Joana Sofia Baião Vieira

On the short-term prediction of Multiple Sclerosis disease progression

Thesis submitted to the Faculty of Science and Technology of the University of Coimbra for the Master's degree in Biomedical Engineering with specialisation in Clinical Informatics and Bioinformatics, supervised by Prof. Dr. César Alexandre Domingues Teixeira and MSc Mauro Filipe da Silva Pinto.

September 2022

1 2 9 0

FACULDADE DE CIÊNCIAS E TECNOLOGIA
UNIVERSIDADE D COIMBRA

On the short-term prediction of Multiple Sclerosis disease progression

Joana Sofia Baião Vieira

Dissertion presented to the University of Coimbra in order to complete the necessary requirements to obtain the Master's degree in Biomedical Engineering.

Supervisors:
Prof. Dr. César Alexandre Domingues Teixeira (CISUC)
Prof. Mauro Filipe da Silva Pinto (CISUC)

Coimbra, 2022

This work was developped in collaboration with:

CISUC - Center for Informatics and Systems of the University of Coimbra

This work was supported by the Portuguese Foundation for Science and Technology through the CISUC project (UIDB/00326/2020).

Agradecimentos

E assim se concluiu o capítulo final de uma longa caminhada.

Quero começar por agradecer aos meus orientadores por todo o apoio desde o início deste projeto. Ao Professor Doutor César Teixeira expresso a minha admiração por todo o apoio científico, motivação e confiança que demonstrou ao longo deste ano. Ao Mauro Pinto, o meu profundo agradecimento, não só por toda a disponibilidade, mas também pelo rigor da crítica e comentários que me deu, pelo otimismo e pelo encorajamento constante. Fico-vos grata pelas aprendizagens que me proporcionaram ao longo deste percurso, tanto a nível pessoal como profissional.

Ao Departamento de Física e à jeKnowledge, o meu sincero obrigada, tanto pelas competências, como pelo grupo de pessoas queridas que levarei sempre comigo.

Agradeço a todas as amizades de Coimbra que tornaram estes 5 anos tão especiais e a todos aqueles que cruzaram o meu caminho e o marcaram de alguma forma. Em especial, ao meu grupo de amigos da universidade, por todos os momentos, companheirismo e gargalhadas que tornaram esta experiência tão boa. À minha Jacinta, a amiga-casa, por todas as conversas e histórias que partilhamos. À Carolina, que tornou este caminho mais leve e bonito. Espero poder contar convosco por muitos mais anos e continuar a fazer memórias incríveis ao vosso lado neste novo capítulo da minha vida.

Aos de sempre, os de Alpendorada, pela amizade, por me acompanharem durante estes anos e por estarem comigo em todas as situações. E assim continuará.

Um grande obrigada ao Luís, por todos os momentos e por tornar os meus dias melhores com todo o amor e carinho.

Deixo o maior agradecimento à minha família, em especial aos meus pais e irmã, pelo apoio incondicional, pelos ensinamentos, pela confiança que sempre depositaram em mim e por me encorajarem sempre a seguir os meus sonhos. Tornarem tudo isto possível.

Resumo

A Esclerose Múltipla é uma doença inflamatória crónica do sistema nervoso central que leva à incapacitação dos doentes e tem também muitos impactos sociais e económicos. Os sintomas e a sua evolução variam muito de pessoa para pessoa. Por conseguinte, um componente crítico da gestão da doença é a previsão dos doentes que irão transitar para o curso Secundário Progressivo (SP). Esta previsão precoce é uma abordagem promissora que permitiria a adoção de melhores estratégias de tratamento e uma gestão das expectativas do doente mais realista.

Apesar da evolução das técnicas de Machine Learning (ML) ao longo dos anos, até à data estes modelos de previsão do curso da doença ainda não atingiram a aplicabilidade clínica. Os principais fatores que limitam a sua utilização são a não-transparência dos resultados e consequente falta da garantia de confiança e segurança dos modelos. Nos últimos anos o conceito de explicabilidade ganhou um maior peso e atualmente diversos estudos concentram-se na transformação dos modelos de ML em modelos mais interpretáveis.

No presente estudo foi utilizada a base de dados do serviço de Esclerose Múltipla do hospital de Sant'Andrea, de Roma, para prever se um doente transitará para o curso SP numa janela temporal de 180, 360 ou 720 dias. Os modelos desenvolvidos por Seccia et al. [1] foram parcialmente replicados e melhorados. Foram estudados dois cenários: o orientado para as visitas (VO), no qual foram utilizados os classificadores Random Forest (RF), Support Vector Machines (SVMs) linear e não-linear, k-nearest neighbours (KNN) e AdaBoost (AB) para prever considerando uma única visita; o orientado para a história clínica (HO), no qual se aplicou uma rede neuronal Long Short-Term Memory (LSTM) que considera o histórico de visitas do doente para fazer a previsão.

Os modelos de previsão obtiveram medidas de F1-score de 28 a 37% para o cenário VO e 71 a 77% para o cenário HO quando se utilizaram os datasets com o maior número de visitas. Estes resultados mostram que as redes neuronais LSTM prevêem eficazmente o curso SP quando se utilizam dados disponíveis na rotina clínica em maiores quantidades.

Foram também aplicados diversos métodos de explicabilidade para gerar explicações sobre o comportamento global dos modelos de ML desenvolvidos e sobre previsões específicas para certos doentes. As conclusões obtidas foram limitadas pela falta de conhecimento sobre o significado de cada característica. Ainda assim, as explicações mostraram que a escala de quantificação da condição neurológica (EDSS) é bastante relevante na classificação da progressão da doença.

