model was then re-trained with the entire training set.

**Model Evaluation metrics**

After model realization, the testing set composed of HC was used to evaluate AE performance, i.e., how good could it reconstruct an HC, whereas the BD subjects was used to evaluate anomaly detection capabilities. For the latter purposes, a Deviation Metric (DM) was defined, as the reconstruction MSE for all features or all subjects. At the subject level, it can be viewed as a reconstruction score, a proxy for the reconstruction error of each subject, whereas at the feature level, a reconstruction score for each feature within the HC group and BD group, denoting a local reconstruction effort by the network. These metrics allow us to evaluate the model reconstruction capabilities and to investigate differences between HC and BD patients' data.

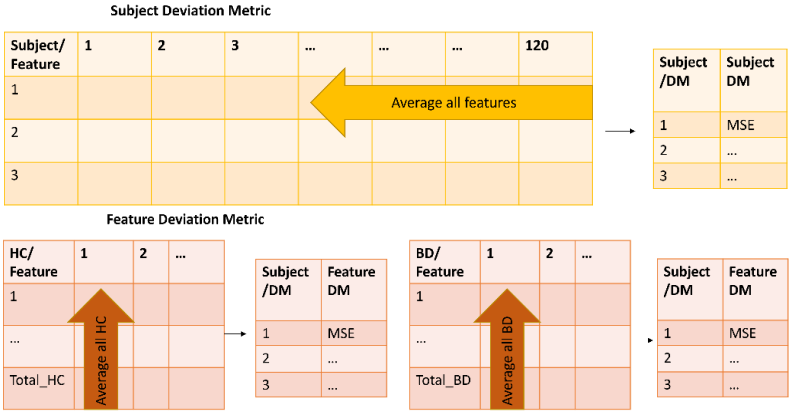



Figure 5.6: Deviation Metrics.

### 5.7.2 Normative Approach Framework

The following stage in the AE pipeline was that of employing the model for anomaly detection. The goal is to discriminate HC from BD patients. The trained model was tested in the test set, composed of both HC and BD subjects. Each subject had associated a reconstruction score, through the calculation of the subject DM. It was assumed that a BD patient would contain irregularities in data for which the AE has not learned to model, hence, data from BD patients should be more difficult to reconstruct, yielding a higher score than HC subjects.

**Discriminating HC and BD**

The first step to evaluate the normative model capacity to detect "anomalies", i.e., data that is not from HC subjects, is to perform a Mann-Whitney U test (MWU) to search for significant differences between HC and BD subjects' reconstruction error scores. The MWU one-sided test was applied to the subject DM, assuming the alternative hypothesis of BD-DM to be greater than HC-DM. The p-value was analyzed to confirm significant differences between the two groups. Once this step was performed, a ROC was carried out, using the subject DM as the data to be thresholded and the diagnosis as the binary target variable, 1 for BD and 0 for HC. Ideally, higher subject DM should be classified as BD, and if the network works as supposed, BD patients would have higher reconstruction error overall. The AUC results will let us conclude whether the latter point is true.

### 5.7.3 Feature Selection

The next stage is to identify abnormal brain regional morphologies belonging to the BD patients. For this purpose, each feature was considered separately, and the DM score was calculated for each subject as the square error between the original and reconstructed values. In this step, we used all BD subjects' dataset, 500 patients, instead of the BD test set, to have a higher statistical power in inspecting the deviating regions in the BD group. The BD and HC groups are compared with a one-sided MWU-Test, as shown in Figure 5.7, with the alternative hypothesis being reconstruction error greater in the BD subject group, and each feature was associated with a p-value. Besides, Cliff's delta absolute value was used to measure effect size. It was preferred to Cohen's d in our case because the reconstruction error distributions might not follow a normal distribution. It measures how often one value in one distribution is higher than in the other distribution.

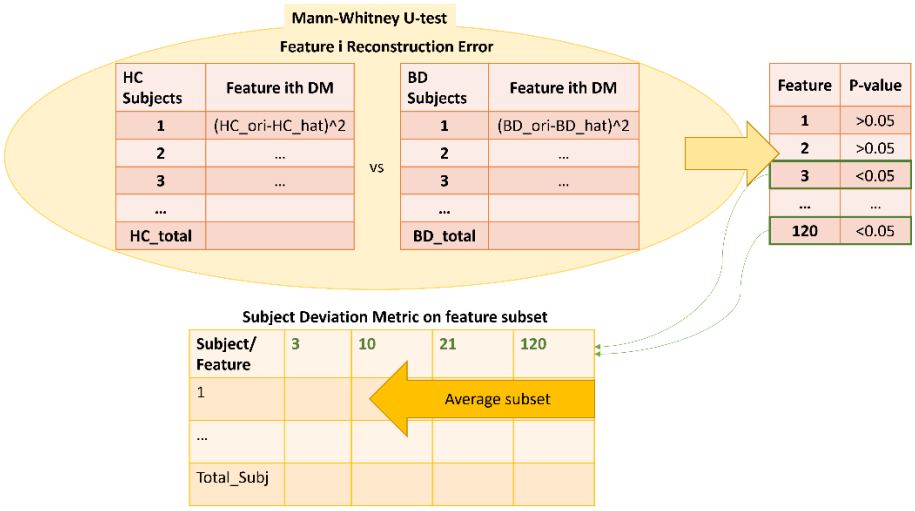



Figure 5.7: Subject DM on feature subset.

The brain features that were found to be associated with a significant p-value (i.e. p-value < 0.05) indicate the brain regional abnormalities belonging to the BD group, i.e., outside the normative range associated with HC. An MWU-Test was repeated this time on the Subject DM for the feature subset. If discrimination ability is improved, the p-value should reflect this. After finding the abnormal brain features for the BD group, a new evaluation of the discriminative power of the AE model was performed on the test set, by computing the AUC-ROC curve. Because the features were selected in the test set and the evaluation was re-performed in the test set, inevitably an increase in performance was expected, thus the AUC-ROC results should improve as this step is a form of circular analysis. Nevertheless, the AUC-ROC curve was re-performed, using the subject DM, which is calculated only on the former selected features.

#### *5.7.3.1 Classifying HC and BD*

After feature selection, a further validation step must be performed to test the classification performance of the discovered brain features subset. Since the subset of features is selected in the test set, this subset of features must be validated in an external independent set. Furthermore, the subset of features is supposed to represent a disease-specific-signature pattern, thus, it needs to be generalizable to an external set. The PITTS external set is thus passed through the network and the subject DM was calculated. If the brain feature subset that was found to be abnormal in the BD patient group is generalizable, it should have an equivalent performance to the test set, when classifying HC and BD patients in the external set. The AUC-ROC results on the entire feature set are compared to the results in the feature subset for the PITTS center data.

## 5.8 SVM Model

The SVM model was used as a baseline comparison for the BD classification performance since it is the most used ML model for neuroimaging data features and has been applied extensively to classify psychiatry disorders.

From the 7 centers contained in the dataset, one out of four was holdout as an external site set, specifically all data from MI_POLI, OSR, PITTS, and UBC sites, thus following a semi-LOSO-CV framework. The choice of these sites was based on the little amount of HC data it would remove from the training set since the previous centers contain much fewer data compared to AUOC, FSL_ROME, and JUH. Afterward, for each LOSO trial, the rest of the dataset was split into a training set and a test set, using the function *train_test_split* , 70% for the training set and 30% for the test set, stratifying for center proportions. The target variable was also retrieved, based on the concurrent split of the diagnosis covariate. In this case, since data on the training set included both HC and BD, the diagnosis was included as a biological covariate in the harmonization with ComBat.

The SVM model was imported from *sklearn.svm* library, namely *SVC()*, for which the probability option was set to True, to evaluate the model using the ROC curve. No hyperparameter tunning was performed here, instead, the SVM model which was used was the one reported in the ENIGMA study, employing multi-site data from HC and BD patients [17]. The SVM model uses a linear kernel and a fixed hyperparameter C=1.

The analysis performed with the SVM model was both LOSO-CV using multi-site data for which processing pipelines 1 and 5 were analyzed, and a site-level analysis, equivalent to the analysis performed in the ENIGMA study.

## 5.9 Comparison of Results

Finally, the results from all the pipelines were compared, through the AUC-ROC metric. The goal was to compare both the data processing pipelines and the results of the classification task of the AE model and SVM model.

We evaluated the results assessing the AE-based normative and SVM model, by comparing the outcomes obtained from pipeline 5, the application of CV-ComBat (i.e., harmonization option B), the classification performances when no harmonization is performed ( i.e., option A), pipeline and with diagnosis clinical state-of-art

Furthermore, for the normative approach, we report the comparison across all processing pipelines. Regarding the SVM model, we compared SVM results across

pipelines 1 and 5 for the LOSO-CV framework and the site-specific analysis, as well comparing with respect to ENIGMA SVM model.

The comparations that were performed were:

- **AE-based normative model**
    1. ***Processing Pipelines: 1,2,3,4,5***

       The goal was to understand how the harmonization process and biological covariate correction influenced the model's capability to generalize, validate the harmonization of an external set, and have a discussion about data leakage in an ML pipeline and how that can bias the final results.

    2. **Normative approach for BD**

       We assessed the successfulness of the normative approach in discriminating BD subjects.

- **SVM model**
    1. ***LOSO-CV with processing pipelines : 1,5***

       The goal was the same as *point 1* related to the normative approach. We assessed how harmonization influences the SVM classification and generalization performances.

    2. ***ENIGMA Study***

       We compared our SVM results on site-level and LOSO-CV analysis with processing pipeline 1 (no data harmonization) with those achieved by the ENIGMA Study (which also does not employ data harmonization), as we used the same SVM model.

- **AE-based normative model compared to SVM model and clinical state-of-art**
    1. ***CV-harmonization processing pipeline 5: option B+C***

       We set ourselves to understand whether a normative approach for the classification of BD disorder would yield better or comparable results to a classic SVM model. Thus, we determined whether there is a significant advantage of using this approach specifically with BD patients' data.

    2. ***Processing pipeline 1 (no data correction): option A***

       To determine how both models handle no harmonized data and which generalization performance it yields.

    3. ***Incremental utility of normative model compared to state-of-art***

We compared the results of the AE Normative Approach with the state-of-art clinical diagnostic performance and with the best ML state-of-art BD classification performance, which we consider the ENIGMA results.

# 6. Results

## 6.1 Demographic Results

Demographic analyses were performed on the dataset by comparing HC and BD subject groups, whose demographic characteristics are presented in Table 6.1 and Table 6.2, respectively. Their overall age distribution can be seen in Figure 6.1. It is noticeable that the BD group is slightly older than the HC group, on average, and within each group, females have a higher mean age than males.

**HC**

| | Male | Female | Total |
|---|---|---|---|
| **Sex** | 270 | 335 | 605 |
| **Age(years)** | 35.0±14.1 | 37.6±15.2 | 36.4±14.8 |

Table 6.1: Demographic data in HC group.

**BD**

| | Male | Female | Total |
|---|---|---|---|
| **Sex** | 242 | 316 | 558 |
| **Age(years)** | 40.3±15.5 | 42.0±14.4 | 41.3±14.8 |

Table 6.2: Demographic data in BD group.

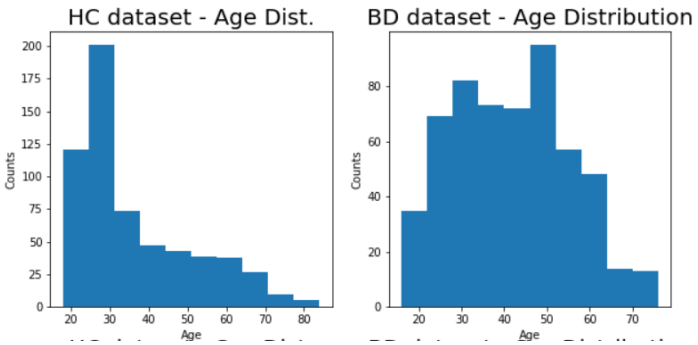



Figure 6.1: Age Distribution for HC and BD groups.

Besides, after dataset splitting, a demographic analysis was performed, comparing the training set with the test set and the external set (PITTS dataset) populations. The sex-specific age characteristics of the three datasets are reported in Table 6.3-Table 6.7. Lastly, in Table 6.8 the results of applying a two-sided MWU test comparing the HC training set to all other age distributions are presented, from which we can conclude that the only comparable age distribution is between the training set and HC test set.

**Training Set**

| | Male | Female | Total |
|---|---|---|---|
| **Sex** | 230 | 289 | 519 |
| **Age(years)** | 35.2±14.3 | 38.7±15.5 | 37.1±15.0 |

Table 6.3: Demographic data in the Training Set.

**Test Set HC**

| | Male | Female | Total |
|---|---|---|---|
| **Sex** | 27 | 31 | 58 |
| **Age(years)** | 36.9±14.8 | 31.3±12.7 | 33.9±14.0 |

Table 6.4: Demographic data in the Test Set HC.

**Test Set BD**

| | Male | Female | Total |
|---|---|---|---|
| **Sex** | 34 | 41 | 75 |
| **Age(years)** | 39.5±12.4 | 41.0±12.9 | 40.3±12.7 |

Table 6.5: Demographic data in Test Set BD.

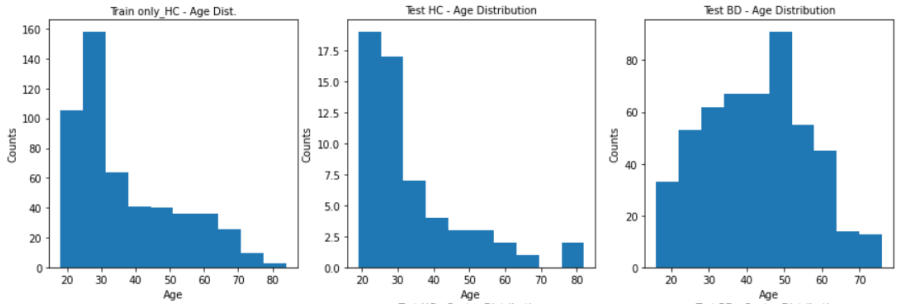



Figure 6.2 Age Distribution in Training set, Test Set HC, and Test Set BD.

**PITTS External Set HC**

| | Male | Female | Total |
|---|---|---|---|
| **Sex** | 13 | 15 | 28 |
| **Age (years)** | 28.3±4.5 | 28.9±4.7 | 28.6±4.6 |

Table 6.6: Demographic data in HC from PITTS center.

**PITTS External Set BD**

| | Male | Female | Total |
|---|---|---|---|
| **Sex** | 24 | 34 | 58 |
| **Age (years)** | 27.1±4.3 | 29.4±4.7 | 33.8±10.4 |

Table 6.7: Demographic data in BD from PITTS center.

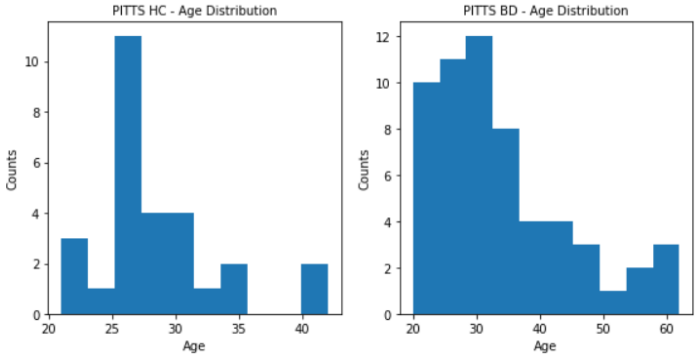



Figure 6.3: Age Distribution of HC and BD from external PITTS center.

| Age | HC: Train vs Test | HC: Train vs PITTS | HC Train vs BD test | HC Train vs BD PITTS |
|---|---|---|---|---|
| **statistics** | 13324.5 | 5457.5 | 100012.0 | 5625.0 |
| **p-value** | 0.065 | 0.013 | $1.197_{10}$-10 | 0.005 |

Table 6.8: MWU test on age distributions.

## 6.2 Harmonization Results

To evaluate and confirm that data was successfully harmonized, we apply PCA to data, before and after harmonization, and check whether with the 2 principal components (PCs) it was possible to distinguish data based on its center label, i.e., whether the 2 components captured site-related data variance. The PCs scores were colored by center, in order to visualize whether the orthogonal directions of variance of the ROI-based measures were associated with the center before and after harmonization.

If data is successfully harmonized, the clusters representing the PCs scores from different centers should be confused, thus there should be no visible and differentiated clusters corresponding to the center. The PCA results relative to different harmonization pipelines are illustrated hereinafter. The labels for each center are respectively:

| 1: AUOV | 2: FSL_ROME | 3: JUH | 4: MI_POLI_3T_3 | 5: OSR | 6: PITTS | 7: UBC |
|---|---|---|---|---|---|---|

**A. No Harmonization:** We apply PCA to the non-harmonized whole data set. It is visible at least one clear cluster corresponding to center 5 (i.e., OSR), which likely employs a very different MRI acquisition protocol from the others.