Palavras-chave: Esclerose Múltipla; Machine Learning; Progressão; Previsão; Explicabilidade.

Abstract

Multiple Sclerosis (MS) is a chronic inflammatory disease of the Central Nervous System (CNS) that leads to disability in patients and has many social and economic impacts. Symptoms and their course vary significantly from person to person. Therefore, a critical component of disease management is predicting patients who will transition to the Secondary Progressive (SP) course. This early prediction is a promising approach that would allow for better treatment strategies and more realistic management of patient expectations.

Despite the evolution of ML techniques over the years, these disease course prediction models have not yet reached clinical applicability. The non-transparency of the results and consequent lack of confidence and safety of the models are the main factors limiting their use. In recent years, the concept of explainability has gained more significant weight and, currently, several studies focus on transforming ML models into more interpretable models.

In the present study, the dataset of the MS service of Sant'Andrea hospital was used to predict whether a patient will transition to the SP phase in a time window of 180, 360 or 720 days. The models developed by Seccia et al. [1] were partially replicated and improved. Two scenarios were studied: the Visited-Oriented (VO), in which RF, linear and non-linear SVM, KNN and AB classifiers were used to predict the transition to SP considering a single visit; the History-Oriented (HO), in which it was applied a LSTM Neural Network (NN) that considers the patient's entire clinical history to make the prediction.

When using the datasets with the highest number of visits, the prediction models obtained F1-score measures of 28 to 37% for the VO scenario and 71 to 77% for the HO scenario. These results show that LSTM NNs effectively predict the SP course when using larger quantities of data available in the clinical routine.

Several explainability methods were also applied to explain the overall behaviour of the developed ML models and specific predictions for particular patients. The conclusions obtained were limited by the lack of knowledge about the meaning of each feature. Still, the explanations showed that the Expanded disability status scale (EDSS) scale is quite relevant in classifying disease progression.

Keywords: Multiple Sclerosis; Machine Learning; Progression; Prediction; Explainability.

Contents

List of Figures

List of Tables

List of Abbreviations

FLAIR Fluid-attenuated inversion recovery. 55

FN False Negatives. 30, 31, 32, 78

FP False Positives. 30, 31, 32, 78

FPR False Positive Ratio. 32

FS Functional System. 7, 8, 9, 10, 47, 56

GDPR General Data Protection Regulation. 33, 54

HLA Human Leukocyte Antigen. 5, 6

HO History-Oriented. iv, vi, xii, xiii, xiv, xv, xvi, 3, 58, 68, 75, 76, 80, 89, 90, 91, 92, 93, 94, 96, 97, 100, 102, 105, 125, 127, 128

ICE Individual Conditional Expectation. xii, xiii, 37, 38, 73, 83, 84, 90, 91, 101

IM Intramuscular. 15

IV Intravenous. 15

KNN *k*-nearest neighbours. iv, vi, xi, 18, 24, 36, 41, 47, 48, 51, 63, 66, 67, 76, 77, 99

LASSO Least Absolute Shrinkage and Selection Operator. 20, 66, 78, 80, 123, 124, 125, 127, 128

LDA Linear Discriminant Analysis. 19, 51, 98

LIME Local Interpretable Model-Agnostic Explanations. xii, xiii, 40, 41, 54, 56, 74, 75, 85, 86, 87, 88, 92, 93, 94, 101, 102

LOGO Leave One Group Out. xii, 53, 64, 65, 69, 97

LOOCV Leave One Out Cross Validation. 22

LR Logistic regression. 23, 24, 36, 47, 48, 49, 50, 52, 98

LRP Layer-wise Relevance Propagation. 54, 55

LSTM Long Short-Term Memory. iv, vi, xi, xii, 28, 51, 52, 58, 68, 69, 70, 71, 72, 76, 79, 96, 97, 105

MAP Maximum *a posteriori*. 25

MAR Missing at Random. 17

MCAR Missing Completely at Random. 17

MEP Motor Evoked Potential. 49

ML Machine Learning. iv, v, vi, ix, xi, xii, xiv, 2, 3, 4, 16, 17, 18, 23, 29, 32, 33, 34, 35, 36, 45, 46, 47, 48, 49, 52, 53, 54, 55, 56, 57, 58, 61, 63, 64, 65, 68, 74, 78, 95, 97, 100, 103, 105

MLE Maximum Likelihood Estimation. 23

MMD Maximum Mean Discrepancy. 42

MNAR Missing Not at Random. 17

MRI Magnetic Resonance Imaging. 6, 7, 11, 12, 14, 46, 47, 48, 49, 54, 55, 59, 61, 63, 81, 97

MS Multiple Sclerosis. vi, ix, xi, xiv, xvii, 1, 2, 3, 4, 5, 6, 7, 10, 11, 12, 13, 14, 15, 21, 32, 44, 45, 46, 47, 48, 49, 50, 51, 52, 53, 54, 55, 56, 57, 58, 59, 85, 96, 98, 101, 102, 103, 104, 105, 106

NB Naive Bayes. 25, 26, 36, 49

NN Neural Network. vi, xiv, xv, 28, 34, 36, 43, 44, 48, 49, 51, 52, 56, 59, 68, 69, 70, 75, 79, 80, 96, 97, 98, 100, 105, 125, 126