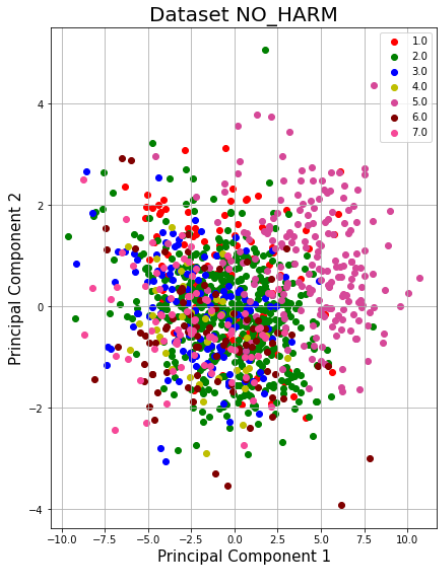
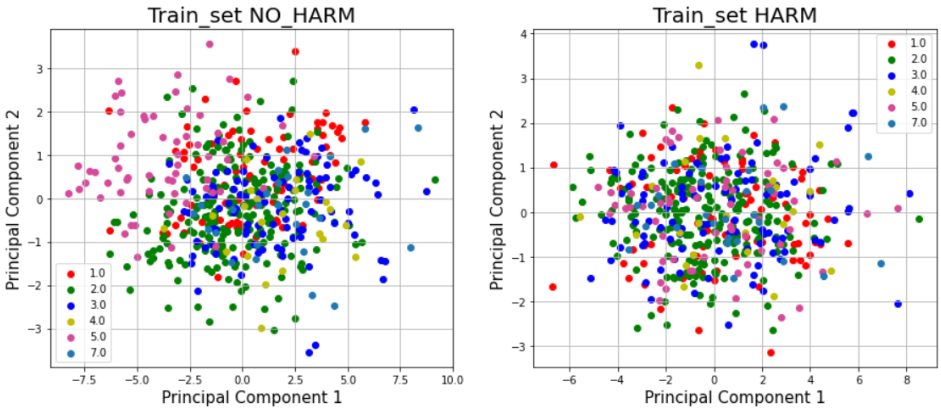
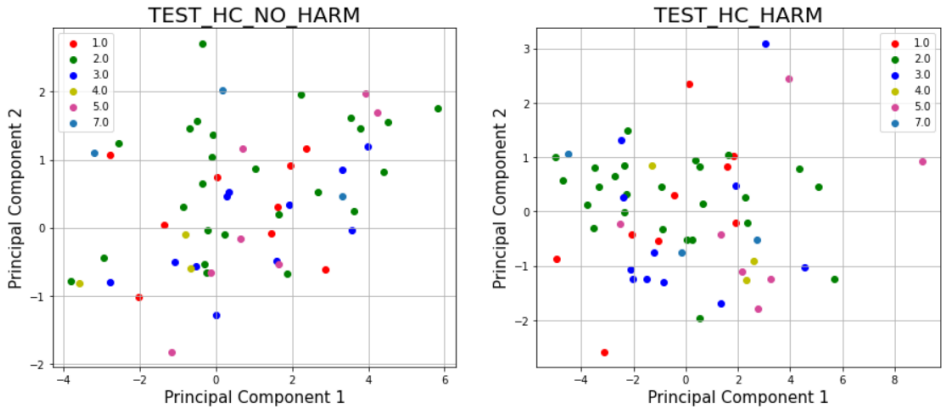



Figure 6.4: First and Second PCs extracted from the original raw data.

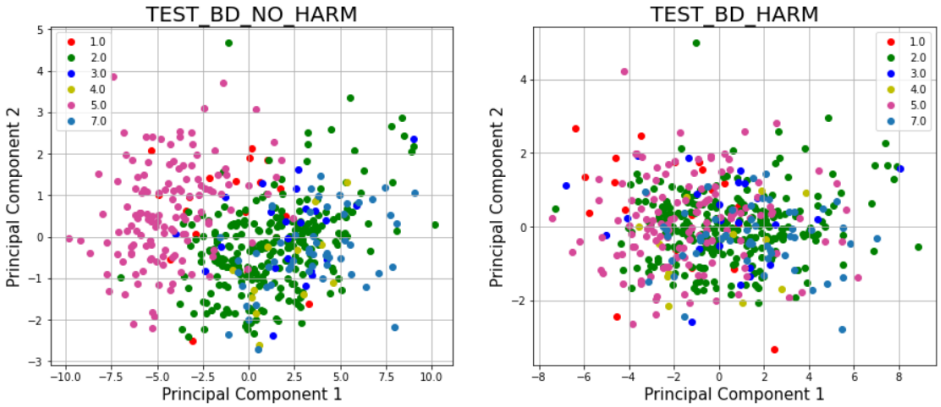
**B. CV-Harmonization:** This option is used in pipeline 5, where harmonization is performed after dataset splitting, respecting the CV framework, i.e., the center adjustment parameters estimated by ComBat are estimated in the training set, applied to the training set and test set like-wise. This is made possible by the fact that data in the training set and test set belong to the same centers. The test set harmonization effectiveness is clearly seen in Figure 6.5c), corresponding to the BD dataset, particularly for the cluster represented by center 5 OSR, before and after harmonization.




a)**Training Set.**




b)**Test Set HC.**




c)**Test Set BD.**

Figure 6.5: First and Second PCs extracted before (left) and after (right) ComBat harmonization of Training (a) and Test set (b,c).

C. **Harmonization of an External Set:** This option is necessary as the external set belongs to an external center (PITTS), therefore there are no center adjustment parameters estimated from the training set to apply directly to the external set. Thus, the harmonized training set is used as a reference batch for the estimation of PITTS site effects, as discussed previously in section 5.5. In Figure 6.6, the red label 0 corresponds to the previously harmonized training set, composed of harmonized examples from centers 1, 2, 3, 4, 5, and 7, all represented as a unique center (i.e., cluster). The green label 6.0 are represented the samples from PITTS center (i.e., center 6), before and after harmonization. It is noticeable that PITTS data before harmonization was already not isolated from the main cluster (i.e., harmonized training set), hence the visible changes on the PITTS cluster, after harmonization, are very subtle.

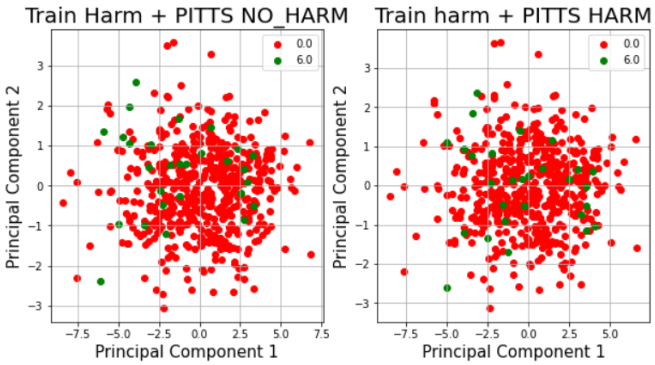
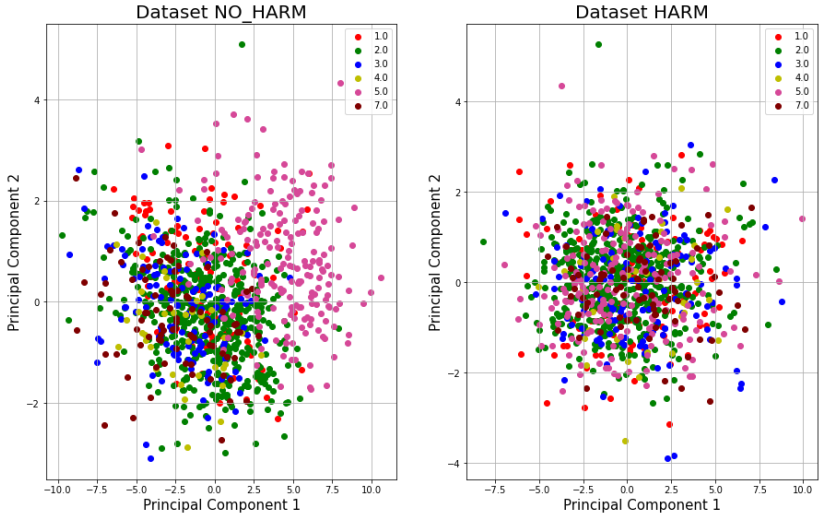


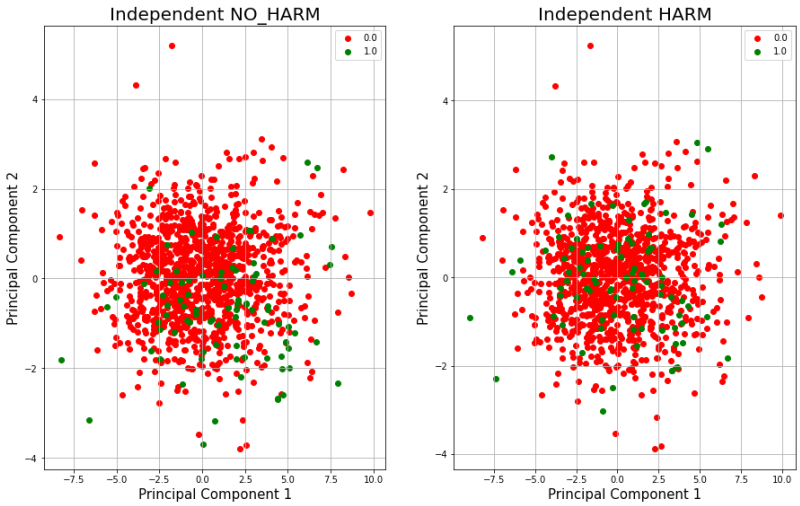

Figure 6.6: First and second PCs before (left) and after (right) ComBat harmonization of the external test set, PITTS, indicated by green label 6.0.

## D. Whole-Dataset Harmonization

**D.1+C Leave out an External Set:** The whole dataset is harmonized, without PITTS data. We keep the PITTS center as an external set.




a) Whole Dataset without External Set (PITTS data).




**b)** External set (PITTS data).

Figure 6.7: First and Second PCs extracted before and after ComBat (D+C) from (a) the whole dataset (excluding PITTS data) and (b) from the external set (PITTS data).

**D.2 Harmonize all 7 Centers:** The whole 7 centers dataset is harmonized, before the splitting of data in the CV framework.

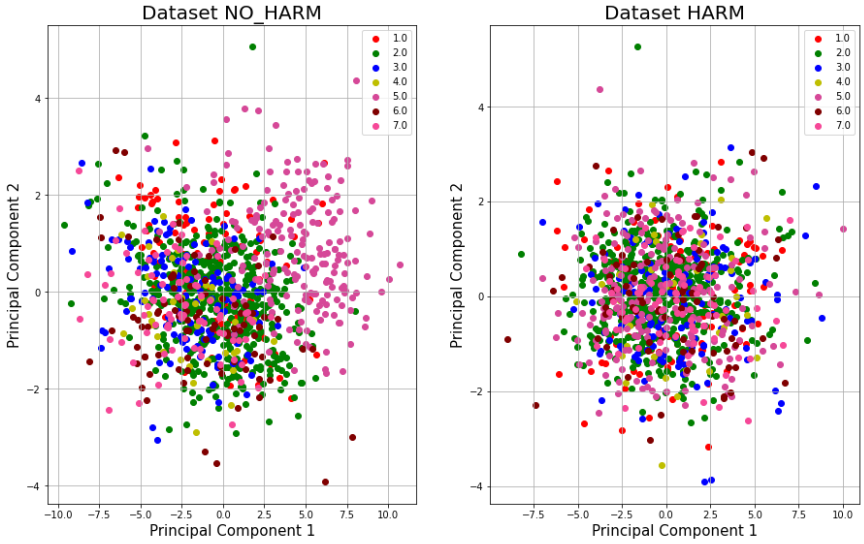



Figure 6.8: First and Second PCs extracted before and after ComBat (D) from the whole dataset (including PITTS data).

An observation that is worth to be highlighted is that we do not find substantial site effects between all centers, except for center 5 OSR, as it can be concluded by not observing clear isolated center clusters on the scatterplots of the two PCs, extracted

on non-harmonized data. After assessing all harmonization options, we can conclude that option B, harmonization within the CV framework, used in processing pipeline 5, was as rigorous and effective as the harmonization option D, which accounted for harmonizing the entire dataset (hence, more numerous data). In option B, CV-Harmonization, ComBat site effects are estimated in the training set, composed of 519 samples, while option D, WD-Harmonization, accounted for 1163 samples. Regardless, the results of a posteriori test set harmonization in option B, seen in Figure 6.5c), clearly show that the estimated site effects in a smaller sample as the training set, are reliable to be applied separately to the test set.

## 6.3 Regressing-out biological confounders

Linear regression is fitted to data considering the brain features as the dependent variable and the biological covariates as the independent variables. In this procedure, we assume that the brain features and biological covariates have a linear relationship and that the covariates' joint effect is the sum of their separate effects. This approach has the advantage of being simple and straightforward, but the drawback is the risk of misspecification.

To analyze the validity of such an assumption we investigate several statistics such as the significance of the coefficients through the p-values of the corresponding t-tests, and the analysis of the variance through the p-value of the model F-statistics. As explained in section 3.2, to validate a regression model, the variance of the dependent variable explained by the predictive variables must be significant, thus F-p-value<0.05. Not only, to see whether the biological covariates do have a linear relationship with the brain feature as hypothesized, we see if the dependency of the dependent variable on the predictive variables is significant, thus T-p-value <0.05.

As an example, in Figure 6.9 we can conclude that the cortical thickness of the right hemisphere insula cannot be well predicted from age and gender, having a non-significant F-statistics (p-value=0.122) and gender beta coefficient distribution clearly includes the zero value, thus the negative linear relationship is not significant. Overall, the linear regressions model fitting, estimated in the training set, were satisfactory, with 12 out of 68 from the cortical thickness features and only 2, right and left fornix, out of 52 from the volumetric features not achieving a significant F-p-value (<0.05), Figure 6.10 and Figure 6.11. We can say that the age covariate had the worst performance for anatomical volumes predictions, thus its linear relationship with many volumetric brain features was not proved. The latter might be due to TIV inclusion in the linear model.

OLS Regression Results

| | | | |
|---|---|---|---|
| Dep. Variable: | 67 | R-squared: | 0.008 |
| Model: | OLS | Adj. R-squared: | 0.004 |
| Method: | Least Squares | F-statistic: | 2.109 |
| Date: | Thu, 09 Jun 2022 | Prob (F-statistic): | 0.122 |
| Time: | 08:06:55 | Log-Likelihood: | -769.79 |
| No. Observations: | 544 | AIC: | 1546. |
| Df Residuals: | 541 | BIC: | 1558. |
| Df Model: | 2 | | |
| Covariance Type: | nonrobust | | |

| | coef | std err | t | P>\|t\| | [0.025 | 0.975] |
|---|---|---|---|---|---|---|
| const | 0.2405 | 0.125 | 1.921 | 0.055 | -0.005 | 0.486 |
| age | -0.0058 | 0.003 | -1.998 | 0.046 | -0.012 | -9.66e-05 |
| gender | -0.0611 | 0.087 | -0.704 | 0.482 | -0.232 | 0.109 |

| | | | |
|---|---|---|---|
| Omnibus: | 11.747 | Durbin-Watson: | 1.858 |
| Prob(Omnibus): | 0.003 | Jarque-Bera (JB): | 11.906 |
| Skew: | -0.339 | Prob(JB): | 0.00260 |
| Kurtosis: | 3.255 | Cond. No. | 123. |

Figure 6.9 Summary Statistics of Linear Regression for Cortical Thickness measure of Insulta on Right Hemisphere.

The age coefficients are estimated at net TIV, which in itself brings the most age-related effects. Sex linear relationship was also not significant with many of the brain features, both cortical thickness and volumetric measures. On the other hand, TIV performed well as a predictor variable for volumetric brain features. These results are reported extensively in Appendix A.

| | regions | t_pvalue_age | t_pvalue_sex | F_pvalue | R_square |
|---|---|---|---|---|---|
| 2 | [lcaudalanteriorcingulate] | 0.01948 | 0.75710 | 0.06506 | 0.00639 |
| 8 | [lentorhinal] | 0.38864 | 0.71480 | 0.66501 | -0.00218 |
| 9 | [rentorhinal] | 0.60388 | 0.35622 | 0.59904 | -0.00180 |
| 24 | [lmedialorbitofrontal] | 0.06379 | 0.55796 | 0.12993 | 0.00385 |
| 28 | [lparahippocampal] | 0.42363 | 0.72837 | 0.65837 | -0.00215 |
| 29 | [rparahippocampal] | 0.54012 | 0.45176 | 0.65456 | -0.00213 |
| 49 | [rrostralanteriorcingulate] | 0.01985 | 0.85367 | 0.05976 | 0.00670 |
| 61 | [rfrontalpole] | 0.11548 | 0.48517 | 0.25279 | 0.00139 |
| 62 | [ltemporalpole] | 0.45692 | 0.02147 | 0.06314 | 0.00650 |
| 63 | [rtemporalpole] | 0.66397 | 0.05887 | 0.16365 | 0.00300 |
| 66 | [linsula] | 0.11404 | 0.35838 | 0.21776 | 0.00194 |
| 67 | [rinsula] | 0.04626 | 0.48201 | 0.12230 | 0.00407 |

Figure 6.10 Cortical thickness regions with non-significant F_statistics.

| | regions | t_pvalue_age | t_pvalue_sex | t_pvalue_TIV | F_pvalue | R_square |
|---|---|---|---|---|---|---|
| 19 | [lFor] | 0.26649 | 0.05550 | 0.16547 | 0.19866 | 0.00307 |
| 45 | [rFor] | 0.71614 | 0.08329 | 0.05553 | 0.22737 | 0.00248 |

Figure 6.11 Neuroanatomical volumes' regions with non-significant F_statistics.

## 6.4 Model Optimization

The model optimization was performed employing a grid search technique within a 10-fold CV framework. Because the time and memory resources requests are very high for a 10-fold CV hyperparameter tuning, the parameter grid, composed of 79 combinations, was divided into 3 subgrids, so 3 search steps were performed, each step containing 10 iterations of the 10-fold CV for each parameter combination evaluation, totaling 790 iterations.

The fixed hyperparameters were:

- Loss Function: Mean Square Error, 'MSE'
- Activation Function: SELU
- Parameter Initializer: Lecun_normal
- A.F output layer: Linear
- Parameter Initializer output layer: Gorot_Uniform
- Optimizer: Adam
- Batch size: 35
- Epochs: 2000 with early stopping (patient 250 epochs)

| Trial Combination | **Layer 2,4** | **Layer 3** | **Regularizer** | **Learning Rate** | **Search Step** | **Best MSE validation** |
|---|---|---|---|---|---|---|
| **1.1** | **100**,80 | **75** | L2: 0.00001, **0.0001**, 0.001, 0.01 | **0.0001**, 0.001, 0.01, lr_schedule | 1-10th | 0.09665 |
| **1.2** | 100,**80** | **75** | L2: 0.00001, **0.0001**, 0.001, 0.01 | 0.0001, 0.001, 0.01, **lr_schedule** | 11-27th | 0.09604 |
| **2.1** | **100** | **80**,60 | L2: 0.00001, **0.0001**, 0.001, 0.01 | 0.0001, **0.001**, 0.01, lr_schedule | 0-6th | **0.08983** |
| **2.2** | **100** | **80**,60 | L2: 0.00001, **0.0001**, 0.001, 0.01 | **0.0001**, 0.001, 0.01, lr_schedule | 7-18th | **0.08570** |
| **2.3** | **100** | **80**,60 | L2: **0.00001**, 0.0001, 0.001, 0.01 | **0.0001**, 0.001, 0.01, lr_schedule | 19-31th | 0.09902 |
| **3** | **100** | **85**,70, 65 | L2: 0.00001, **0.0001** | **0.0001**, lr_schedule | - | **0.07390** |

*Layer 2 and 4 have the same dimensionality in the AE. Layers 1 and 2 are input and output, with dimension 120.

Table 6.9 Hyperparameter Tunning Results.