NPV Negative Predictive Value. 48

PCA Principal Component Analysis. 19

PDP Partial Dependence Plot. xi, xiii, 37, 43, 73, 83, 84, 90, 91, 101, 102

PFI Permutation Feature Importance. xiii, 56, 73, 82, 90, 101

PP Primary Progressive. 2, 11, 12, 15, 21, 44, 60

PPV Positive Predictive Value. 48

PR Progressive Relapsing. 11, 12

RBF Radial Basis Function. 26, 98

RF Random Forest. iv, vi, 48, 50, 51, 52, 63, 66, 67, 76, 77, 99

RIS Radiologically Isolated Syndrome. 12

RK Record-keeping. xii, xiii, xiv, xv, xvi, 58, 60, 61, 62, 63, 64, 68, 77, 78, 79, 80, 81, 82, 83, 84, 85, 86, 87, 88, 89, 90, 91, 92, 93, 94, 96, 97, 98, 99, 100, 103, 105, 124, 125, 126, 128

RNN Recurrent Neural Network. 28, 69

ROC Receiver Operating Characteristic. xi, 32, 33

RR Relapse Remitting. 1, 2, 3, 11, 12, 13, 14, 15, 16, 21, 31, 32, 44, 45, 46, 47, 48, 51, 52, 58, 60, 68, 75, 78, 85, 97, 98, 102, 105

SBS Sequential Backward Selection. 20

SC Subcutaneous. 15

SFS Sequential Feature Selection. 20

SHAP SHapley Additive exPlanations. 54, 56, 103

SP Secondary Progressive. iv, v, vi, ix, 1, 2, 3, 11, 12, 13, 14, 15, 16, 21, 31, 32, 44, 45, 46, 47, 48, 49, 50, 51, 52, 58, 60, 64, 68, 75, 76, 78, 79, 83, 84, 85, 88, 91, 96, 97, 98, 101, 102, 105, 106

SP-LIME Submodular Pick-LIME. xiii, 74, 75, 85, 86, 91, 92, 93, 101, 102

SVM Support Vector Machines. iv, vi, xi, xiv, 26, 27, 39, 47, 48, 49, 50, 51, 52, 63, 66, 67, 76, 77, 78, 97, 98, 99, 105

SWI Susceptibility-weighted images. 55

TN True Negatives. 30, 31
TNR True Negative Ratio. 31
TP True Positives. 30, 31
TPR True Positive Ratio. 31, 32

VO Visited-Oriented. iv, vi, xii, xiii, xiv, xv, 3, 58, 63, 64, 75, 76, 77, 78, 81, 82, 83, 84, 86, 87, 88, 96, 97, 98, 99, 100, 103, 105, 123, 124, 125

XGBoost Extreme Gradient Boosting. 56

1

Introduction

This chapter is divided into four sections. Initially, the motivations of this work are presented (section 1.1). Section 1.2 explains the context of the this work, and in section 1.3, the main goals are enumerated. Lastly, the structure of the document and a brief description of each chapter appear in section 1.4.

1.1 Motivation

Multiple Sclerosis (MS) affects more than 2.8 million people worldwide, which means that this disease affects 1 in every 3000 people. MS is the leading cause of non-traumatic disability in adults between 20 and 40 years old. Its prevalence has increased in most countries, with an increase of about 30% since 2013 [2]. In addition, it is known that women are more affected than men, with a ratio of approximately 3:1 [19].

MS is an autoimmune neurologic disorder of the Central Nervous System (CNS) characterised by myelin damage. This damage forms scar tissue (sclerosis), followed by an alteration or stoppage of electrical impulses conduction to and from the brain and spinal cord. The MS cause is unknown but is thought to be triggered by multiple factors [20]. MS is not a curable disease, but treatment can help manage it. The treatment is multidisciplinary and includes different management strategies that help modify or slow the course of the disease, such as rehabilitation, Disease-modifying teraphies (DMTs), symptom treatment, psychological support, and lifestyle modifications [21].

The symptomatology is very heterogeneous and includes motor, cognitive, and sometimes psychiatric problems [22]. These MS symptoms vary widely from person to person and are unpredictable, can change over time, and be mild, moderate, or severe [22]. Thus, predicting the evolution of the disease over the years and the transition from Relapse Remitting (RR) to Secondary Progressive (SP) are challenges that put pressure on the physicians.

It is fundamental to invest in strategies for early diagnosis and prognosis, and treatment since they lead to better results in controlling the long-term progression of the disease and, consequently, to a reduction of disability and economic and personal costs [23]. It also allows the patient to manage expectations in relation to treatment processes, treatment outcomes, or overall disease management since it is a chronic disease that impacts mental health [24].

1.2 Context

MS is usually divided into three phenotypes, namely RR, Primary Progressive (PP), and SP, but its course is highly variable and heterogeneous. The majority of MS patients will present the RR course (>85%), which is characterised by new symptoms appearing in isolated attacks and its complete disappearance with no disease progression between relapses. The large majority of RR patients (>80%) will eventually evolve to SP course, a progressive phase of the disease with worsening of the neurological function and accumulation of disability [5, 20].

The physical and cognitive disability in patients with SP gradually increases over time. The challenges and limitations faced by increased disability greatly impact the patient's basic life activities and ability to work, leading to disruptive consequences on family life, interpersonal relationships, and economic status. Treatment costs tend to increase significantly as the disease worsens [25].

Furthermore, identifying patients at higher risk and adopting more aggressive and appropriate treatments lead to more efficient control of long-term progression. This early diagnosis is a key point to prevent the disease from progressing to a stage where treatment is not effective, since late diagnosis of SP course significantly influences permanent disability [26]. Additionally, identifying high-risk patients prevents low-risk patients from being exposed to aggressive and relatively unsafe therapies. Finally, it will also allow the selection of patients for clinical trials in a more homogeneous method [27].

However, there are no clear clinical and imaging criteria to identify this gradual transition, and it is difficult to understand when it occurs [28, 29]. A possibility is a Machine Learning (ML) algorithm capable of predicting the SP course in an early and individualised way. Therefore, predicting and consequently preventing or delaying the onset of SP will allow better management of patients at higher risk of worsening disability and SP conversion [28, 29].

1.3 Main goals

The goal of this thesis is to evaluate and predict MS progression from the RR to the SP form, using the dataset processed and worked on by Seccia et al. [1], respectively. This goal can be divided into:

- Development of ML algorithms similar to the ones developed by Seccia et al. [1] to predict if the patient will pass to SP course at 180, 360, or 720 days from the last visit, considering results from a single visit (Visited-Oriented (VO) approach) and sequences of consecutive visits (History-Oriented (HO) approach);
- Exploration of different methods to improve the results from the previously created models;
- Exploration of strategies to identify the features of the dataset. The dataset authors hid the meaning of the features to protect patient identity. Still, in this work, it is essential to know their meaning to explore the explicability of the models;
- Exploration of the explainability of ML methods designed for MS disease progression;
- Analysis of the applicability of these methods in the clinical context, considering their safety and physicians' trust.

1.4 Structure

This dissertation is divided into seven chapters, beyond the introduction:

- **Chapter 2:** presents the background information about the MS disease and ML methods that will be mentioned throughout the document;
- **Chapter 3:** summarises the state of the art ML methods designed for MS disease progression and their limitations;
- **Chapter 4:** describes the steps of the experimental procedure adopted in this Master's thesis;
- **Chapter 5:** reports the results obtained in this study;
- **Chapter 6:** contains the discussion of the dataset, methodology and results.
- **Chapter 7:** presents the main conclusions and addresses future work.

2

Background Concepts

This chapter presents the main concepts necessary to understand the work developed in this thesis. In section 2.1 the Multiple Sclerosis (MS) disease is described, along with several related concepts, such as risk factors, disease stages, diagnosis, and therapy. Section 2.2 introduces and describes several fundamental Machine Learning (ML) algorithms and strategies. Lastly, section 2.3 presents the concepts of explainability and interpretability, and the methods used to achieve it.

2.1 Multiple Sclerosis

MS is a chronic autoimmune neurologic disorder of the Central Nervous System (CNS) in which inflammation, demyelination, and axonal loss occur. It is the most common non-traumatic disabling disease in young adults globally, typically affecting patients between 20 and 40 years of age [30]. The course of MS is highly diverse and unpredictable [31]. It affects 2.8 million people worldwide, and about two-thirds of those affected are women. The global median prevalence is 1 in 3000 people living with MS [2].

The damaging of myelin, the protective coating surrounding nerve fibres, changes the way nerve impulses are conducted, making it more difficult to send messages. The nerve fibres become progressively vulnerable to damage [32]. Its ongoing damage disrupts the body's normal functioning, resulting in a continuous decrease in motor function, eventually leading to disability. MS causes different symptoms among patients, such as fatigue, walking difficulties, blurred vision, depression, and dizziness. Additionally, some patients have periods of relapse and remission while others have a progressive pattern [2, 25].

2.1.1 Risk factors

Although the cause of MS remains unknown, its aetiology and pathogenesis are best explained by a combination of factors, such as interactions between genetic, lifestyle, and environmental influences [33].

In the past few decades, studies on MS genetics have been an important piece in understanding the aetiology of this complex disease, showing that more than 200 genetic variants, mainly Human Leukocyte Antigen (HLA), are associated with the modification of the MS risk. These variants affect gene activity and regulatory mechanisms but only explain 20-30% of MS heritability, implying that gene-gene or gene-environment interactions may significantly influence the level of risk of contracting the disease [34].

This disease is also affected by the latitude gradient. Countries at high latitudes (northern hemisphere) have a higher prevalence of MS, as seen in the map in Figure 2.1. The difference in prevalence is thought to be related to ethnicity, socioeconomic structure, and the diagnostic criteria and methods countries adopt [35]. Thus, the lower number of cases in low-risk countries may result from the absence of data due to worse medical facilities and lower life expectancy. However, several studies [36] have shown the relationship between geographical latitude and levels of sun exposure and vitamin D in the risk of MS. The risk is highest in countries with low sunlight exposure and vitamin D deficiency [33, 34].

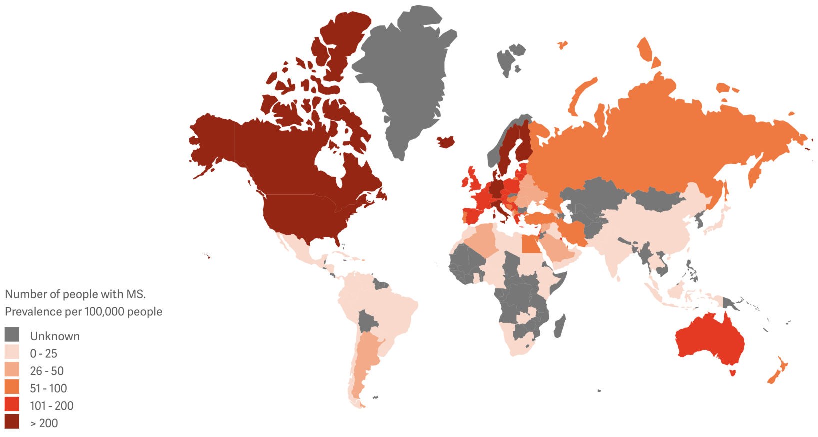


Figure 2.1: Worldwide prevalence of MS per 100 000 population in 2020. Extracted from [2].