The learning rate schedule, denoted as *lr_schedule* in the table above had the following fixed hyperparameters:

- Initial Learning Rate: 0.001
- Decay Step: number_subjects/ batch_size
- Decay Rate: 0.9977

Finally, the best-retrieved network architecture was from trial combination 3, reporting the following hyperparameters:

- Layer 1,3: 100
- Layer 2: 85
- Regularizer: L2 = 0.0001
- Learning Rate: 0.0001.

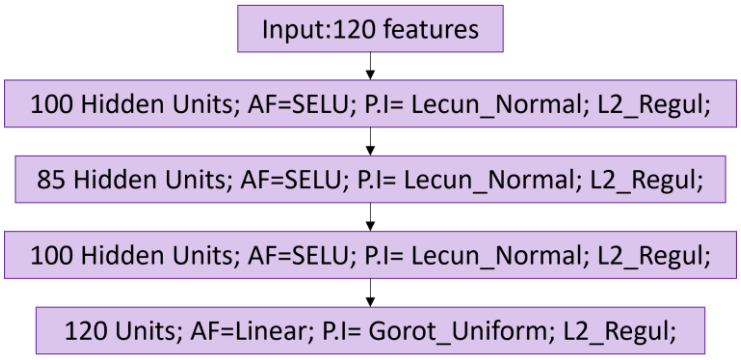



Figure 6.12: Best Network Architecture (hyperparameter combination from trial combination 3).

## 6.5 AE Model: Normative Approach

In this section, the results regarding normative model reconstruction and anomaly detection performances are reported for several processing pipelines. The test set is composed of 58 HC and 75 BD subjects and the external PITTS set of 28 HC and 58 BD. To select the significant deviating brain regional features in the BD group, we perform an additional analysis by considering all 500 BD subjects from the test set.

Data was processed according to the four harmonization options described in section 5.6 that were alternated or combined in the five processing pipelines resumed below.

Harmonization pipelines:

A. No Harmonization
B. CV-ComBat
C. Ref-ComBat
D. WD-ComBat

Given rise to 5 parallel processing pipelines:

**1) No Data Correction (A) Pipeline**

Including biological covariates correction:

**2) No Harmonization (A) Pipeline**

**3) Whole Dataset Harmonization (D) Pipeline**

**4) Whole Dataset Harmonization (D) + External Set Harmonization (C) Pipeline**

**5) CV-Harmonization pipeline (B+ C):** described in detailed section 5.6.1.

### 6.5.1 Data Processing Pipelines Results

All results in the following section were obtained using the best model retrieved from the hyperparameter tuning. The results for pipeline 5, using CV-Harmonization option B, will be presented in a detailed manner because we suggest that this processing pipeline, which respects the validation frameworks, is the most rigorous one. At the end of this section, a summary table will report the results for all five processing pipelines.

**-Best Model:** Model retrieved from the hyperparameter tunning, trial combination 3.

| | | | |
|---|---|---|---|
| **Layer 1,3:** 100 | **Layer 2:** 85 | **L2:** 0.0001 | **Lr:** 0.0001 |

- **Pipeline 5**

**I. Training**

The training was employed as described in sub-section 5.7.1. The average reconstruction errors on the training set and HC test set were 0.0278 and 0.0710, respectively, as stated in Table 6.10.

| Epochs | Train Loss | Train MSE | Test Loss | Test MSE |
|---|---|---|---|---|
| 2000 | 0.0419 | 0.0278 | 0.0850 | 0.0710 |

Table 6.10: Training Results from pipeline 5.

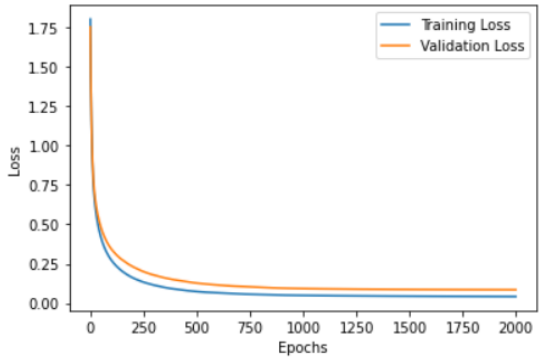
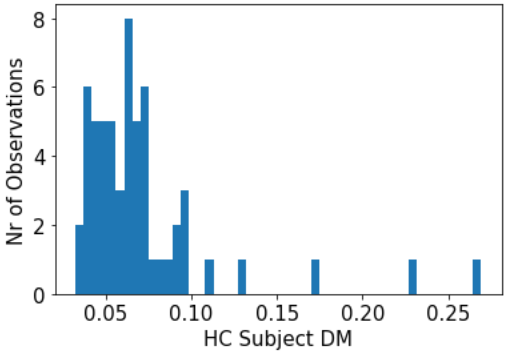



Figure 6.13: Training Evolution for pipeline 5.

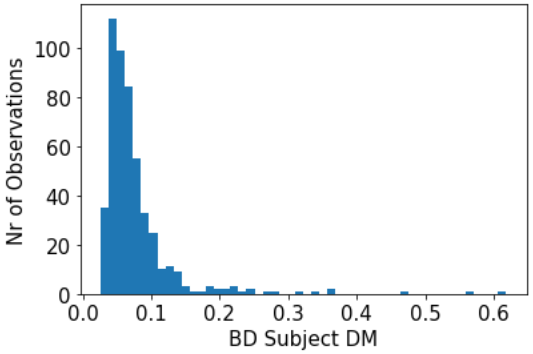
### II. Testing

Once the AE model was trained, the test set was passed through the network and each subject's data was reconstructed by the model. The DM distributions in the HC test set and BD dataset (i.e., all 500 patients) groups can be seen in Figure 6.14 a) and b). When visually comparing the distributions between HC and BD, it is not possible to see a clear distribution shift, rather both distributions are overlapping, as shown in Figure 6.14 c). This factor hints at a difficulty in detecting abnormal samples, which should be the BD patients. In Appendix A, two output data reconstructions are reported, one for the 20th subject from the HC Test Set and the other for the 56th subject from the BD dataset.

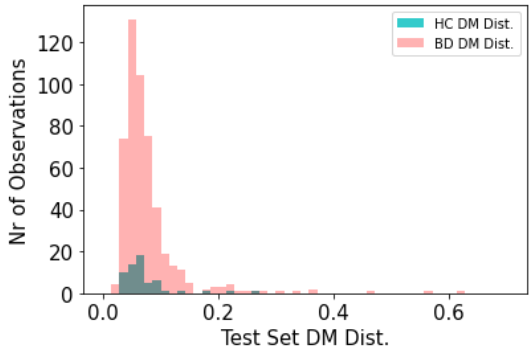
A one-sided MWU test is performed on the BD DM vs. HC DM, with alternative hypothesis option *greater*, to check whether BD subjects' DM is significantly greater than HC subjects' DM, with the resulting p-value=0.28172 (statistic=15172), thus clearly confirming BD group reconstruction error not to be significantly greater than HC group one.




a)HC Subjects.




b)BD Subjects.




c) Distributions Overlapping.

Figure 6.14: Reconstruction Error Distribution on Test set, for Pipeline 5.

The HC and BD subjects' DMs are used then to evaluate the discriminative power of the normative approach. Using a BD test set of 75 subjects, randomly selected from the 500 BD dataset, stratifying for center proportions, as explained in section 5.5, the AUC-ROC curve is performed. From Figure 6.15 it is possible to see that the ROC curve lies on the chance line level, with an AUC=0.51, which means the data classification is random, and no clear threshold can be found on the DM to discriminate between HC and BD.

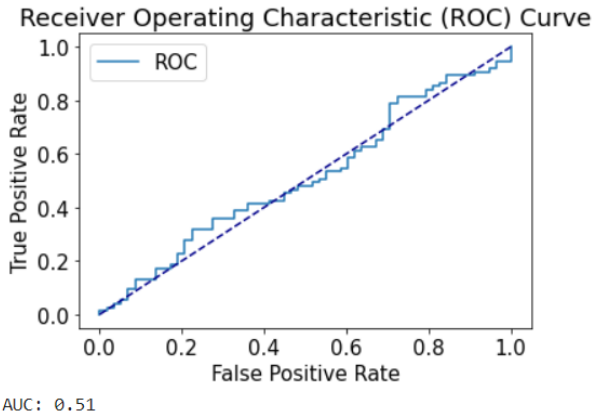



Figure 6.15: AUC-ROC curve on test set for pipeline 5: Discriminating HC vs BD subject.

### III. Feature Selection

Considering each feature individually, the DM within the 500 BD subjects is compared against the 58 HC test set subjects, and the features which are found to have a significantly greater DM score in the BD group are retrieved (p-value<0.05),

reported in Table 6.11, with the effect size measured by Cliff's delta absolute value. The effect sizes found are all small or negligible.

After selecting the BD abnormal brain regions with respect to HC, the subjects' DM is recalculated, for the test set, 58 HC and 75 BD, taking into account only those features, as explained in sub-section 5.7.3 Feature Selection. A new MWU test is performed on the new subjects' DM scores, to test whether considering only the subset of features, the BD group holds a greater reconstruction error, resulting in a p-value=0.001040 (Statistic=2854) thus confirming the alternative hypothesis. The AUC-ROC curve is re-performed, improving the discrimination power with an AUC=0.66, as reported in Figure 6.16. The latter is a circular analysis.

| ID | Regions | Statistic | p-value | Effect size |
|---|---|---|---|---|
| **24** | [lmedialorbitofrontal] | 16552.0 | 0.038778 | 0.1415 |
| **54** | [lsuperiorparietal] | 16973.0 | 0.016699 | 0.1706 |
| **75** | [lSupPostCerebLVI] | 16493.0 | 0.043238 | 0.1374 |
| **85** | [lHCA1] | 17200.0 | 0.010101 | 0.1862 |
| **95** | [rGloPal] | 17337.0 | 0.007335 | 0.1957 |
| **110** | [rAmy] | 16417.0 | 0.049584 | 0.1322 |

Table 6.11: Abnormal Brain Regions in the BD group compared to HC from Pipeline 5.

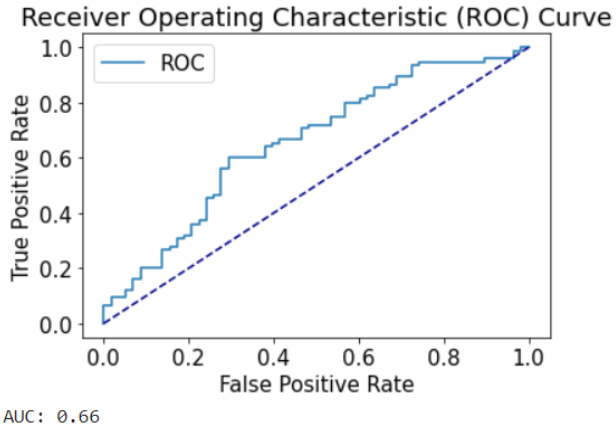
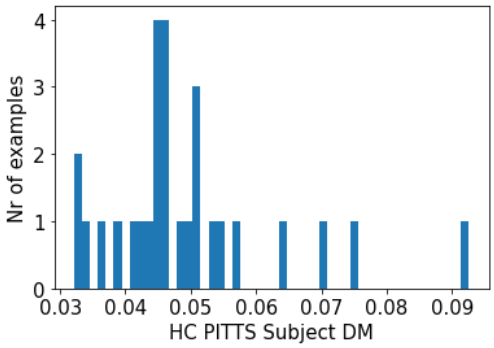



Figure 6.16: AUC-ROC curve on a subset of features for Pipeline 5.

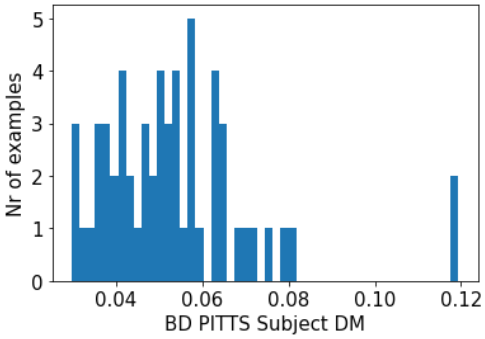
### IV. Classification HC and BD

To understand whether the subset of features is generalizable to BD we need to test if this subset of features yields a comparable discriminative power in an external independent set. This external set of data was never seen by the model in any step of the ML pipeline, nor has data from the same center acquisition been included in the

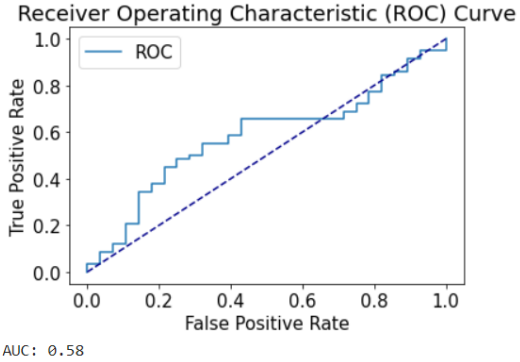
training and test set. The data from the PITTS center is inputted into the network and the subjects' reconstruction error scores are calculated. In Figure 6.17 a) and b) the subjects' DMs distribution are reported and in c) the AUC-ROC curve, resulting in an AUC=0.58, achieving higher discriminative performance than in the test set.



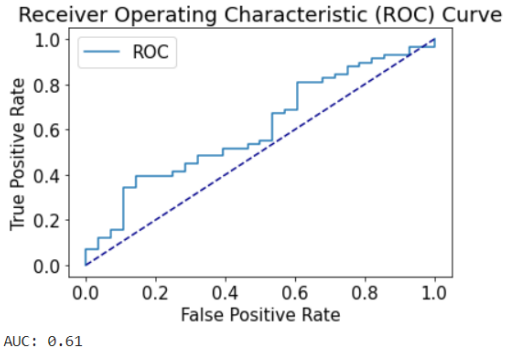

a) HC subjects' DM distribution.




b) BD subject' DM distribution.




c)AUC-ROC curve: all brain features.




d)AUC-ROC curve: feature subset

Figure 6.17: External set (PITTS data) results for Pipeline 5.

Then, the AUC-ROC curve is re-performed considering the subjects' DMs calculated on the subset of features selected in the test set. The result, seen in Figure 6.17 d), shows an AUC=0.61, which represents a slight improvement from c)- considering all brain features. As expected, the AUC obtained considering only the subset of features in the external set was lower than the one in the test set (AUC=0.66), but we should consider that the latter result comes from a circular analysis.

- **Summary Results**

In Appendix A the BD abnormal brain regions, for the feature selection step III, are reported for pipelines 1 to 4.

| Pipe-line | Test Set | | | | | | | External Set | |
|---|---|---|---|---|---|---|---|---|---|
| | HC Test DM | BD Test DM | All features | | | Feature Subset | | | All Feat. | Feat. Subset |
| | | | Stat. | p-value | AUC | Stat. | p-value | AUC | AUC | AUC |
| 1 | 0.0473± 0.0266 | 0.0591± 0.0525 | 16548 | .0391 | **0.56** | 3011 | 7.51e-5 | **0.69** | 0.45 | 0.51 |
| 2 | 0.0524±0.0 301 | 0.0607±0.0 4761 | 16241 | .0671 | **0.56** | 3149 | 5.00e-6 | **0.72** | 0.39 | 0.43 |
| 3 | 0.0654±0.0 333 | 0.0674±0.0 430 | 14673 | .4410 | **0.52** | 2668 | .0127 | **0.61** | 0.92 | 0.71 |
| 4 | 0.0669±0.0 355 | 0.0686±0.0 415 | 14825 | .3900 | **0.50** | 2752 | .0045 | **0.63** | 0.45 | 0.54 |
| 5 | 0.0710± 0.0417 | 0.0764± 0.0577 | 15172 | .2817 | **0.51** | 2854 | .0010 | **0.66** | 0.58 | 0.61 |

Table 6.12: Normative Approach results apply to the test set.

| Option | External Set | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| | HC PITTS DM | BD PITTS DM | All Features | | | Feature Subset | | |
| | | | Stat. | p-value | AUC | Stat. | p-value | AUC |
| 1 | 0.0446±0.0 144 | 0.0416±0.0 130 | 731 | 0.7737 | **0.45** | 825 | 0.4541 | **0.51** |
| 2 | 0.0483±0.0 122 | 0.0438±0.0 1382 | 626 | 0.9572 | **0.39** | 705 | 0.8391 | **0.43** |
| 3 | 0.0286±0.0 074 | 0.0518±0.0 195 | 1497 | 1.411e-10 | **0.92** | 1147 | 0.0010 | **0.71** |
| 4 | 0.0548±0.0 151 | 0.0518±0.0 151 | 726 | 0.7873 | **0.45** | 880 | 0.2669 | **0.54** |
| 5 | 0.0494± 0.0129 | 0.0534± 0.0174 | 942 | 0.1163 | **0.58** | 984 | 0.0570 | **0.61** |

Table 6.13: Normative Approach results on PITTS external set.

### 6.5.2 Discussion

- **Pipelines 1 vs. 2**

The processing pipeline that obtains the lowest reconstruction error on the test set is Pipeline 1 – No Data Correction (A). No harmonization option A is the one that presents higher anomaly detection capabilities in the test set, having an AUC=0.56

(for all features). When data is not harmonized but is corrected for biological covariates pipeline 2 (No Harmonization - Option A) the mean reconstruction error of the test set is higher but the discriminative performance maintains an AUC=0.56 (for all features). Regarding the feature selection, there is a slight increase in discriminative performance in the test set, comparing pipelines 1 to 2. Nevertheless, both these pipelines' results demonstrated a poor generalization capability, with AUC results in the external set below chance and in the chance line. The latter was expected because we knew the AE-based model could be able to learn center effects structure in data, thus failing to generalize when a new set presents a different site effects structure.

Interestingly, the AUC results on the external set decrease when data is corrected for biological covariates, from pipeline 1 to 2. To extract some interpretation from the latter fact we inspect differences between pipelines 1 and 2 if other centers were to be considered as external sets, as reported in Table 6.14. It can be seen in the reported table that an improvement happens in the discriminative performance on the external set, for pipeline 2 compared to 1, for the three external set trials.

| Pipeline | AUC | Test set | Ext. Set 4-MI_POLI | Test set | Ext. Set 5-OSR | Test set | Ext. Set 7-UBC |
|---|---|---|---|---|---|---|---|
| 1 | all | 0.55 | 0.48 | 0.56 | 0.64 | 0.61 | 0.51 |
| | subset | 0.66 | 0.35 | 0.65 | 0.63 | 0.70 | 0.53 |
| 2 | all | 0.54 | 0.53 | 0.58 | 0.70 | 0.60 | 0.53 |
| | subset | 0.70 | 0.44 | 0.60 | 0.65 | 0.72 | 0.55 |

Table 6.14: Pipeline 1 and 2 AUC results for external sets: 4,5,7.