Like most autoimmune diseases, MS has a higher prevalence among women, occurring at a rate of three women to every man. This disease appears mainly in the reproductive years, suggesting that puberty-associated neuroendocrine factors may play a role in the development of the disease, particularly in females [19].

Many infectious agents have been suggested to have a role in MS. The infection with Epstein–Barr virus (EBV) has consistently been a risk factor. The fact that a large part of adult patients have serologic evidence of prior infections [34] supports the idea that EBV infection during adolescence and childhood increases the risk of MS [37].

Certain lifestyle factors, such as smoking and obesity, are also associated with increased MS risk. Smoking and passive exposure to smoking provokes lung inflammation, which can trigger inflammatory and immune responses. Obesity in adolescence and young adulthood has been linked to an increased risk of developing MS. This risk could be influenced by interactions between obesity and both HLA antigen MS risk variants and EBV infection [37].

Despite the evidence of the impact of risk factors on the MS onset and modification of disease activity in MS patients, these factors are not considered in the methodology developed due to the lack of available data to explore this issue.

2.1.2 Diagnosis

Early diagnosis of MS is important since treatment can slow the disease and improve the patients quality of life. The MS heterogeneity, both in clinical and imaging manifestations, and the similarity with other diseases are two factors that make the diagnosis of MS challenging, often leading to misdiagnosis [15].

Since there is no single diagnostic test, it is made by combining clinical, imaging, and laboratory findings [21], such as Magnetic Resonance Imaging (MRI), to find disease-related alterations in anatomical connectivity, and Cerebrospinal fluid (CSF) testing, to identify CSF-specific oligoclonal bands. These paraclinical evaluations allow earlier and more sensitively and specifically diagnoses [15, 38].

Currently, diagnosis is based on the McDonald Criteria 2017 [15]. These criteria for diagnosing MS have been continuously improved over the years. They are based on two main pillars: the dissemination in time and space of the clinical picture caused by CNS lesions and the exclusion of other diseases with similar symptoms. The dissemination in time is the appearance of new CNS lesions over time, and the dissemination in space is the presence of distinct zones of the CNS [15].

The different components of the 2017 McDonald Criteria are presented in Table 2.1, and they result in three different cases:

- **Confirmed MS:** if the 2017 McDonald criteria are met and there is no better explanation for the clinical picture.
- **Possible MS:** if MS is suspected as a result of a Clinically Isolated Syndrome (CIS) but the 2017 McDonald criteria are not completely met.
- **Not MS:** if there is another diagnosis that better explains the clinical picture.

Table 2.1: The 2017 McDonald criteria for diagnosis of MS in patients with an attack at onset. [15]

Number of relapses	**Number of with objective clinical evidence**	**Additional data needed to the MS diagnosis**
≥ 2	≥ 2	No additional tests are required to demonstrate dissemination in space and time
≥ 2	1 (as well as clear-cut historical evidence of a previous attack involving a lesion in a distinct anatomical location)	No additional tests are required to demonstrate dissemination in space and time
≥ 2	1	Dissemination in space demonstrated by an additional clinical attack implicating a different CNS site or by MRI
1	≥ 2	Dissemination in time demonstrated by an additional clinical attack or by MRI OR demonstration of CSF-specific oligoclonal bands
1	≥ 2	Dissemination in space demonstrated by an additional clinical attack implicating a different CNS site or by MRI AND Dissemination in time demonstrated by an additional clinical attack or by MRI OR demonstration of CSF-specific oligoclonal bands

2.1.3 Expanded disability status scale (EDSS)

The Expanded disability status scale (EDSS) is a scale developed by Kurtzke [3] that is widely used in clinical trials and in the evaluation of people with MS. This scale allows the quantification of disability in MS, the monitoring of changes in the level of disability over time, and the evaluation of the effectiveness of therapeutic interventions.

The EDSS provides a score on a scale ranging from 0 (normal status) to 10 (death due to MS), with an increment of 0.5 units, where greater values represent higher levels of disability, as illustrated in Figure 2.2. Patients at levels 1 to 4.5 have a high degree of ambulatory capacity, while those at levels 5.0 to 9.5 have a loss of ambulatory capacity [3, 39].

The final EDSS score is determined by the gait analysis (locomotion) and Functional System (FS) score. The FS indicates the extent of impairment in each of the eight areas of the CNS [3, 39]:

1. **Pyramidal:** Related with muscle weakness and loss of voluntary control of muscular movements.
2. **Cerebellar:** Associated with changes in movement coordination and balance;

3. **Brain Stem:** Related to influences on the cranial nerves that can cause problems with speech, swallowing, and breathing.
4. **Sensory:** Related to loss of sensation below the head;
5. **Bowel and Bladder:** Responsible for urinary retention and incontinence;
6. **Visual:** Associated with vision impairments;
7. **Cerebral:** Responsible for problems with thinking, concentration and memory, as well as mood disorders.
8. **Other:** Related to other symptoms that do not fall under the functional systems above, such as pain and fatigue.