The issue presented in the external set PITTS could be explained by different age distributions between train, test, and external set, as reported in Table 6.3Table 6.7. However, by assessing the age distributions for MI_POLI, OSR, and UBC, we conclude that the UBC center data is also demographically different from the respective training and test set, being composed of only young subjects ranging from 16 to 33 years old, with an age average of 22.7176±4.1885. Yet, the drop in the performance due to biological covariates correction is not posed. Hence, it seems that in PITTS data, correcting for biological covariates worsens the discriminative performance, as opposed to what happens for the other three external sets trials. This could be explained by a covariate shift in the PITTS dataset, where the estimated age, sex and TIV beta coefficients have a certain linear relationship direction with regional brain features that is reversed in the PITTS case. For example, age could have a significant negative linear relationship with some regional brain features in the

training set and in PITTS this relationship being positive or non-significant, thus eliminating some data variance that explains the diagnosis, worsening discriminative performance. It could also be an overestimation of the coefficients which would lead to the same result.

A new observation that can be drawn from Table 6.14 is that from the four centers experimented as external sets, only when using OSR as an external set, did the model manage to generalize for non-harmonized data (results in green). The reconstruction error in OSR was much worse (HC: 0.2068±0.100, BD:0.2976±0.2088) compared to what is shown in Table 6.13 for PITTS external set (HC: 0.0446±0.0144, BD: 0.0416±0.0130), nevertheless, the discriminatory performance was good (AUC=0.70). This is an interesting result because OSR was also the only center for which a clear separate cluster could be seen when plotting the 2 PCs extracted with PCA for non-harmonized data, Figure 6.4. We think the model was able to generalize well to the OSR center dataset, the center most affected by site effects, because all the others were already very homogenous and those were the ones included in the training set. By having a more homogenous training set, although data was not harmonized and the external center was not comparable, the model didn't learn site effects and was able to generalize to OSR both in pipelines 1 and 2.

Conclusive remarks: by assessing both pipelines, we have verified that not modeling the multi-site characteristics of the dataset through harmonization leads to good results in an internal validation framework but then the model fails to generalize to an external set. This is further confirmed by verifying that the model manages to generalize only for an external set when is trained with more homogenous data, even if not harmonized. We also verified that correcting for biological covariates, pipeline 2, worsens the discriminative performance in the PITTS center.

- **Pipeline 1 and 2 vs. 3**

In the test set there is a drop in the AUC metric from 0.56 to 0.52 (for all features), between pipelines 1,2 to 3. Harmonization thus seems to worsen discriminative performances in the test set (internal validation), which further confirms that the AE-based model was learning site effects encoded in data in pipelines 1 and 2, which helped improve its performance.

- **Pipelines 3 vs. 4**

The training set and test set results, in both processing pipelines, do not differ substantially. The best results on the external set were obtained with pipeline 3 (Harmonization option D), when the independency of this set is broken, by harmonizing it together with the rest of the dataset, resulting in an AUC=0.92 in the

external set. This result is rather high, considering that in the test set the result is a random classification, AUC=0.52. It possibly indicates that the results in the test set when grouped by center origin are heterogenous and this heterogeneity is canceled out by the averaging effect. However, pipeline 4 was mainly designed to witness how not taking into account a CV framework positively biases the results potentially leading to false optimistic conclusions. When the procedure is repeated but this time leaving PITTS as an external set and harmonizing it separately with *ref_comBat*, pipeline 4 (Harmonization option D+C), the performance on the external set drops to the chance line (AUC=0.54), from the previous AUC=0.92.

Conclusive remark: by assessing both pipelines we can verify that performing harmonization prior to dataset splitting might positively bias the results and does not model an external validation framework.

- **Pipelines 4 vs. 5**

In both pipeline 4 and pipeline 5 the external set was harmonized a posteriori, using harmonization option C, however, the results in pipeline 4 for the external set are much worse. The possible reason why in pipeline 4 the performance in the external set drops to the chance line might be related to the chosen model. As mentioned previously, for the sake of comparability and to reduce the number of estimations to be made, the hyperparameter combination was chosen by performing 10-fold CV, while integrating processing pipeline 5 in each fold. Thus, the best model is the best model for data processed according to pipeline 5. Even if a better model could be found for processing pipeline 4 yielding comparable results to pipeline 5 we would still argue that the test set in pipeline 4 is not independent of the training set because of WD-Harmonization, and it would only indicate that including more data in the harmonization process could be an advantage to estimate site effects, which is expected.

Conclusive remark: the aim of including pipeline 4 was to assess that the performance would drop from pipeline 3 to 4 in the external set. The magnitude of this drop or the actual AUC results is not relevant because there could still be a model yielding a better performance for data processed according to pipeline 4 than the chosen one (and perhaps comparable to pipeline 5 results).

- **Discriminative performance in the test set**

All processing pipelines yield chance or close to chance line performances on the test set when all features are considered. Both the training and test set result from pooling data for random partition, thus, containing data from all sites split randomly. While in the external set, data is not partitioned by a random factor, thus, the data center characteristic that might influence positively or negatively the model

reconstructions and discriminatory performance will directly reflect the evaluation metrics, while in the test set this factor might be canceled out, as mentioned previously in discussion pipeline 3 vs.4. The hypothesis is that data, even though harmonized and corrected for biological covariates, is still heterogeneous, and this heterogeneity is highly specific for each site cluster. Thus, evaluating model performance at the site level should give heterogeneous results. This can be shown by projecting the AUC-ROC results grouped by center origin, within the test set, for pipeline 1 and pipeline 5, Table 6.15. The AUC results within each scenter differ very much, ranging from 0.3 to 1 on pipeline 1 for all features and ranging from 0.3 to 0.67 on pipeline 5 for all features. Notably, only pipeline 5 performance improves or maintains for all sites in the feature subset. Results at the site-level analysis for pipeline 5 are equally heterogenous compared to pipeline 1 possibly pointing to the fact that there exist clinical variables which are center-specific and not modeled for, for example, a specific center might be composed of more severe cases of BD patients, thus making it easier to discriminate than within others. However, samples for each center are imbalanced regarding diagnosis and have few data, and such a limitation hinders our ability to conclude if the source of heterogeneous results at the site-level are due to the latter or due to clinical patient heterogeneity covarying with site origin.

| Pipeline | Features | 1-AUOV | 2-ROME | 3-JUH | 4-MI_POLI | 5-OSR | 7-UBC |
|---|---|---|---|---|---|---|---|
| | | 9HC 3BD | 25HC 39BD | 11HC 3BD | 3HC 2BD | 7HC 20BD | 3HC 8BD |
| 1 | all | 0.481 | 0.533 | 0.303 | 1 | 0.579 | 0.583 |
| | subset | 0.741 | 0.732 | 0.454 | 0.333 | 0.628 | 0.458 |
| 5 | all | 0.407 | 0.556 | 0.303 | 0.667 | 0.5 | 0.667 |
| | subset | 0.851 | 0.58 | 0.393 | 1 | 0.6 | 0.667 |

Table 6.15: AUC test set results grouped by center.

Concluding remark: It is clear from the above table that harmonized and corrected data still yields heterogeneous discriminative AUC results across sites – comparing pipelines 1 and 5 at site-level analysis. Thus in the test set, which is composed of subjects belonging to 6 different centers, the heterogeneous results are canceled out, due to an averaging effect, and we get an overall result close to chance classification. The source of these heterogeneous results could be within-site numerosity or clinical heterogeneity covarying with site origin.

- **Pipeline 5**

The data processing pipeline which we were sure respected the CV framework, reported as Pipeline 5 (Harmonization option B+C), achieved unsatisfactory results on the external set. When all features were considered, it achieved an AUC=0.58 compared to an AUC=0.51 in the test set. The latter result might be a reflection of the previous discussion and the AUC results grouped by center on the test set for pipeline 5 are shown in Table 6.15. The test set is intrinsically heterogeneous due to the random partition while the external set is not. Then, the AUC in the feature subset was 0.61, improving slightly from the AUC of 0.58 considering all features. A higher improvement would be reassuring but we conclude the subset of features do generalizes at some level to the external set, as they help improve discriminatory performance. This is also the best classification performance from the AE-based normative model results.

- **BD Abnormal Brain Regions**

The most difficult features for the model to reconstruct in the BD group were: left medial orbital frontal, left superior parietal, Left Superior Posterior Cerebellar Lobule VI, left Hippocampus CA1, Right Globus Pallidus, and right Amygdale. As described in section 1.2, there have been reports of alterations in volumes of the amygdala, and hippocampus. In the ENIGMA study [9], both left medial orbital frontal and left superior parietal had shown significantly reduced cortical thickness in BD patients, although not with the highest effect size. However, as suggested in Pinaya et al. [57], this normative approach represents a multivariate analysis method, thus features that are found to have high discriminative power should be interpreted as a spatially distributed pattern rather than individually. The founded feature pattern certainly is model-dependent and not necessarily linked to disease pathophysiology. Because the model is only capable of discriminating HC and BD locally, the differences that exist at the sMRI level are very subtle and may not represent, at least individually, a tool to help diagnose BD disorder. Besides, clinical data such as disease status or medication were not included in this study. Further analysis must be performed to test the normative approach hypothesis.

## 6.6 SVM Classification Results

In this section, the results of the SVM classifier will be reported for pipeline 1 (Harmonization Option A) and pipeline 5 (Harmonization Options B+C. The choices of the processing pipelines included here are related to the fact that to compare our SVM results with ENIGMA study results, we have to employ pipeline 1, and pipeline

5 to compare the SVM model with the AE-based normative model results, as it represents the most rigorous processing pipeline.

The SVM model did not undergo hyperparameter tunning, it employs a linear kernel and C=1. Because of this, a LOSO-CV was performed, and the model was tested on 4 external sets, by leaving one of these centers out of the train and testing set at each trial:

- Milano Policlinico (MI_POLI)
- Ospedale San Raffaele, Milano, Italy (OSR)
- University of Pittsburgh, Pittsburgh, US (PITTS)
- University of British Columbia, Vancouver, Canada (UBC)

By iteratively considering 4 different external sets we could report more robust results on the external set performance of the model. Finally, a site-level analysis will be performed, where each center data will be used to train and test the SVM model individually. In Appendix A detailed information is reported for each LOSO-CV iteration.

- **External set: MI_POLI**

| | Training set | Test set | External set |
|---|---|---|---|
| **HC** | 406 | 174 | 26 |
| **BD** | 382 | 164 | 12 |

Table 6.16: Dataset Split for MI_POLI external set.

| Pipeline | Set | AUC | Precision | Recall | F1-score |
|---|---|---|---|---|---|
| **1** | Test Set | 0.65 | 0.58 | 0.6 | 0.59 |
| | Ext. Set | 0.66 | 0.38 | 0.5 | 0.43 |
| **5** | Test Set | 0.54 | 0.54 | 0.54 | 0.54 |
| | Ext. Set | 0.59 | 0.41 | 0.58 | 0.48 |

Table 6.17: Results for MI_POLI external set.

- **External Set: OSR**

| | Training set | Test set | External set |
|---|---|---|---|
| **HC** | 377 | 161 | 67 |
| **BD** | 297 | 128 | 133 |

Table 6.18: Dataset Split for OSR external set.

| Pipeline | Set | AUC | Precision | Recall | F1-score |
|---|---|---|---|---|---|
| **1** | Test Set | 0.62 | 0.50 | 0.45 | 0.47 |
| | Ext. Set | 0.42 | 0.51 | 0.49 | 0.55 |
| **5** | Test Set | 0.54 | 0.50 | 0.45 | 0.47 |
| | Ext. Set | 0.54 | 0.74 | 0.26 | 0.38 |

Table 6.19: Results for OSR external set.

- **External Set: PITTS**

| | Training set | Test set | External set |
|---|---|---|---|
| **HC** | 403 | 174 | 28 |
| **BD** | 350 | 150 | 58 |

Table 6.20: Dataset Split for PITTS external set.

| Pipeline | Set | AUC | Precision | Recall | F1-score |
|---|---|---|---|---|---|
| **1** | Test Set | 0.64 | 0.57 | 0.56 | 0.57 |
| | Ext. Set | 0.47 | 0.30 | 0.43 | 0.35 |
| **5** | Test Set | 0.55 | 0.51 | 0.47 | 0.49 |
| | Ext. Set | 0.50 | 0.69 | 0.60 | 0.64 |

Table 6.21: Results for PITTS external set.

- **External Set: UBC**

| | Training set | Test set | External set |
|---|---|---|---|
| **HC** | 402 | 173 | 20 |
| **BD** | 352 | 151 | 55 |

Table 6.22: Dataset Split for UBC external set.

| Pipeline | Set | AUC | Precision | Recall | F1-score |
|---|---|---|---|---|---|
| **1** | **Test Set** | 0.61 | 0.54 | 0.48 | 0.51 |
| | **Ext. Set** | 0.47 | 0.70 | 0.25 | 0.37 |
| **5** | **Test Set** | 0.51 | 0.49 | 0.43 | 0.46 |

| | | | | |
|---|---|---|---|---|
| **Ext. Set** | 0.42 | 0.61 | 0.49 | 0.55 |

Table 6.23: Results for UBC external set.

- **Average Performance for LOSO-CV**

| Pipeline | Ext. Set AUC | Test set AUC |
|---|---|---|
| **1** | 0.5050±0.0918 | 0.6300±0.0158 |
| **5** | 0.5125±0.0621 | 0.5350± 0.0150 |

Table 6.24: LOSO-CV Results.

- **Individual site-level analysis**

| Center | HC Training set | BD Training set | HC Test Set | BD Test set | Pipeline 1 | |
|---|---|---|---|---|---|---|
| | | | | | AUC-ROC | F1-score |
| 1 | 61 | 15 | 32 | 5 | 0.9125 | 0.5000 |
| 2 | 166 | 174 | 84 | 83 | 0.4674 | 0.5497 |
| 3 | 80 | 16 | 31 | 7 | 0.6359 | 0.3333 |
| 4 | 20 | 7 | 6 | 5 | 0.3667 | 0.2222 |
| 5 | 50 | 96 | 17 | 37 | 0.6701 | 0.7222 |
| 6 | 20 | 43 | 8 | 15 | 0.2500 | 0.6897 |
| 7 | 26 | 40 | 4 | 15 | 0.7407 | 0.4500 |

*The f1-score is not based on any probability decision threshold. It is calculated on the native SVM outputs, thus, inconsistencies with the AUC-ROC are expected.

Table 6.25: Site-level analysis results.

### 6.6.1 Discussion

- **Pipeline 1**

The results on the external set, training the SVM classifier with a LOSO-CV (out-of-4), is on average a chance-level AUC, for both pipelines. The results on the test set are better when data was not corrected, following pipeline 1, but leading to worst results generalizing to the external set, as seen before in the normative approach.

- **Pipeline 5**

For CV-Harmonization pipeline 5, the average result on the four external sets is an AUC of 0.51 and 0.53 in the test set. The results were better on the test set for pipeline 1 but the drop between test set and external set performance is also higher. Pipeline 5 in the SVM model obtained results in the test set that are similar to the normative model approach.

- **Enigma Study Comparision**

The ENIGMA study[17], for which we have set ourselves to compare results using the same SVM model, reported an AUC for LOSO-CV framework of 60.92, an AUC of 71.49 for Aggregated subject-level analysis, and an AUC ranging from 40.00 to 71.00 for site-level analysis. The latter did not harmonize data but rather modeled for site differences. To compare our results, we consider pipeline 1, where site and biological confounders are not corrected for. For our SVM model, the site-level analysis AUC results range from 25.00 to 91.00, while the LOSO-CV analysis results in an AUC=63.00 in the test set and AUC=53.00 in the external set. The results are worse than those achieved by the ENIGMA study.

The key methodological differences between the analysis are mainly in the CV method used and data numerosity. They have performed Synthetic Minority Oversampling Technique with Tomek link in all analyses and used k-fold CV for which in all analyses the validation fold would have 3(±1) cases. The dataset they used for the several ML analysis consisted of data from 13 sites, ranging from 30 to 749 BD cases, due to using the oversampling technique. The data used in our analysis was gathered from 7 sites, ranging from 7 to 174 BD cases, and we did not model for dataset imbalanced. They have performed k-fold CV, setting k such that all analyses had 6 samples (3 HC and 3 BD) as test set, while in our analysis we have used the Holdout method leaving 30% of data for the test set. Thus, we can say the analysis is not fully comparable, and our results might be hindered due to training the SVM model on fewer data, both on the site-level and aggregated analysis, and not using a k-fold CV framework leads to test set results being statistically weaker as model generalizability estimates.

## 6.7 Comparing the Normative Approach, SVM, and Clinical state-of-art diagnostic performance

- **Pipeline 1: Normative model vs SVM model**

In both Normative and SVM models, for pipeline 1 – no data correction (A) – the AUC-ROC curve results were better on the test set, but not on the external set. We conclude that when confounding variables such as site and biological covariates are

not modeled, the generalization capability of an ML model to data from new sites is hampered. The latter finding, although expected, also implies that an internal validation framework is insufficient to estimate models' generalization error, in a setting where usage of that same model is projected to include data from new centers, thus for multi-site studies.

- **Pipeline 5: Normative model vs SVM model**

Comparing pipeline 5, between the normative model and SVM model, with PITTS set as the external set, we conclude the AUC results on the test set are comparable. While the SVM model achieved an AUC of 0.55 in the test set, the normative model achieved an AUC of 0.51, slightly lower. However, in the external set, the SVM model achieved an AUC of 0.50 while the normative model achieved an AUC of 0.58. The advantage of the SVM model approach is that of allowing direct classification, while in the normative approach one has to classify a positive case through a unique set of features. The normative approach classification achieved an AUC of 0.61 in the external set, however, a LOSO-CV would be necessary to improve confidence in the later result. Further limitations on these results will be discussed later in chapter 7.

- **Increment Utility: Clinical state-of-art**

The classification result achieved in the external set is an AUC=0.61. According to [62] the BD misdiagnosis rate is around 69% - a false-negative rate - thus the average psychiatrist's sensitivity in diagnosing BD is estimated at 31%. Recall or sensitivity tells us how many positive cases are spotted from the total amount of positive cases. Clinically, it can be more important to have high confidence in a positive diagnosis, than to risk a wrong BD diagnosis. The burden of misdiagnosis for the patient, currently means that more than one-third remain misdiagnosed for 10 years or more. To evaluate the increment utility an ML model yields regarding BD diagnosis, the accuracy metric is not enough. It is important to achieve good precision, i.e., the positive cases a model find should very likely be corrected, i.e., achieving a low false-positive rate. The recall-precision curve performed for the subset of features on the External Set show two possible optimal thresholds, with the second one yielding a recall/sensitivity of 40% - comparable to the clinical state-of-art- and a precision of 0.85, metric for which a clinical comparable metric was not found in the literature but is expected to be high.

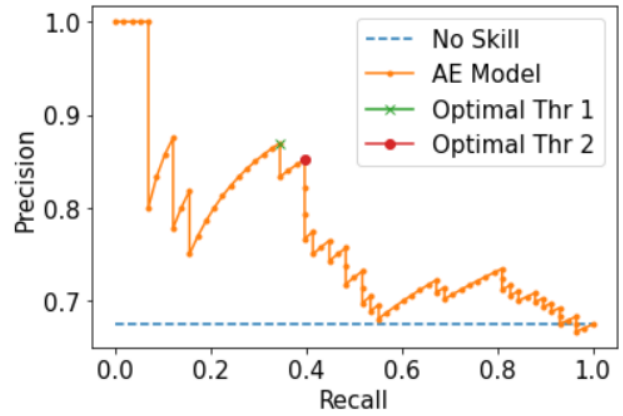



Figure 6.18 Precision-Recall Curve: Feature subset on External Set for Pipeline 5.

# 7. Conclusions

The goals of this work were both focused on the evaluation of a Normative approach starting from ROI-based volume and cortical thickness data to classifying HC and BD disorder patients and on the adaptation and comparison of different harmonization processing pipelines integrated into an ML analysis. From chapter 4. Aim of the Work, we report here what we have proposed ourselves to analyze:

1. Produce a successful normative model to reconstruct healthy brain features;
2. Discriminate BD against HC using the normative model;
3. Extract brain-feature abnormalities characterizing patients within the heterogeneous BD spectrum;
4. Assess if BD can be classified by using the subset of unique relevant brain features (aim 3) instead of all brain features;
5. Assess any improvement in BD classification obtained using the normative-based approach with respect to the classical SVM classifier;
6. Identify the optimal site-effect removal pipeline to be integrated in a ML analysis by comparing different multisite harmonization pipelines combined with biological covariates correction;

In this chapter conclusions on the several aforementioned points will be drawn and the limitations of this work and solutions to those limitations will be discussed.

## 7.1 Conclusions

In this section, we will report the conclusions we can extract from the harmonization methods pipelines, the normative model performance, and the SVM model performance.

**1) Harmonization methods**

Regarding the different harmonization methods, we conclude that harmonizing data with option B (used in pipeline 5) is as effective as when harmonizing the whole dataset together and that the previous option is more rigorous because is performed within the CV framework. Not only, the option that led to better model

generalization to the external set was also harmonization integrated into the external validation framework, option C also used in pipeline 5.

A novelty has been introduced by this work is the harmonization of neuroimaging data on an external set, using the *reference_batch* option in the ComBat function, as well as, data harmonization within a k-fold CV framework. To the best of our knowledge, this type of discussion and practice has not been presented in the literature for neuroimaging data, within the scope of ML analysis.

**2) AE-based Normative Model Generalization**

Considering all the trials that were made, the best external set generalization results were obtained by employing processing pipeline 5 and processing pipeline 2 when the external set was OSR center. The former results reveal the importance of training the model with a lot of data but somehow homogenous or devoided of confounding variables. When the model is not presented with confounding effects it won't learn them and it will be able to generalize better. In the case of our dataset, composed of 7 centers, only one of them showed marked site effects, center OSR. When this center was not included in the training set and was used in the external validation procedure, although it yield a much worse reconstruction error, the model was still able to detect the main structural differences between HC and BD.

**Conclusions from 1) + 2):** Having data that is clear of noise and confounding effects might yield models with better performance, which are domain-relevant and generalizable. However, is it quite easy for a model to have good results with the CV framework and these internal validation techniques don't allow to assess if the model is domain-relevant. By testing in independent and external sources of data and having consistently replicable good results we can have more confidence the model did not overfit to confounding information represented in the training samples [63].

**3) Normative Approach**

The normative approach results show that the model was not successful in discriminating BD patients from HC subjects, even though was quite successful in reconstructing HC brain features. The problem posed is that the model was also successful in reconstructing BD brain features. This either points to the fact that BD patients and HC brain regional features are too similar, resulting in difficult discrimination through reconstruction error scores, or that minimizing the network reconstruction error in the HC does not maximize the network discriminative ability.

**4) Neuroanatomical deviating features generalization**

The model is used to identify a neuroanatomical deviating pattern in the BD group. This subset of features was then used to classify BD in the external set. We conclude the features were generalizable to the external set, yielding an improved AUC from 0.58 in the test set to 0.61 in the external set, being the achieved classification of BD the AUC=0.61. The latter represents a promising result from our proposed methodological approach.

**5) SVM model**

The SVM model yield low performances in the test set, comparable to the one of the normative approach. In the LOSO-CV, the ENIGMA study achieved higher generalizability performances compared to our SVM model, regardless of not having harmonized data. As discussed in section 6.6, we argue that increasing data numerosity could improve our analysis.

Finally, the key achievements of this work are:

- ✓ A harmonization processing pipeline proposal, to be integrated both in internal and external validation frameworks of ML analysis.
- ✓ A normative approach pipeline proposal going from discrimination to classification of disorder patients.
- ✓ A multivariate feature extraction normative model.

## 7.2 Limitations and Future Developments

Limitations of the presented work concern the neuroimaging features, the sample clinical heterogeneity, CV framework used, the dataset size, the psychiatry disorder diagnostic specificity and possibly the confounder adjustment method. In this section, those limitations will be discussed and possible solutions and future developments will be proposed to tackle them.

- ROI-based features

The data that was used was feature engineering in an automatic ROIs feature extraction. Ideally, using voxel-based data would be better and would possibly allow achieving higher discriminative performances. When ROIs are extracted a lot of information is lost, although one gains by reducing data dimensionality, in time and memory resources. Not only, it would be preferred to input raw data to a Deep Learning model, as it works very well in this setting. We think that voxel-based features would increase the network capability of learning from normality samples, encoding a more powerful normality feature space. Besides, it would possibly lead to a better discriminative performance in BD patients, since the sMRI differences are subtle and difficult to spot.

Another future development would be to include in this study WM brain features together with the ones already used or on their own. It could be interesting to analyze the performance of the model when including these features, since as discussed in section 1.2, WM tracks in the LN are also compromised in BD disorder patients.

- CV Framework

The CV framework that was used in this work is also a limitation. Ideally, a nested 10-fold CV should have been performed, instead, data was split using the Holdout Method, and the training set underwent into a 10-fold CV for model optimization. In a nested 10-fold CV the number of iterations would increase 10 times because each complete hyperparameter search step would have to be repeated for 10 different training sets. Adding to this, because we are randomly selecting 15% of the BD patient set to be used as a BD test set, to have a more balanced test set, this split should be repeated for 10 different random partitions. Thus for each test set, 10 different BD test sets should be used and performances averaged. The latter should be performed to model BD patient heterogeneity. The alternative is to use the entire BD set as a test set but this would lead to a very imbalanced test set, although this fact could be dealt with by choosing the appropriate evaluation metrics, such as precision-recall curves, f1-scores. Besides, a LOSO-CV would also be recommended, iteratively considering each site as the external set. The current model optimization time resources were 45 minutes to complete one parameter combination evaluation (composed of 10 iterations – 10-folds). If we would have 10 folds for training/test set split, one hyperparameter combination evaluation would take seven hours and a half. Considering all 7 centers as external sets one at a time would mean repeating the later procedure 7 times, thus one hyperparameter combination trial would take around 2 days to evaluate. Since the parameter grid is usually composed of 50 or more different combinations, it would increase immensely the time and computer resources needed to train the Deep Learning model.

- Samples' clinical heterogeneity

The clinical heterogeneity of the sample should be considered in the future. For the BD group, clinical dimensions were not included at all in the covariates, and this could affect the reconstruction error results in this group. Some important factors, such as medication, were not modeled for. There are known effects of several types of medication-related to structural brain alterations which do not correlate with BD. For a new undiagnosed individual, which consequently does not take BD-related medication, those alterations would not be presented. Thus, a classification model cannot use that information to learn how to accurately classify a BD subject, during

model design, as it is not generalizable to a new BD case. These considerations should be taken into account in the future.

- Sample size and Anomaly Samples Variety

Another limitation of this work is potentially the number of samples in the training set as well as the test set. Although the training set is big containing 519 HC subjects, a deep learning model would benefit from a larger dataset, and because we wanted to keep the maximum amount of data for the training set, only 10% were used to test the model, thus reinforcing more the necessity of performing a nested k-fold CV. A future development to be considered is to integrate an oversampling technique to increase numerosity and balance the dataset. There are many oversampling techniques, from re-sampling techniques to the artificial generation of new samples. Besides sample size, an approach such as the normative one requires to be tested on more than one psychiatry disorder group. To evaluate the discriminative performance of the network, it should be tested besides BD patients. It might be the case that the model is good at discriminating against other psychiatric disorders. Also, to evaluate whether the BD abnormal brain features are uniquely predictors of BD they should be tested against other psychiatry disorders whose diagnoses or clinical biomarkers usually overlap, such as MDD, UD, and Schizophrenia. Only by performing the latter points, we can validate the discriminative performance and the classification ability of the Normative Model. Several public neuroimaging databases include both HC and cases of different brain disorders. Those could be used to increase the amount of data but also data variability and heterogeneity of the dataset. The Human Connectome Project (HCP) database [64] which contains HC, the Autism Brain Imaging Data Exchange (ABIDE) initiative [65], and the Northwestern University Schizophrenia Data and Software Tool (NUSDAST) [66] are three examples of public databases that could be included in further studies.

- Anomaly Detection with AE models

The issue of improving anomaly detection capabilities in autoencoders is discussed in S.Wang [67], where the author state that the assumption that an AE learning on training data will produce higher reconstruction errors for anomalous samples does not hold in practice. Minimizing the reconstruction error does not mean maximizing anomaly detection capabilities. This is also confirmed by the fact that the model which yields better generalization performance in the external set, reported in Appendix A, is not the model which yields the lower reconstruction error, deemed the best model in the results. They argue that for training data contaminated with anomalous data the AE ends up generalizing so well for the training set that it can reconstruct well both normal and anomalous samples. The latter argument is especially severe for unsupervised anomaly detection, where the data label is

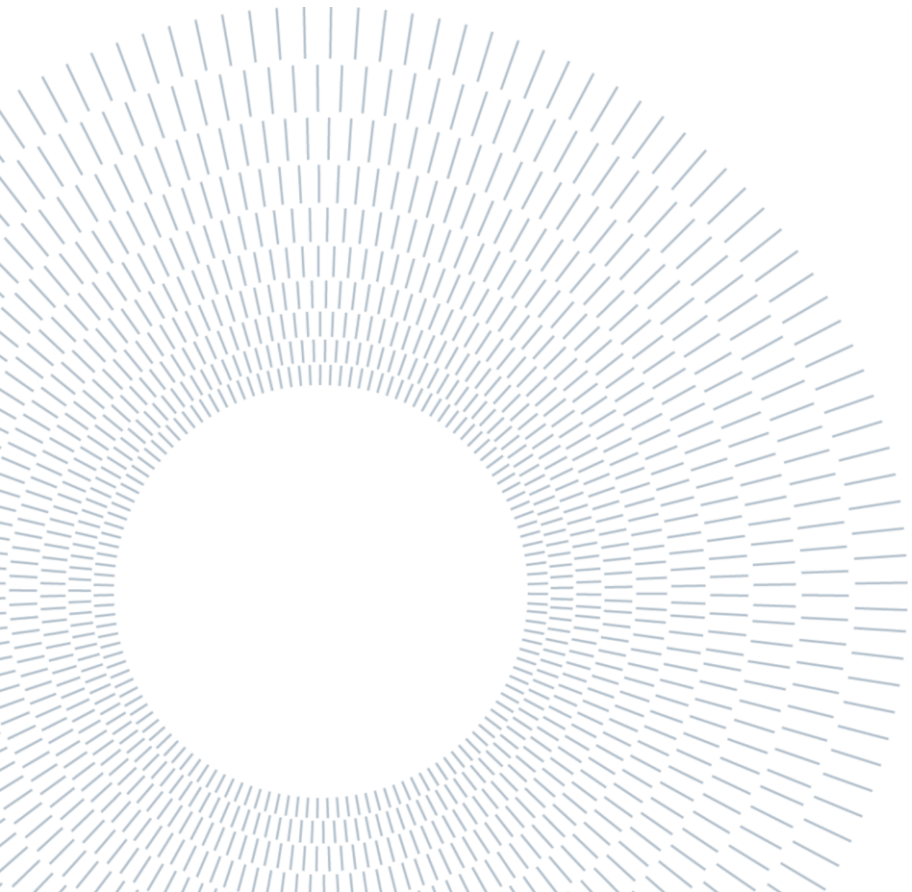
unknown and corrupted with anomaly events. Nevertheless, the issue is valid for both supervised anomaly detection – use of both labels to train a model-, and semi-supervised anomaly detection – using only normal data to train a model. Our case is not exactly training data contaminated with anomalous BD samples but perhaps correlated with the inter-variability of HC subjects' brain features. The hypothesis is that HC outliers are being taken into account as a normative sample when they should not be modeled in this way. Then, these outliers HC, which do not represent BD, contaminate the training set. During training, the AE is force to learn how to reconstruct HC in a too wide-variability range leading to the model generalizing too well, thus generalizing as well to BD patient, given the subtle differences it exists between the 2 groups at the neuroanatomy level. This leads to the reconstruction error between the two groups being less separable, which is the hypothesis we propose for the fact that our model reconstructs very well both HC and BD subjects. The problem that is posed with the latter argument is that defining an outlier in the HC population is rather problematic. HC that are more deviating from the normative range do not have a disorder necessarily. Eliminating all together these subjects would also bias the model, possibly leading it to label "anomalous" HC as disordered, which is unwanted. The authors of the aforementioned study proposed an Improved AE for Anomaly Detection (IAEAD) by proposing incorporating an SVDD loss into the AE instead of using the reconstruction loss as a strategy to spot anomalies. They argue that minimizing the reconstruction loss does not necessarily mean maximizing anomaly detection performance and with the SVDD loss the volume of a hypersphere that encloses the network representation of data can be minimized, leading to detecting anomalies based on the distance to the centroid of this hypersphere. Thus, their method detects anomalies in the feature space. There are other methods proposed in the literature that seem to improve separability between normal and anomaly data, such as including information that aids separability between samples in the training set or including sample labels for the same reason. Possibly, the Normative model could be modified to a supervised framework including labels to help improve the separability of data. In our case especially, since there is access to a lot of label data from both groups, it could be an advantage. An improvement to the loss function as proposed in [68] could also do the trick. Another possible method to be investigated is to analyze HC outliers before the training of the normative model. This could be done by employing an autoencoder method as proposed in [69] with an application example in [70] on T1-weighted MRI data. Concluding, there are several proposed methods to improve a AE model for anomaly detection.

- Harmonization with ComBat

Future development of this work, regarding the harmonization option, would be to use M-ComBat [55] variant to harmonize all data, which we have identified in this work as Ref_ComBat. This means choosing one center as the reference batch, perhaps the one with better quality or more updated acquisition protocols or sequences, in order to bring all the other centers, one by one, to the reference-batch level. With the latter approach, there would not be differences in the harmonization technique between test sets and external sets. It could be an advantage to eliminate this source of variation in the processing pipeline. The test set would be composed of several external sets, thus leading to evaluate the model solely in an external validation framework, which is by itself already more informative than using internal validation frameworks.

- Regressing-out confounding effects

For the biological covariates, although regressing out their effects is a common and standard procedure in literature, the risk of misspecification is a major drawback. Usually, linear regression is used due to its simplistic approach and easiness of application, however, the linear relationship assumption is not proven, and when it does not hold, data might be still confounded by some unwanted signals for which higher-order models would be necessary. In W. Pinaya et al. [57], the effects of the biological covariates, age, and sex were modeled within the AE model. The authors design a semi-supervised network architecture that would learn to reconstruct HC brain-feature data unsupervised and parallelly learn to predict age and sex for each subject, in a supervised framework. The AE loss function was modified to include two objectives regarding age and sex prediction, to guide the model to learn this information from neuroimaging data, and an extra term called XCov that guides the training to disentangle the age and sex signals encoded in the data from other latent features. The authors argue that with this framework the latent variables encoded by the AE are devoided of the biological covariates information. A detailed description of this semi-supervised framework is given by B.Cheung et al. [71]. The latter is an interesting method to deal with confounding effects without making assumptions about their relationship with features of interest and could be further investigated.



# Bibliography

[1] E. Severus, N. Schaaff, and H. J. Möller, "State of the Art: Treatment of Bipolar Disorders," *CNS Neurosci. Ther.*, vol. 18, no. 3, pp. 214–218, 2012, doi: 10.1111/j.1755-5949.2011.00258.x.

[2] J. M. Tamayo *et al.*, "Bipolar Spectrum Disorder: Origins and State of the Art," *Curr. Psychiatry Rev.*, vol. 9, no. 1, pp. 3–20, 2013, doi: 10.2174/1573400511309010003.

[3] J. R. C. De Almeida and M. L. Phillips, "Distinguishing between unipolar depression and bipolar depression: Current and future clinical and neuroimaging perspectives," *Biological Psychiatry*, vol. 73, no. 2. 2013, doi: 10.1016/j.biopsych.2012.06.010.

[4] E. Sigitova, Z. Fišar, J. Hroudová, T. Cikánková, and J. Raboch, "Biological hypotheses and biomarkers of bipolar disorder," *Psychiatry Clin. Neurosci.*, vol. 71, no. 2, pp. 77–103, 2017, doi: 10.1111/pcn.12476.

[5] P. J. Harrison, J. R. Geddes, and E. M. Tunbridge, "The Emerging Neurobiology of Bipolar Disorder," *Trends Neurosci.*, vol. 41, no. 1, pp. 18–30, 2018, doi: 10.1016/j.tins.2017.10.006.

[6] G. Delvecchio, E. Maggioni, L. Squarcina, and P. Brambilla, "Brain Network Dysfunction in Bipolar Disorder: Evidence from Structural and Functional MRI Studies," *Brain Netw. Dysfunct. Neuropsychiatr. Illn.*, pp. 313–332, 2021, doi: 10.1007/978-3-030-59797-9_15.

[7] G. Delvecchio *et al.*, "A Critical Review on Structural Neuroimaging Studies in BD: a Transdiagnostic Perspective from Psychosis to Fronto-Temporal Dementia," *Curr. Behav. Neurosci. Reports 2020 72*, vol. 7, no. 2, pp. 86–95, Mar. 2020, doi: 10.1007/S40473-020-00204-7.