These systems are scored on a scale from 0 (low level of problems) to 5 or 6 (high level of problems), except the last one, which is 0 if no other symptoms are present and 1 if neurological findings are present. The first steps of the EDSS (0 to 3.5) are obtained according to FS, and the following steps also take into account the mobility impairment, and restrictions in the patients' daily life [3, 39].

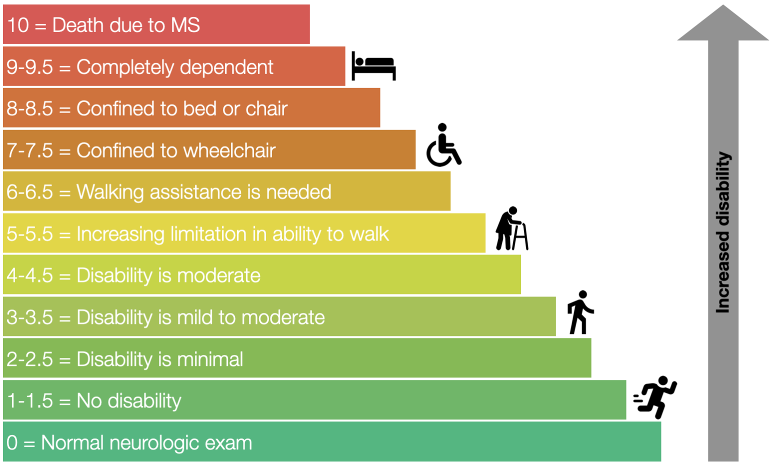

10 = Death due to MS
9-9.5 = Completely dependent
8-8.5 = Confined to bed or chair
7-7.5 = Confined to wheelchair
6-6.5 = Walking assistance is needed
5-5.5 = Increasing limitation in ability to walk
4-4.5 = Disability is moderate
3-3.5 = Disability is mild to moderate
2-2.5 = Disability is minimal
1-1.5 = No disability
0 = Normal neurologic exam
Increased disability

Figure 2.2: Schematic representation of the EDSS. Adapted from [3].

Thus, EDSS has a total of 20 steps which are described in detail below [3]:

- **EDSS 0:** Normal neurological examination. All FS have grade 0, except for Cerebral, which can assume grade 1.
- **EDSS 1:** No disability. One of the FS has grade 1, except for from Cerebral, which can assume grade 1.
- **EDSS 1.5:** No disability. More than one FS has grade 1, except for from Cerebral, which can assume grade 1.
- **EDSS 2:** Minimal disability in a FS with grade 2. The remaining FS can assume grade 0 or 1.

- **EDSS 2.5:** Minimum deficiency in two FS with grade 2. The remaining FS can assume grade 0 or 1.
- **EDSS 3:** Fully ambulatory, but with minimal efficiency in one FS with grade 3 or slight deficiency in three or four FS with grade 2; all other FS with grade 0 or 1.
- **EDSS 3.5:** Fully ambulatory, but with moderate disability in a FS with grade 3 and one or two FS with grade 2, or two FS with grade 3, or five FS with grade 2; all other FS with grade 0 or 1.
- **EDSS 4:** Fully ambulatory unaided and self-sufficient, but relatively severely impaired in one FS with grade 4 and remaining FS with grade 0 or 1, or with other combinations of lower grades exceeding the previous limits. Able to walk unaided or rest about 500 meters.
- **EDSS 4.5:** Fully ambulatory without help, but may have some limitation in full activities and require minimal assistance. Has a relatively severe disability, with one FS with grade 4 and remaining FS grades 0 or 1, or with other combinations of lower grades exceeding the previous limits. Able to walk unaided or rest about 300 meters.
- **EDSS 5:** Ambulatory without aid or rest for 200 meters, but with disability severe enough to impair full daily activities. Characterised by one FS with grade 5 and remaining FS with grade 0 or 1, or with other combinations of lower grades exceeding the limits of step 4.
- **EDSS 5.5:** Ambulatory without aid or rest for 100 meters, but with impairment severe enough to preclude the performance of full daily activities. It is characterised by one FS of grade 5 and remaining FS with grade 0 or 1, or with other combinations of lower grades exceeding the limits of step 4.
- **EDSS 6:** Needing intermittent or unilateral help (cane, crutch, or brace) to walk about 100 meters, regardless of the existence of rest. More than two FS have a grade of 3 or higher.
- **EDSS 6.5:** Need constant bilateral help (cane, crutch or device) to walk about 20 meters without rest. More than two FS have a grade of 3 or higher.
- **EDSS 7:** Inability to walk more than 5 meters, regardless of help, and limitation to a wheelchair, with ability to move and transfer alone. More than one FS has a grade of 4 or higher or, rarely, the pyramidal system alone has grade 5.
- **EDSS 7.5:** Inability to take more than a few steps and limitation to a wheelchair, considering that motorised wheelchair and transfer assistance may

be required. More than one FS has a grade of 4 or higher.

- **EDSS 8:** Restriction to bed or chair or limitation to a wheelchair, with the ability to be out of bed for much of the day and to perform self-care functions with effective use of arms. Generally, several FS have a grade of 4 or higher.
- **EDSS 8.5:** Restricted to bed for most of the day, with the ability to perform self-care functions with effective use of arms. Usually, several FS have a grade of 4 or higher.
- **EDSS 9:** Bedridden and helpless patient with ability to communicate and eat. Most FS have a grade of 4 or higher.
- **EDSS 9.5:** Bedridden and totally helpless patient with no ability to communicate, eat or swallow. Almost all FS have a grade of 4 or higher.
- **EDSS 10:** Death due to MS.

The EDSS has been used for about two decades and has proven to be useful in assessing the progression of MS, but it has some limitations. This clinical rating scale can evaluate reliability, validity and responsiveness [40]. The application of EDSS highly depends on the neurologist's interpretation; therefore, its evaluation may differ among physicians. Moreover, at different times, even the same neurologist may obtain different results [39]. Some studies [40, 41] show that the EDSS scale is also not sensitive to significant clinical changes in short periods, i.e. it is non-responsive. Regarding validity, this scale proves to be effective in assessing disability and impairment [40].

Additionally, it is important to note that the EDSS scale considers significant motor function impairments but is insensitive to other affected functions, such as cognitive ability, which is rarely analysed in routine evaluation. This limitation significantly impacts this work since cognitive impairment is as disabling as physical disabilities, and it isn't considered. Finally, the nature of the symptoms evaluated and the time spent in each step of the EDSS scale are non-linear. Thus, the disability analysed varies between stages, and the difference between stages is not homogeneous [39].

2.1.4 Courses

The clinical courses of MS were defined in 1996 [42] to standardise the terminology used in clinical practice and communication among clinicians and to unify the advances made in clinical research.

The first formally defined MS phenotypes are described below and represented in Figure 2.3:

- **Relapse Remitting (RR):** The RR course is marked by episodes of relapses or exacerbations, with new symptoms or worsening of existing symptoms, followed by periods of remission without progression of the disease [4, 42]. It is the most common form of MS, affecting about 85% of MS patients [43].
- **Secondary Progressive (SP):** Patients diagnosed with RR may transition to the SP course. In this course, there is disease progression and consequent accumulation of disability over time, with or without periods of remission [4, 42].
- **Primary Progressive (PP):** PP course is characterised by continuous and gradual disease progression from the onset of symptoms. Although there are no relapses or remissions, there may be plateaus and occasional minor improvements [4, 42]. It affects approximately 10% of MS patients [5].
- **Progressive Relapsing (PR):** Like the PP course, the PR course is marked by disease progression from the onset. However, this course has acute relapses, with full or partial recovery, and the periods between relapses have a continuous progression. It affects approximately 5% of MS patients [4, 42].

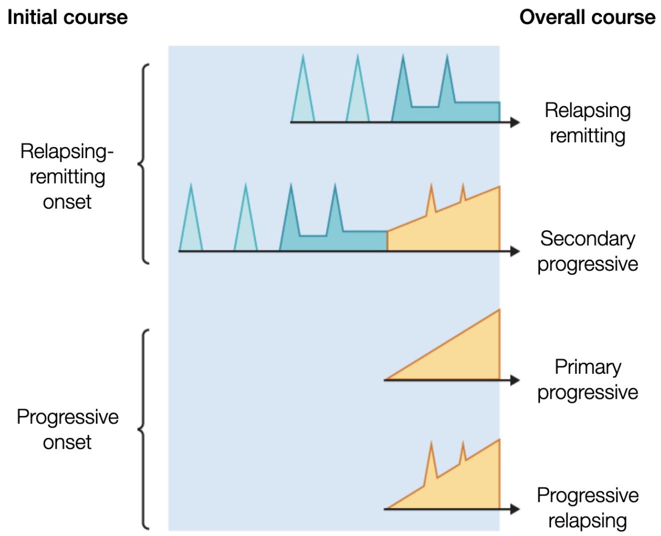


Figure 2.3: 1996 classification of the course of MS, with relapses in blue and disease progression in yellow. Extracted from [4].

Since then, the understanding of the disease and its different phases has increased significantly. Additionally, the above clinical course descriptors proved to be limited because they were purely clinical and based on the subjective views of MS experts. A review of clinical descriptive terminology, MRI and other imaging techniques, and fluid biomarker analysis led to the update of these definitions in

2013 [28].

Since 2013, the phenotypes used to characterise MS are CIS, RR, PP and SP. The course PR was eliminated and the PR patients are now also classified as PP [28]. The introduction of CIS was one of the major differences. CIS is the first inflammatory or demyelinating episode in the CNS that may become MS [5]. The relationship between MS and CIS has been the subject of several studies, concluding that not all people who suffer from CIS develop MS [43]. Radiologically Isolated Syndrome (RIS) may raise suspicion about MS depending on the morphology and location of the lesions detected by MRI. RIS is not considered a phenotype of MS since patients, despite presenting abnormalities suggestive of demyelination, are asymptomatic [28].

MS phenotypes can be classified taking into account the disease activity and progression, as described in Table 2.2. Regarding disease activity, all MS courses (CIS, RR, SP and PP) can be defined as active or not active. On the other hand, disease progression only describes progressive courses (SP and PP), which can be divided into progressive and not progressive [16, 28]. Disease activity and progression should be time-framed, at least annually, to allow the current assessment of the disease and the monitoring of changes over time [44].

Table 2.2: Definitions of disease's activity and progression [16].

Disease's activity and progression		
Disease activity	**Active**	Characterised by relapses or episodes of new or increasing neurological dysfunction followed by full or partial recovery or occurrence of gadolinium-enhancing or new/larger T2 lesions, preferably at least one year.
	Non-active	No evidence of disease's activity.
Disease progression	**Progressive**	Characterised by increasing neurological dysfunction/disability without full recovery, even though there may be phases of stability, preferably at least one year.
	Not Progressive	No evidence of disease worsening, during at least one year.