[8] P. Magioncalda and M. Martino, "A unified model of the pathophysiology of bipolar disorder," *Mol. Psychiatry*, vol. 27, no. 1, pp. 202–211, 2022, doi: 10.1038/s41380-021-01091-4.

[9] D. P. Hibar *et al.*, "Cortical abnormalities in bipolar disorder: An MRI analysis of 6503 individuals from the ENIGMA Bipolar Disorder Working Group," *Mol. Psychiatry*, vol. 23, no. 4, pp. 932–942, 2018, doi: 10.1038/mp.2017.73.

[10] V. P. B. Grover, J. M. Tognarelli, M. M. E. Crossey, I. J. Cox, S. D. Taylor-Robinson, and M. J. W. McPhail, "Magnetic Resonance Imaging: Principles and Techniques: Lessons for Clinicians," *J. Clin. Exp. Hepatol.*, vol. 5, no. 3, pp. 246–255, 2015, doi: 10.1016/j.jceh.2015.08.001.

[11] G. Katti, S. Arshiya Ara, and A. Shireen, "Magnetic Resonance Imaging (MRI) – A Review," *Int. J. Dent. Clin.*, vol. 3, no. 1, pp. 65–70, 2011.

[12] K. Broadhouse, "The Physics of MRI and How We Use It to Reveal the Mysteries of the Mind.," *Front. Young Minds*, 2019, doi: 10.3389/frym.2019.00023.

[13] C. R. K. Ching *et al.*, "What we learn about bipolar disorder from large-scale neuroimaging: Findings and future directions from the ENIGMA Bipolar Disorder Working Group," *Hum. Brain Mapp.*, no. March, pp. 1–27, 2020, doi: 10.1002/hbm.25098.

[14] J. B. Ding and K. Hu, "Structural MRI Brain Alterations in Borderline Personality Disorder and Bipolar Disorder," *Cureus*, vol. 13, no. 7, Jul. 2021, doi: 10.7759/CUREUS.16425.

[15] Z. Jan *et al.*, "The Role of Machine Learning in Diagnosing Bipolar Disorder: Scoping Review," *J. Med. Internet Res.*, vol. 23, no. 11, p. e29749, Nov. 2021, doi: 10.2196/29749.

[16] F. Colombo *et al.*, "Machine learning approaches for prediction of bipolar disorder based on biological, clinical and neuropsychological markers: A systematic review and meta-analysis," *Neurosci. Biobehav. Rev.*, vol. 135, p. 104552, Apr. 2022, doi: 10.1016/J.NEUBIOREV.2022.104552.

[17] A. Nunes *et al.*, "Using structural MRI to identify bipolar disorders – 13 site machine learning study in 3020 individuals from the ENIGMA Bipolar Disorders Working Group," *Mol. Psychiatry*, vol. 25, no. 9, pp. 2130–2143, 2020, doi: 10.1038/s41380-018-0228-9.

[18] L. A. Claude, J. Houenou, E. Duchesnay, and P. Favre, "Will machine learning applied to neuroimaging in bipolar disorder help the clinician? A critical review and methodological suggestions," *Bipolar Disord.*, vol. 22, no. 4, pp. 334–355, 2020, doi: 10.1111/bdi.12895.

[19] I. Goodfellow, Y. Bengio, and A. Courville, *Deep Learning*. MIT Press, 2016.

[20] C. Vercellis, *Business Intelligence: Data Mining and Optimization for Decision Making*. 2009.

[21] G. Koppe, A. Meyer-Lindenberg, and D. Durstewitz, "Deep learning for small and big data in psychiatry," *Neuropsychopharmacology*, vol. 46, no. 1, pp. 176–190, 2021, doi: 10.1038/s41386-020-0767-z.

[22] J. Shawe-Taylor and S. Sun, "A review of optimization methodologies in support vector machines," *Neurocomputing*, vol. 74, no. 17, pp. 3609–3618, 2011, doi: 10.1016/j.neucom.2011.06.026.

[23] P. Kassraian-Fard, C. Matthis, J. H. Balsters, M. H. Maathuis, and N. Wenderoth, "Promises, pitfalls, and basic guidelines for applying machine learning classifiers to psychiatric imaging data, with autism as an example," *Front. Psychiatry*, vol. 7, no. DEC, 2016, doi: 10.3389/fpsyt.2016.00177.

[24] C. Su, Z. Xu, J. Pathak, and F. Wang, "Deep learning in mental health outcome research: a scoping review," *Transl. Psychiatry*, vol. 10, no. 1, 2020, doi: 10.1038/s41398-020-0780-3.

[25] M. A. Nielsen, *Neural Networks and Deep Learning*. Determination Press, 2015.

[26] C. C. Aggarwal, *Neural Networks and Deep Learning: A Textbook*. Cham: Springer International Publishing AG, 2018.

[27] G. Klambauer, T. Unterthiner, A. Mayr, and S. Hochreiter, "Self-normalizing neural networks," *Adv. Neural Inf. Process. Syst.*, vol. 2017-Decem, pp. 972–981, 2017.

[28] D. P. Kingma and J. L. Ba, "Adam: A method for stochastic optimization," *3rd Int. Conf. Learn. Represent. ICLR 2015 - Conf. Track Proc.*, pp. 1–15, 2015.

[29] M. Cearns, T. Hahn, and B. T. Baune, "Recommendations and future directions for supervised machine learning in psychiatry," *Transl. Psychiatry*, vol. 9, no. 1, 2019, doi: 10.1038/s41398-019-0607-2.

[30] W. Penny, K. Friston, J. Ashburner, S. Kiebel, and T. Nichols, "Statistical Parametric Mapping: The Analysis of Functional Brain Images," *Stat. Parametr. Mapp. Anal. Funct. Brain Images*, 2007, doi: 10.1016/B978-0-12-372560-8.X5000-1.

[31] C. Gase, R. Dahnk, K. K, and L. E, "CAT- A Computational Anatomy Toolbox for the Analysis of Structural MRI Data.," *Neuroimage, Rev.*

[32] J. A. Andrea Mechelli*, Cathy J. Price, Karl J. Friston, "Voxel-Based

Morphometry," *arXiv*, pp. 1–9, 2017, doi: 10.1142/s2424942417500086.

[33] N. Tzourio-Mazoyer, B. Landeau, D. Papathanassiou, F. Crivello, O. Etard, and N. Delcroix, "Automated anatomical labelling of activations in spm using a macroscopic anatomical parcellation of the MNI MRI single subject brain.," *Neuroimage*, vol. 15(1), pp. 273–289, 2002, doi: 10.1006/nimg.2001.0978.

[34] V. Tavares, D. Prata, and H. A. Ferreira, "Comparing SPM12 and CAT12 segmentation pipelines: a brain tissue volume-based age and Alzheimer's disease study," *J. Neurosci. Methods*, vol. 334, no. December 2019, p. 108565, 2020, doi: 10.1016/j.jneumeth.2019.108565.

[35] J. Ashburner and K. J. Friston, "Unified segmentation," *Neuroimage*, vol. 26, no. 3, pp. 839–851, 2005, doi: 10.1016/j.neuroimage.2005.02.018.

[36] J. Radua, E. J. Canales-Rodríguez, E. Pomarol-Clotet, and R. Salvador, "Validity of modulation and optimal settings for advanced voxel-based morphometry," *Neuroimage*, vol. 86, pp. 81–90, 2014, doi: 10.1016/j.neuroimage.2013.07.084.

[37] P. Vizza, G. Tradigo, D. Messina, G. L. Cascini, and P. Veltri, "Methodologies for the analysis and classification of PET neuroimages," *Netw. Model. Anal. Heal. Informatics Bioinforma.*, vol. 2, no. 4, pp. 191–208, 2013, doi: 10.1007/s13721-013-0035-9.

[38] J. Ashburner, "A fast diffeomorphic image registration algorithm," *Neuroimage*, vol. 38, no. 1, pp. 95–113, 2007, doi: 10.1016/j.neuroimage.2007.07.007.

[39] C. H. Salmond, J. Ashburner, F. Vargha-Khadem, A. Connelly, D. G. Gadian, and K. J. Friston, "Distributional Assumptions in Voxel-Based Morphometry," *Neuroimage*, vol. 17, no. 2, pp. 1027–1030, 2002, doi: 10.1006/nimg.2002.1153.

[40] J. C. Mazziotta, A. W. Toga, A. Evans, P. Fox, and J. Lancaster, "A probabilistic atlas of the human brain: theory and rationale for its development. The International Consortium for Brain Mapping (ICBM).," *NeuroImage*, vol. 2, no. 2. pp. 89–101, 1995, [Online]. Available: http://www.ncbi.nlm.nih.gov/pubmed/9343592.

[41] S. Tullo, G. A. Devenyi, R. Patel, M. T. M. Park, D. L. Collins, and M. M. Chakravarty, "Warping an atlas derived from serial histology to 5 high-resolution MRIs," *Sci. data*, vol. 5, Jun. 2018, doi: 10.1038/SDATA.2018.107.

[42] "Neuromorphometrics, Inc. | Building a Model of the Living Human Brain." http://www.neuromorphometrics.com/ (accessed Jun. 23, 2022).

[43] R. S. Desikan *et al.*, "An automated labeling system for subdividing the human cerebral cortex on MRI scans into gyral based regions of interest," *Neuroimage*, vol. 31, no. 3, pp. 968–980, Jul. 2006, doi: 10.1016/J.NEUROIMAGE.2006.01.021.

[44] K. J. Jager, C. Zoccali, A. MacLeod, and F. W. Dekker, "Confounding: What it is and how to deal with it," *Kidney Int.*, vol. 73, no. 3, pp. 256–260, 2008, doi: 10.1038/sj.ki.5002650.

[45] O. Environ and R. Mcnamee, "Regression modelling and other methods to control confounding," *Occup. Environ. Med.*, vol. 62, no. 7, pp. 500–506, Jul. 2005, doi: 10.1136/OEM.2002.001115.

[46] L. Snoek, S. Miletić, and H. S. Scholte, "How to control for confounds in decoding analyses of neuroimaging data," *Neuroimage*, vol. 184, no. September 2018, pp. 741–760, 2019, doi: 10.1016/j.neuroimage.2018.09.074.

[47] A. Rao, J. M. Monteiro, and J. Mourao-Miranda, "Predictive modelling using neuroimaging data in the presence of confounds," *Neuroimage*, vol. 150, no. January, pp. 23–49, 2017, doi: 10.1016/j.neuroimage.2017.01.066.

[48] C. Wachinger, A. Rieckmann, and S. Pölsterl, "Detect and correct bias in multi-site neuroimaging datasets," *Med. Image Anal.*, vol. 67, 2021, doi: 10.1016/j.media.2020.101879.

[49] B. Glocker, R. Robinson, D. C. Castro, Q. Dou, and E. Konukoglu, "Machine Learning with Multi-Site Imaging Data: An Empirical Study on the Impact of Scanner Effects," pp. 1–5, 2019, [Online]. Available: http://arxiv.org/abs/1910.04597.

[50] R. McNamee, "Regression modelling and other methods to control confounding," *Occup. Environ. Med.*, vol. 62, no. 7, pp. 500–506, 2005, doi: 10.1136/oem.2002.001115.

[51] B. C. Kahan, H. Rushton, T. P. Morris, and R. M. Daniel, "A comparison of methods to adjust for continuous covariates in the analysis of randomised trials," *BMC Med. Res. Methodol.*, vol. 16, no. 1, pp. 1–10, 2016, doi: 10.1186/s12874-016-0141-3.

[52] W. E. Johnson, C. Li, and A. Rabinovic, "Adjusting batch effects in microarray expression data using empirical Bayes methods," *Biostatistics*, vol. 8, no. 1, pp. 118–127, 2007, doi: 10.1093/biostatistics/kxj037.

[53] J. P. Fortin *et al.*, "Harmonization of cortical thickness measurements across scanners and sites," *Neuroimage*, vol. 167, no. June 2017, pp. 104–120, 2018, doi:

10.1016/j.neuroimage.2017.11.024.

[54] J. Radua *et al.*, "Increased power by harmonizing structural MRI site differences with the ComBat batch adjustment method in ENIGMA," *Neuroimage*, vol. 218, p. 116956, Sep. 2020, doi: 10.1016/J.NEUROIMAGE.2020.116956.

[55] C. K. Stein *et al.*, "Removing batch effects from purified plasma cell gene expression microarrays with modified ComBat," *BMC Bioinformatics*, vol. 16, no. 1, pp. 1–9, 2015, doi: 10.1186/s12859-015-0478-3.

[56] A. Behdenna, J. Haziza, C.-A. Azencott, and A. Nordor, "pyComBat, a Python tool for batch effects correction in high-throughput molecular data using empirical Bayes methods," *bioRxiv*, p. 2020.03.17.995431, Mar. 2020, doi: 10.1101/2020.03.17.995431.

[57] W. H. L. Pinaya, A. Mechelli, and J. R. Sato, "Using deep autoencoders to identify abnormal brain structural patterns in neuropsychiatric disorders: A large-scale multi-sample study," *Human Brain Mapping*, vol. 40, no. 3. pp. 944–954, 2019, doi: 10.1002/hbm.24423.

[58] DNV GL, "Clinical Decision Support Software: Regulatory Landscape in Europe," *BigMed*, 2020.

[59] K. A. Gorgens, "Structured Clinical Interview For DSM-IV (SCID-I/SCID-II)," *Encycl. Clin. Neuropsychol.*, pp. 2410–2417, 2011, doi: 10.1007/978-0-387-79948-3_2011.

[60] D. V. Sheehan *et al.*, "The Mini-International Neuropsychiatric Interview (M.I.N.I.): The development and validation of a structured diagnostic psychiatric interview for DSM-IV and ICD-10," *J. Clin. Psychiatry*, vol. 59, no. SUPPL. 20, pp. 22–33, 1998.

[61] C. Gaser, "Manual Computational Anatomy Toolbox- cat12," *Struct. Brain Mapp. Gr. Dep. Psychiatry Neurol. Univ. Jena*, pp. 1–53, 2017, [Online]. Available: http://www.neuro.uni-jena.de/cat/index.html.

[62] T. Singh and M. Rajput, "Misdiagnosis of Bipolar Disorder," *Psychiatry (Edgmont)*, vol. 3, no. 10, p. 57, Aug. 2005, Accessed: Jun. 20, 2022. [Online]. Available: /pmc/articles/PMC2945875/.

[63] S. Y. Ho, K. Phua, L. Wong, and W. W. Bin Goh, "Extensions of the External Validation for Checking Learned Model Interpretability and Generalizability," *Patterns*, vol. 1, no. 8, p. 100129, 2020, doi: 10.1016/j.patter.2020.100129.

[64] "Human Connectome Project | Mapping the human brain connectivity." http://www.humanconnectomeproject.org/ (accessed Jun. 21, 2022).

[65] "ABIDE." http://fcon_1000.projects.nitrc.org/indi/abide/ (accessed Jun. 21, 2022).

[66] "Home | SchizConnect: public neuroimaging (MRI) data." http://schizconnect.org/ (accessed Jun. 21, 2022).

[67] S. Wang, "Improved autoencoder for unsupervised anomaly detection," no. June, 2021, doi: 10.1002/int.22582.

[68] W. Xu, J. Jang-Jaccard, A. Singh, Y. Wei, and F. Sabrina, "Improving Performance of Autoencoder-Based Network Anomaly Detection on NSL-KDD Dataset," *IEEE Access*, vol. 9, pp. 140136–140146, 2021, doi: 10.1109/ACCESS.2021.3116612.

[69] S. Hawkins, H. He, G. Williams, and R. Baxter, "Outlier detection using replicator neural networks," *Lect. Notes Comput. Sci. (including Subser. Lect. Notes Artif. Intell. Lect. Notes Bioinformatics)*, vol. 2454 LNCS, pp. 170–180, 2002, doi: 10.1007/3-540-46145-0_17/COVER/.

[70] E. Ferrari *et al.*, "Dealing with confounders and outliers in classification medical studies: The Autism Spectrum Disorders case study," *Artif. Intell. Med.*, vol. 108, no. July 2019, p. 101926, 2020, doi: 10.1016/j.artmed.2020.101926.

[71] B. Cheung, J. A. Livezey, A. K. Bansal, and B. A. Olshausen, "Discovering hidden factors of variation in deep networks," *3rd Int. Conf. Learn. Represent. ICLR 2015 - Work. Track Proc.*, pp. 1–10, 2015.

# A. Appendix A

- Desikan-Killiany Atlas Cortical Parcelations

| ID | Abbreviation | ROI Name | Lobe |
|---|---|---|---|
| 2647065<br>2647065 | 'lbankssts'<br>'rbankssts' | Banks superior temporal sulcus | Temporal |
| 10511485<br>10511485 | 'lcaudalanteriorcingulate'<br>'rcaudalanteriorcingulate' | Caudal anterior-cingulate cortex | Frontal |
| 6500<br>6500 | 'lcaudalmiddlefrontal'<br>'rcaudalmiddlefrontal' | Caudal middle frontal gyrus | Frontal |
| 3294840<br>3294840 | 'lcorpuscallosum'<br>'rcorpuscallosum' | Corpus Callosum | WM |
| 6558940<br>6558940 | 'lcuneus'<br>'rcuneus' | Cuneus cortex | Occipital |
| 660700<br>660700 | 'lentorhinal'<br>'rentorhinal' | Entorhinal cortex | Temporal |
| 9231540<br>9231540 | 'lfusiform'<br>'rfusiform' | Fusiform gyrus | Temporal |
| 14433500<br>14433500 | 'linferiorparietal'<br>'rinferiorparietal' | Inferior parietal cortex | Parietal |
| 7874740<br>7874740 | 'linferiortemporal'<br>'rinferiortemporal' | Inferior temporal gyrus | Temporal |
| 9180300<br>9180300 | 'listhmuscingulate'<br>'risthmuscingulate' | Isthmus – cingulate cortex | Parietal |
| 9182740<br>9182740 | 'llateraloccipital'<br>'rlateraloccipital' | Lateral occipital cortex | Occipital |
| 3296035<br>3296035 | 'llateralorbitofrontal'<br>'rlateralorbitofrontal' | Lateral orbital frontal cortex | Frontal |
| 9211105<br>9211105 | 'llingual'<br>'rlingual' | Lingual gyrus | Occipital |
| 4924360<br>4924360 | 'lmedialorbitofrontal'<br>'rmedialorbitofrontal' | Medial orbital frontal cortex | Frontal |
| 3302560<br>3302560 | 'lmiddletemporal'<br>'rmiddletemporal' | Middle temporal gyrus | Temporal |

| | | | |
|---|---|---|---|
| 3988500<br>3988500 | 'lparahippocampal'<br>'rparahippocampal' | Parahippocampal gyrus | Temporal |
| 3988540<br>3988540 | 'lparacentral'<br>'rparacentral' | Paracentral lobule | Frontal |
| 9221340<br>9221340 | 'lparsopercularis'<br>'rparsopercularis' | Pars opercularis | Frontal |
| 3302420<br>3302420 | 'lparsorbitalis'<br>'rparsorbitalis' | Pars orbitalis | Frontal |
| 1326300<br>1326300 | 'lparstriangularis'<br>'rparstriangularis' | Pars triangularis | Frontal |
| 3957880<br>3957880 | 'lpericalcarine'<br>'rpericalcarine' | Pericalcarine cortex | Occipital |
| 1316060<br>1316060 | 'lpostcentral'<br>'rpostcentral' | Postcentral gyrus | Parietal |
| 14464220<br>14464220 | 'lposteriorcingulate'<br>'rposteriorcingulate' | Posterior-cingulate cortex | Parietal |
| 14423100<br>14423100 | 'lprecentral'<br>'rprecentral' | Precentral gyrus | Frontal |
| 11832480<br>11832480 | 'lprecuneus'<br>'rprecuneus' | Precuneus cortex | Parietal |
| 9180240<br>9180240 | 'lrostralanteriorcingulate'<br>'rrostralanteriorcingulate' | Rostral anterior cingulate cortex | Frontal |
| 8204875<br>8204875 | 'lrostralmiddlefrontal'<br>'rrostralmiddlefrontal' | Rostral middle frontal gyrus | Frontal |
| 10542100<br>10542100 | 'lsuperiorfrontal'<br>'rsuperiorfrontal' | Superior frontal gyrus | Frontal |
| 9221140<br>9221140 | 'lsuperiorparietal'<br>'rsuperiorparietal' | Superior parietal cortex | Parietal |
| 14474380<br>14474380 | 'lsuperiortemporal'<br>'rsuperiortemporal' | Superior temporal gyrus | Temporal |
| 1351760<br>1351760 | 'lsupramarginal'<br>'rsupramarginal' | Supramarginal gyrus | Parietal |
| 6553700<br>6553700 | 'lfrontalpole'<br>'rfrontalpole' | Frontal pole | Frontal |
| 11146310<br>11146310 | 'ltemporalpole'<br>'rtemporalpole' | Temporal pole | Temporal |
| 13145750<br>13145750 | 'ltransversetemporal'<br>'rtransversetemporal' | Transverse temporal cortex | Temporal |
| 2146559<br>2146559 | 'linsula'<br>'rinsula' | Insula | Insula |

- CoBra Atlas Regions for anatomical volumes estimation

| ID | Abbreviation | ROI Name |
|---|---|---|
| 0 | lStriatum | Left Striatum |
| 1 | lGloPal | Left Globus Pallidus |
| 2 | lTha | Left Thalamus |
| 3 | lAntCerebLI_II | Left Anterior Cerebellar Lobule I-II |
| 11 | lAntCerebLIII | Left Anterior Cerebellar Lobule III |
| 12 | lAntCerebLIV | Left Anterior Cerebellar Lobule IV |
| 13 | lAntCerebLV | Left Anterior Cerebellar Lobule V |
| 14 | lSupPostCerebLVI | Left Superior Posterior Cerebellar Lobule VI |
| 15 | lSupPostCerebCI | Left Superior Posterior Cerebellar Lobule Crus I |
| 16 | lSupPostCerebCII | Left Superior Posterior Cerebellar Lobule Crus II |
| 17 | lSupPostCerebLVIIB | Left Superior Posterior Cerebellar Lobule VIIB |
| 18 | lInfPostCerebLVIIIA | Left Inferior Posterior Cerebellar Lobule VIIIA |
| 19 | lInfPostCerebLVIIIB | Left Inferior Posterior Cerebellar Lobule VIIIB |
| 20 | lInfPostCerebLIX | Left Inferior Posterior Cerebellar Lobule IX |
| 21 | lInfPostCerebLX | Left Inferior Posterior Cerebellar Lobule X |
| 22 | lAntCerebWM | Left Cerebellar White Matter |
| 23 | lAmy | Left Amygdala |
| 26 | lHCA1 | Left Hippocampus CA1 |
| 31 | lSub | Left Subiculum |
| 32 | lFor | Left Fornix |
| 33 | lCA4 | Left CA4/Dentate Gyrus |
| 34 | lCA2_3 | Left CA2/CA3 |
| 35 | lStratum | Left Stratum Radiatum/Lacunosum/Moleculare |
| 36 | lFimbra | Left Fimbria |
| 37 | lMamBody | Left Mammillary body |
| 38 | lAlveus | Left Alveus |
| 39 | rStriatum | Right Striatum |
| 101 | rGloPal | Right Globus Pallidus |
| 102 | rTha | Right Thalamus |
| 103 | rAntCerebLI_II | Right Anterior Cerebellar Lobule I-II |
| 111 | rAntCerebLIII | Right Anterior Cerebellar Lobule III |
| 112 | rAntCerebLIV | Right Anterior Cerebellar Lobule IV |
| 113 | rAntCerebLV | Right Anterior Cerebellar Lobule V |
| 114 | rSupPostCerebLVI | Right Superior Posterior Cerebellar Lobule VI |
| 115 | rSupPostCerebCI | Right Superior Posterior Cerebellar Lobule Crus I |
| 116 | rSupPostCerebCII | Right Superior Posterior Cerebellar Lobule Crus II |
| 117 | rSupPostCerebLVIIB | Right Superior Posterior Cerebellar Lobule VIIB |

| | | |
|---|---|---|
| 118 | rInfPostCerebLVIIIA | Right Inferior Posterior Cerebellar Lobule VIIIA |
| 119 | rInfPostCerebLVIIIB | Right Inferior Posterior Cerebellar Lobule VIIIB |
| 120 | rInfPostCerebLIX | Right Inferior Posterior Cerebellar Lobule IX |
| 121 | rInfPostCerebLX | Right Inferior Posterior Cerebellar Lobule X |
| 122 | rAntCerebWM | Right Cerebellar White Matter |
| 123 | rAmy | Right Amygdala |
| 126 | rHCA1 | Right Hippocampus CA1 |
| 131 | rSub | Right Subiculum |
| 132 | rFor | Right Fornix |
| 133 | rCA4 | Right CA4/Dentate Gyrus |
| 134 | rCA2_3 | Right CA2/CA3 |
| 135 | rStratum | Right Stratum Radiatum/Lacunosum/Moleculare |
| 136 | rFimbra | Right Fimbria |
| 137 | rMamBody | Right Mammillary body |
| 138 | rAlveus | Right Alveus |

# Results

- **Section 6.3: Linear Regression Estimations**

1)Linear Regressions for which T-test p_value for **age** is not significant

-Cortical Thickness

| | regions | t_pvalue_age | t_pvalue_sex | F_pvalue | R_square | |
|---|---|---|---|---|---|---|
| **8** | [lentorhinal] | 0.38864 | 0.71480 | 0.66501 | -0.00218 | |
| **9** | [rentorhinal] | 0.60388 | 0.35622 | 0.59904 | -0.00180 | |
| **24** | [lmedialorbitofrontal] | 0.06379 | 0.55796 | 0.12993 | 0.00385 | |
| **28** | [lparahippocampal] | 0.42363 | 0.72837 | 0.65837 | -0.00215 | |
| **29** | [rparahippocampal] | 0.54012 | 0.45176 | 0.65456 | -0.00213 | |
| **61** | [rfrontalpole] | 0.11548 | 0.48517 | 0.25279 | 0.00139 | |
| **62** | [ltemporalpole] | 0.45692 | 0.02147 | 0.06314 | 0.00650 | |
| **63** | [rtemporalpole] | 0.66397 | 0.05887 | 0.16365 | 0.00300 | |
| **66** | [linsula] | 0.11404 | 0.35838 | 0.21776 | 0.00194 | |

-Volumetric measures

| | regions | t_pvalue_age | t_pvalue_sex | t_pvalue_TIV | F_pvalue | R_square |
|---|---|---|---|---|---|---|
| **2** | [lTha] | 0.99906 | 0.00000 | 0.00000 | 0.00000 | 0.07046 |
| **17** | [lHCA1] | 0.34310 | 0.07650 | 0.00000 | 0.00000 | 0.13363 |

| | | | | | | |
|---|---|---|---|---|---|---|
| **18** | [lSub] | 0.97010 | 0.25252 | 0.00000 | 0.00000 | 0.15618 |
| **19** | [lFor] | 0.26649 | 0.05550 | 0.16547 | 0.19866 | 0.00307 |
| **20** | [lCA4] | 0.12245 | 0.06177 | 0.00000 | 0.00000 | 0.14019 |
| **21** | [lCA2_3] | 0.37184 | 0.21419 | 0.00000 | 0.00000 | 0.06261 |
| **22** | [lStratum] | 0.17010 | 0.43103 | 0.00000 | 0.00000 | 0.17025 |
| **23** | [lFimbra] | 0.65436 | 0.35974 | 0.00543 | 0.02983 | 0.01098 |
| **24** | [lMamBody] | 0.05387 | 0.00347 | 0.00007 | 0.00021 | 0.03022 |
| **25** | [lAlveus] | 0.19918 | 0.69526 | 0.00000 | 0.00000 | 0.10157 |
| **27** | [rGloPal] | 0.17158 | 0.22536 | 0.27566 | 0.02066 | 0.01245 |
| **28** | [rTha] | 0.84167 | 0.00001 | 0.00000 | 0.00000 | 0.08415 |
| **29** | [rAntCerebLI_II] | 0.05806 | 0.62897 | 0.00011 | 0.00000 | 0.05522 |
| **42** | [rAmy] | 0.08770 | 0.31608 | 0.00000 | 0.00000 | 0.18621 |
| **43** | [rHCA1] | 0.95260 | 0.02769 | 0.00000 | 0.00000 | 0.11219 |
| **44** | [rSub] | 0.93086 | 0.05630 | 0.00000 | 0.00000 | 0.14618 |
| **45** | [rFor] | 0.71614 | 0.08329 | 0.05553 | 0.22737 | 0.00248 |
| **46** | [rCA4] | 0.35498 | 0.02973 | 0.00000 | 0.00000 | 0.16798 |
| **47** | [rCA2_3] | 0.43310 | 0.11757 | 0.00000 | 0.00000 | 0.07336 |
| **48** | [rStratum] | 0.83753 | 0.12255 | 0.00000 | 0.00000 | 0.17487 |
| **49** | [rFimbra] | 0.25257 | 0.04047 | 0.00005 | 0.00045 | 0.02733 |
| **50** | [rMamBody] | 0.14723 | 0.04740 | 0.00233 | 0.00989 | 0.01538 |
| **51** | [rAlveus] | 0.60119 | 0.11199 | 0.00000 | 0.00000 | 0.11066 |

2)Linear Regressions for which T-test p_value for **sex** is not significant

-Cortical Thickness

| | regions | t_pvalue_age | t_pvalue_sex | F_pvalue | R_square |
|---|---|---|---|---|---|
| **2** | [lcaudalanteriorcingulate] | 0.01948 | 0.75710 | 0.06506 | 0.00639 |
| **3** | [rcaudalanteriorcingulate] | 0.02545 | 0.13823 | 0.03859 | 0.00831 |
| **4** | [lcaudalmiddlefrontal] | 0.00000 | 0.07256 | 0.00000 | 0.06140 |
| **6** | [lcuneus] | 0.00001 | 0.98040 | 0.00004 | 0.03352 |
| **7** | [rcuneus] | 0.00000 | 0.50837 | 0.00000 | 0.06218 |
| **8** | [lentorhinal] | 0.38864 | 0.71480 | 0.66501 | -0.00218 |
| **9** | [rentorhinal] | 0.60388 | 0.35622 | 0.59904 | -0.00180 |

| | | | | | |
|---|---|---|---|---|---|
| **10** | [lfusiform] | 0.00280 | 0.35902 | 0.00969 | 0.01336 |
| **14** | [linferiortemporal] | 0.00331 | 0.59707 | 0.00917 | 0.01356 |
| **15** | [rinferiortemporal] | 0.00013 | 0.21491 | 0.00047 | 0.02437 |
| **17** | [risthmuscingulate] | 0.00096 | 0.55694 | 0.00417 | 0.01643 |
| **19** | [rlateraloccipital] | 0.01335 | 0.07201 | 0.01476 | 0.01183 |
| **21** | [rlateralorbitofrontal] | 0.00059 | 0.10432 | 0.00128 | 0.02073 |
| **22** | [llingual] | 0.00004 | 0.07153 | 0.00009 | 0.03048 |
| **23** | [rlingual] | 0.00000 | 0.56971 | 0.00000 | 0.04354 |
| **24** | [lmedialorbitofrontal] | 0.06379 | 0.55796 | 0.12993 | 0.00385 |
| **25** | [rmedialorbitofrontal] | 0.00009 | 0.39739 | 0.00021 | 0.02730 |
| **26** | [lmiddletemporal] | 0.00001 | 0.10324 | 0.00004 | 0.03323 |
| **27** | [rmiddletemporal] | 0.00000 | 0.18889 | 0.00000 | 0.04624 |
| **28** | [lparahippocampal] | 0.42363 | 0.72837 | 0.65837 | -0.00215 |
| **29** | [rparahippocampal] | 0.54012 | 0.45176 | 0.65456 | -0.00213 |
| **30** | [lparacentral] | 0.00002 | 0.06781 | 0.00006 | 0.03194 |
| **31** | [rparacentral] | 0.00000 | 0.15706 | 0.00000 | 0.07941 |
| **32** | [lparsopercularis] | 0.00000 | 0.07348 | 0.00000 | 0.07296 |
| **33** | [rparsopercularis] | 0.00000 | 0.17624 | 0.00000 | 0.05798 |
| **36** | [lparstriangularis] | 0.00000 | 0.29262 | 0.00000 | 0.05883 |
| **37** | [rparstriangularis] | 0.00000 | 0.08003 | 0.00000 | 0.06437 |
| **38** | [lpericalcarine] | 0.00000 | 0.08511 | 0.00001 | 0.03696 |
| **39** | [rpericalcarine] | 0.01474 | 0.08773 | 0.01832 | 0.01103 |
| **40** | [lpostcentral] | 0.00000 | 0.06390 | 0.00000 | 0.07450 |
| **41** | [rpostcentral] | 0.00000 | 0.08219 | 0.00000 | 0.07470 |
| **43** | [rposteriorcingulate] | 0.00002 | 0.09360 | 0.00005 | 0.03266 |
| **44** | [lprecentral] | 0.00000 | 0.11447 | 0.00000 | 0.05583 |
| **46** | [lprecuneus] | 0.00000 | 0.29400 | 0.00000 | 0.05444 |
| **47** | [rprecuneus] | 0.00000 | 0.45401 | 0.00000 | 0.08267 |
| **48** | [lrostralanteriorcingulate] | 0.00021 | 0.82060 | 0.00084 | 0.02223 |
| **49** | [rrostralanteriorcingulate] | 0.01985 | 0.85367 | 0.05976 | 0.00670 |
| **50** | [lrostralmiddlefrontal] | 0.00000 | 0.10887 | 0.00000 | 0.07024 |

| | | | | | |
|---|---|---|---|---|---|
| **51** | [rrostralmiddlefrontal] | 0.00000 | 0.25287 | 0.00000 | 0.07192 |
| **52** | [lsuperiorfrontal] | 0.00000 | 0.12700 | 0.00000 | 0.08273 |
| **53** | [rsuperiorfrontal] | 0.00000 | 0.07319 | 0.00000 | 0.08999 |
| **56** | [lsuperiortemporal] | 0.00000 | 0.30493 | 0.00000 | 0.04648 |
| **57** | [rsuperiortemporal] | 0.00000 | 0.21610 | 0.00000 | 0.04084 |
| **58** | [lsupramarginal] | 0.00000 | 0.05571 | 0.00000 | 0.07771 |
| **59** | [rsupramarginal] | 0.00000 | 0.15719 | 0.00000 | 0.07199 |
| **60** | [lfrontalpole] | 0.00004 | 0.81297 | 0.00021 | 0.02720 |
| **61** | [rfrontalpole] | 0.11548 | 0.48517 | 0.25279 | 0.00139 |
| **63** | [rtemporalpole] | 0.66397 | 0.05887 | 0.16365 | 0.00300 |
| **64** | [ltransversetemporal] | 0.00017 | 0.83627 | 0.00084 | 0.02226 |
| **65** | [rtransversetemporal] | 0.00002 | 0.54351 | 0.00005 | 0.03228 |
| **66** | [linsula] | 0.11404 | 0.35838 | 0.21776 | 0.00194 |
| **67** | [rinsula] | 0.04626 | 0.48201 | 0.12230 | 0.00407 |

-Volumetric Measures

| | regions | t_pvalue_age | t_pvalue_sex | t_pvalue_TIV | F_pvalue | R_square |
|---|---|---|---|---|---|---|
| **0** | [lStriatum] | 0.00000 | 0.33632 | 0.00000 | 0.00000 | 0.16151 |
| **1** | [lGloPal] | 0.03322 | 0.05412 | 0.54572 | 0.00172 | 0.02218 |
| **3** | [lAntCerebLI_II] | 0.03540 | 0.89398 | 0.00533 | 0.00026 | 0.02935 |
| **4** | [lAntCerebLIII] | 0.00035 | 0.27750 | 0.00001 | 0.00000 | 0.09928 |
| **5** | [lAntCerebLIV] | 0.00116 | 0.18719 | 0.00055 | 0.00000 | 0.07438 |
| **6** | [lAntCerebLV] | 0.00038 | 0.78892 | 0.00000 | 0.00000 | 0.12936 |
| **7** | [lSupPostCerebLVI] | 0.00007 | 0.98186 | 0.00000 | 0.00000 | 0.12627 |
| **8** | [lSupPostCerebCI] | 0.00001 | 0.53256 | 0.00000 | 0.00000 | 0.10905 |
| **9** | [lSupPostCerebCII] | 0.00066 | 0.64579 | 0.00003 | 0.00000 | 0.07842 |
| **10** | [lSupPostCerebLVIIB] | 0.00219 | 0.39249 | 0.00000 | 0.00000 | 0.06410 |
| **11** | [lInfPostCerebLVIIIA] | 0.00020 | 0.47065 | 0.00000 | 0.00000 | 0.07549 |
| **12** | [lInfPostCerebLVIIIB] | 0.00013 | 0.76046 | 0.00001 | 0.00000 | 0.08806 |
| **13** | [lInfPostCerebLIX] | 0.00212 | 0.92950 | 0.00060 | 0.00000 | 0.04841 |
| **14** | [lInfPostCerebLX] | 0.03519 | 0.82050 | 0.00021 | 0.00001 | 0.04265 |
| **15** | [lAntCerebWM] | 0.00430 | 0.20660 | 0.00371 | 0.00000 | 0.05619 |
| **16** | [lAmy] | 0.01067 | 0.89119 | 0.00000 | 0.00000 | 0.18984 |

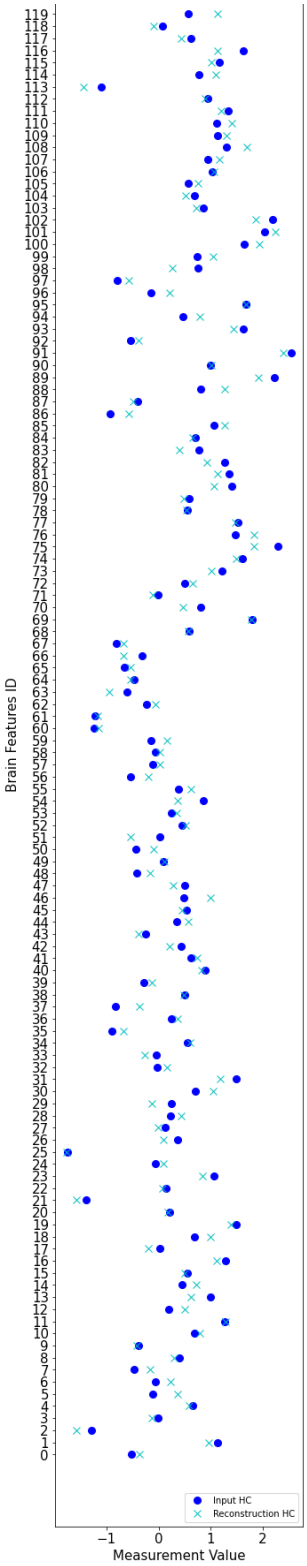
| | | | | | | |
|---|---|---|---|---|---|---|
| **17** | [lHCA1] | 0.34310 | 0.07650 | 0.00000 | 0.00000 | 0.13363 |
| **18** | [lSub] | 0.97010 | 0.25252 | 0.00000 | 0.00000 | 0.15618 |
| **19** | [lFor] | 0.26649 | 0.05550 | 0.16547 | 0.19866 | 0.00307 |
| **20** | [lCA4] | 0.12245 | 0.06177 | 0.00000 | 0.00000 | 0.14019 |
| **21** | [lCA2_3] | 0.37184 | 0.21419 | 0.00000 | 0.00000 | 0.06261 |
| **22** | [lStratum] | 0.17010 | 0.43103 | 0.00000 | 0.00000 | 0.17025 |
| **23** | [lFimbra] | 0.65436 | 0.35974 | 0.00543 | 0.02983 | 0.01098 |
| **25** | [lAlveus] | 0.19918 | 0.69526 | 0.00000 | 0.00000 | 0.10157 |
| **26** | [rStriatum] | 0.00000 | 0.34789 | 0.00000 | 0.00000 | 0.17545 |
| **27** | [rGloPal] | 0.17158 | 0.22536 | 0.27566 | 0.02066 | 0.01245 |
| **29** | [rAntCerebLI_II] | 0.05806 | 0.62897 | 0.00011 | 0.00000 | 0.05522 |
| **30** | [rAntCerebLIII] | 0.00032 | 0.38989 | 0.00003 | 0.00000 | 0.08697 |
| **31** | [rAntCerebLIV] | 0.00109 | 0.37242 | 0.00003 | 0.00000 | 0.08329 |
| **32** | [rAntCerebLV] | 0.00010 | 0.30032 | 0.00000 | 0.00000 | 0.13963 |
| **33** | [rSupPostCerebLVI] | 0.00001 | 0.63251 | 0.00000 | 0.00000 | 0.12782 |
| **34** | [rSupPostCerebCI] | 0.00000 | 0.86117 | 0.00000 | 0.00000 | 0.11023 |
| **35** | [rSupPostCerebCII] | 0.00830 | 0.98475 | 0.00000 | 0.00000 | 0.07550 |
| **36** | [rSupPostCerebLVIIB] | 0.01054 | 0.81302 | 0.00000 | 0.00000 | 0.07394 |
| **37** | [rInfPostCerebLVIIIA] | 0.00043 | 0.76228 | 0.00000 | 0.00000 | 0.08563 |
| **38** | [rInfPostCerebLVIIIB] | 0.00022 | 0.96792 | 0.00002 | 0.00000 | 0.07524 |
| **39** | [rInfPostCerebLIX] | 0.00214 | 0.55281 | 0.00319 | 0.00000 | 0.04842 |
| **40** | [rInfPostCerebLX] | 0.00292 | 0.55313 | 0.00001 | 0.00000 | 0.06335 |
| **41** | [rAntCerebWM] | 0.01837 | 0.08518 | 0.00397 | 0.00000 | 0.05943 |
| **42** | [rAmy] | 0.08770 | 0.31608 | 0.00000 | 0.00000 | 0.18621 |
| **44** | [rSub] | 0.93086 | 0.05630 | 0.00000 | 0.00000 | 0.14618 |
| **45** | [rFor] | 0.71614 | 0.08329 | 0.05553 | 0.22737 | 0.00248 |
| **47** | [rCA2_3] | 0.43310 | 0.11757 | 0.00000 | 0.00000 | 0.07336 |
| **48** | [rStratum] | 0.83753 | 0.12255 | 0.00000 | 0.00000 | 0.17487 |
| **51** | [rAlveus] | 0.60119 | 0.11199 | 0.00000 | 0.00000 | 0.11066 |

3)Linear Regressions for which t_p_value of **TIV** is not significant

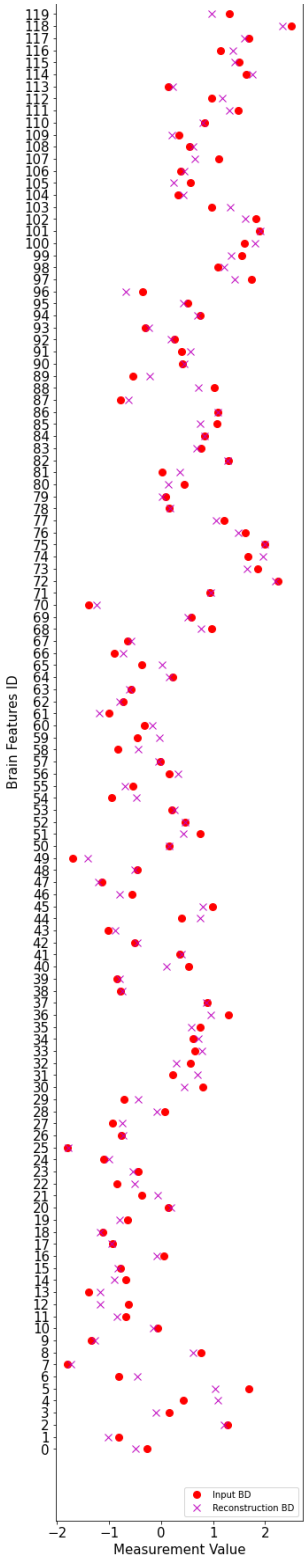
-Volumetric Measures

| | regions | t_pvalue_age | t_pvalue_sex | t_pvalue_TIV | F_pvalue | R_square |
|---|---|---|---|---|---|---|
| **1** | [lGloPal] | 0.03322 | 0.05412 | 0.54572 | 0.00172 | 0.02218 |
| **19** | [lFor] | 0.26649 | 0.05550 | 0.16547 | 0.19866 | 0.00307 |

| | | | | | | |
|---|---|---|---|---|---|---|
| **27** | [rGloPal] | 0.17158 | 0.22536 | 0.27566 | 0.02066 | 0.01245 |
| **45** | [rFor] | 0.71614 | 0.08329 | 0.05553 | 0.22737 | 0.00248 |




a)20th HC subject Reconstruction.




b)56th BD patient Reconstruction.

- **Section 6.5: Normative Model Results**

**BD Abnormal Brain Regions (all processing pipelines)**

**-Pipeline 1**

| ID | regions | Stat. | p-value |
|---|---|---|---|
| 1 | [rbankssts] | 16459.0 | 0.045991 |
| 4 | [lcaudalmiddlefrontal] | 16547.0 | 0.039140 |
| 12 | [linferiorparietal] | 16647.0 | 0.032389 |
| 20 | [llateralorbitofrontal] | 16578.0 | 0.036935 |
| 25 | [rmedialorbitofrontal] | 17694.0 | 0.003002 |
| 39 | [rpericalcarine] | 16544.0 | 0.039359 |
| 43 | [rposteriorcingulate] | 16963.0 | 0.017059 |
| 62 | [ltemporalpole] | 17247.0 | 0.009064 |
| 65 | [rtransversetemporal] | 16570.0 | 0.037494 |
| 68 | [lStriatum] | 18478.0 | 0.000311 |
| 70 | [lTha] | 16981.0 | 0.016415 |
| 85 | [lHCA1] | 17145.0 | 0.011445 |
| 90 | [lStratum] | 16967.0 | 0.016914 |
| 95 | [rGloPal] | 16473.0 | 0.044841 |
| 100 | [rAntCerebLV] | 16959.0 | 0.017206 |
| 101 | [rSupPostCerebLVI] | 16523.0 | 0.040921 |
| 105 | [rInfPostCerebLVIIIA] | 17535.0 | 0.004517 |
| 109 | [rAntCerebWM] | 16691.0 | 0.029739 |
| 115 | [rCA2_3] | 17526.0 | 0.004620 |
| 117 | [rFimbra] | 16499.0 | 0.042766 |

**-Pipeline 2**

| ID | regions | Stat. | p-value |
|---|---|---|---|
| 4 | [lcaudalmiddlefrontal] | 16667.0 | 0.031161 |
| 39 | [rpericalcarine] | 16879.0 | 0.020359 |
| 43 | [rposteriorcingulate] | 18434.0 | 0.000357 |

| | | | |
|---|---|---|---|
| **46** | [lprecuneus] | 16431.0 | 0.048362 |
| **60** | [lfrontalpole] | 17187.0 | 0.010406 |
| **68** | [lStriatum] | 18365.0 | 0.000442 |
| **70** | [lTha] | 16679.0 | 0.030443 |
| **84** | [lAmy] | 16883.0 | 0.020190 |
| **85** | [lHCA1] | 17307.0 | 0.007876 |
| **90** | [lStratum] | 17004.0 | 0.015622 |
| **100** | [rAntCerebLV] | 16693.0 | 0.029623 |
| **105** | [rInfPostCerebLVIIIA] | 17120.0 | 0.012106 |
| **112** | [rSub] | 16662.0 | 0.031465 |

## -Pipeline 3

| ID | regions | stats | p-value |
|---|---|---|---|
| **33** | [rparsopercularis] | 12398.0 | 0.035298 |
| **53** | [rsuperiorfrontal] | 11678.0 | 0.007601 |
| **56** | [lsuperiortemporal] | 12395.0 | 0.035097 |
| **102** | [rSupPostCerebCI] | 12021.0 | 0.016486 |

## -Pipeline 4

| ID | Regions | Stat. | p-value |
|---|---|---|---|
| **1** | [rbankssts] | 17207.0 | 0.009940 |
| **10** | [lfusiform] | 17130.0 | 0.011838 |
| **28** | [lparahippocampal] | 16540.0 | 0.039653 |
| **31** | [rparacentral] | 16761.0 | 0.025895 |
| **43** | [rposteriorcingulate] | 17097.0 | 0.012743 |
| **54** | [lsuperiorparietal] | 16452.0 | 0.046575 |
| **70** | [lTha] | 16439.0 | 0.047675 |
| **72** | [lAntCerebLIII] | 16905.0 | 0.019284 |
| **92** | [lMamBody] | 16641.0 | 0.032765 |
| **95** | [rGloPal] | 17099.0 | 0.012687 |

## Conclusions

- **Best discriminative model – 2.2**

Although the best model was found through a hyperparameter tuning, the hyperparameter combination trials were divided into several steps, thus leading to the testing of several models which until a certain point were considered the "best" model. The model results presented in the following section yield the best generalizability performance in the external set, using pipeline 5.

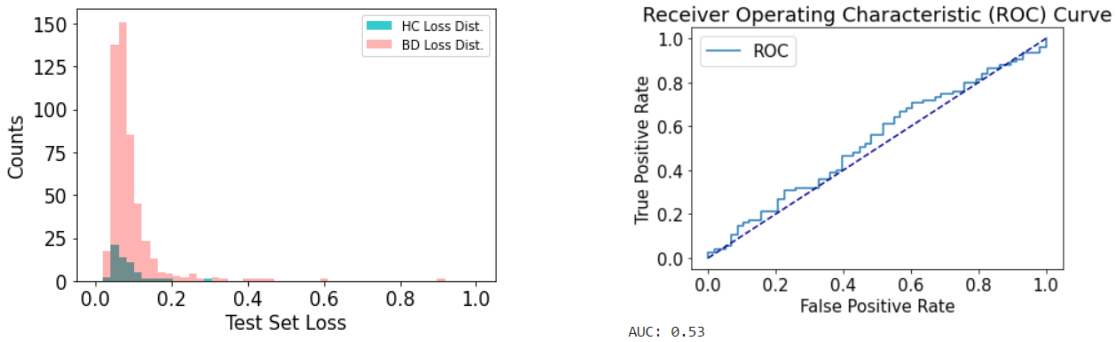
Model: hyperparameter combination from trial 2.2

| | | | |
|---|---|---|---|
| **Layer 1,3:** 100 | **Layer 2:** 80 | **L2:** 0.0001 | **Lr:** 0.001 |

### I. Training

| Epochs | Train Loss | Train MSE | Test Loss | Test MSE |
|---|---|---|---|---|
| 536 | 0.0455 | 0.0331 | 0.0949 | 0.0825 |

### II. Testing

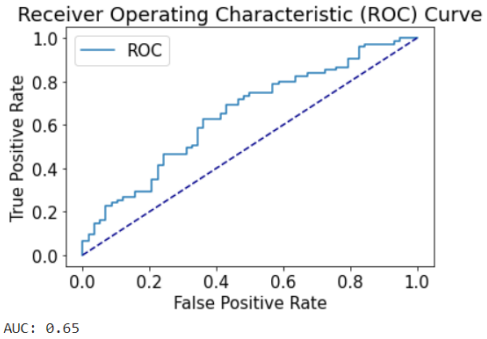



Overlapping Distributions and AUC-ROC on test set.

### III. Feature Selection

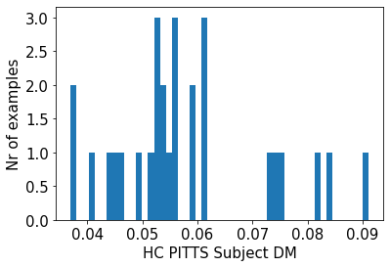
| | Regions | Statistic | p-value |
|---|---|---|---|
| **30** | lparacentral | 17099 | 0.0127 |
| **64** | ltransversetemporal | 16433 | 0.0482 |
| **70** | lTha | 17637 | 0.0035 |

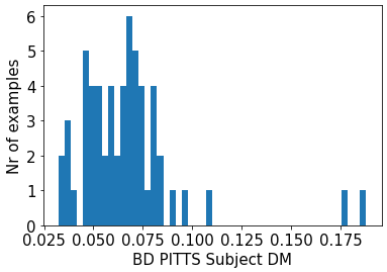
| | | | |
|---|---|---|---|
| **71** | lAntCerebLI_II | 17308 | 0.0078 |
| **83** | lAntCerebWM | 16473 | 0.0448 |
| **85** | lHCA1 | 17087 | 0.0130 |
| **95** | rGloPal | 17245 | 0.0091 |
| **105** | rInfPostCerebLVIIIA | 16911 | 0.0190 |
| **113** | rFor | 16517 | 0.0414 |

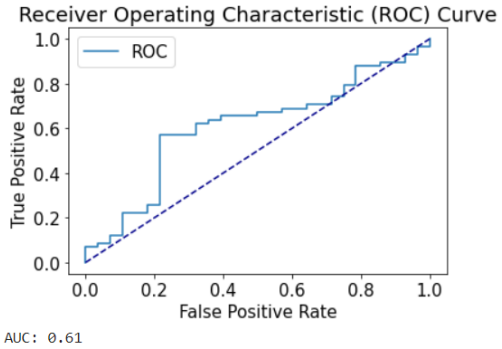


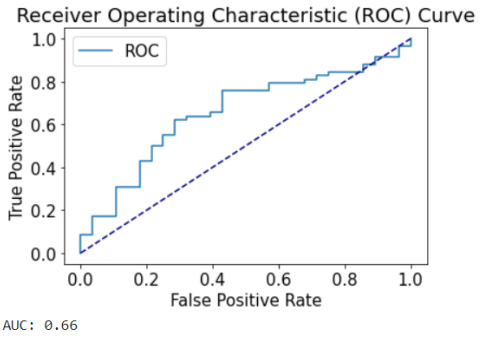

AUC-ROC curve on test set with feature subset

## IV. Classification




Results on external set. AUC-ROC curve (righ) on feature subset.

# List of Figures

# List of Tables

# 2. Acknowledgements

The present thesis work development was possible thanks to the collaboration between B3Lab (Biosignal-Bioimaging-Bioinformatics) in the Department of Electronics, Informatics and Bioengineering (DEIB) at Politecnico di Milano in collaboration with the MiBrain (Milan Brain Research on Affective and Integrative Neuroscience) Lab coordinated by Prof. Paolo Brambilla, co-adviser of this thesis, in the Psychiatry Unit of the Fondazione IRCCS Ca' Granda Ospedale Maggiore Policlinico and the University of Milan.

I would like to thank my adviser Prof. Eleonora Maggioni, for following my work attentively, always being supportive, and for her experienced constructive feedback and advice. I would also like to thank my co-adviser Emma Tassi, Ph.D. candidate at DEIB, for being present during all the steps, and for her willingness in helping me, always making sure to clarify any doubts or uncertainties I might have.

Last, but not least, I want to thank my family. My mum and dad, who have supported and encouraged me in coming to Politecnico di Milano in the first place to take my master's degree. They have believed in me and have cheered me during all my time here. Also want to thank my grandparents and all my family who have always been so present, caring, and supportive. Finally, I could not have finished this phase of my life (in a sane manner) without the constant support of my boyfriend Davide, whom I want to thank with all my heart. You have been my rock and sustain here.