CIS is a part of the RR MS spectrum and can be classified as active or non-active. Active CIS can be considered RR if it meets McDonald's diagnostic criteria for this state. Otherwise, it is considered non-active until a clinic episode or MRI changes. Moreover, RR can also be characterised as active or non-active according to the clinical relapses and MRI findings in a given period. Regarding the progressive state of the disease, patients can be diagnosed with SP status if they have an initial relapsing stage followed by a progressive stage, or PP status if they have a progressive stage from the beginning [16, 28]. Therefore, SP and PP have four possible sub-classifications:

1. **Active with progression:** The patient is gradually worsening and had relapses.

2. **Active without progression:** The patient has relapses but the condition state is stable and not worsening;
3. **Not active but with progression:** The patient don't have relapses but the state is gradually worsening.
4. **Not active without progression:** The patient has a stable form of MS.

Figure 2.4 shows the possible phases of each MS course over time, after the 2013 revision. It is possible to interpret how the deficiency increases gradually through time and analyse the different particularities of each course.

In this master thesis, the focus is on predicting whether a patient will move from the RR course to the SP form of the disease since SP is considered an evolution from RR [45].

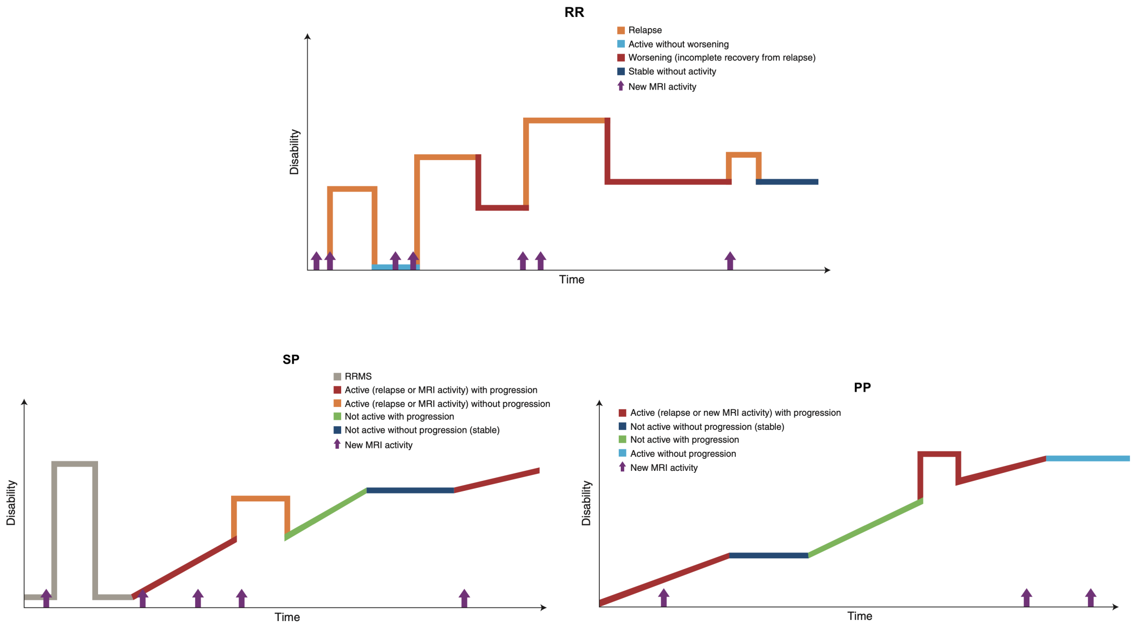


Figure 2.4: MS courses after 2013 revision. Extracted from [5].

The MS disease course is characterised by a wide range of progression rates [46]. There is a subgroup of MS patients who have little or no progression of disease severity over time and minimal disability at least ten years after diagnosis. The terms benign and malignant have started to be used to describe this course of the disease [30, 47]. These terms give an indication of disease severity over time, but their application in clinical practice generates debate [28]. It is now known that MS is rarely a benign disease and that neurological disability is not correctly defined by the EDSS scale. Moreover, this diagnosis is a retrospective determination that may be erroneous since the severity and activity of MS can change significantly even after decades of apparent stability. Thus, this clinical designation should be used with caution in clinical practice [16, 28, 30].

Furthermore, it is important to clarify the difference between the terms worsening and progression. The term worsening should be used to describe patients whose disease is progressing as a result of frequent relapses or incomplete recovery. On the other hand, the term progression refers to patients with a progressive disease with evidence of gradual worsening over time [16].

2.1.5 Therapies

Although MS is a disease with no cure, several treatments are available to manage the course of the disease [31]. Currently, the treatment of MS is multidisciplinary, and concerns disease-modifying therapies, treatment of acute attacks, improvement of symptoms, and rehabilitation [17, 48]. The different alternatives of these treatments depend on the clinical situation of the patient [49], and it should be initiated after the first attack of MS or after diagnosis since studies show that the administration of high-efficacy therapy from the initial phase leads to better long-term results [17]. The first treatment decision clinicians make can be divided into two approaches. The escalation approach is more appropriate for patients with light or moderately active disease. It starts with a first-line treatment and is switched to a second-line treatment if it has an unsatisfactory response. On the other hand, the induction approach begins with an efficient second-line treatment to achieve rapid remission in cases of a very active disease [50].

Disease-modifying teraphies (DMTs) modify the course of the disease by modulating or suppressing immune function. Treatment with DMTs decreases the frequency and severity of relapses, prevents CNS damage, reduces MRI lesion accumulation, and delays disability [17, 21].

Table 2.3 presents the characteristics and information of several DMTs approved by Food and Drug Administration (FDA) [17]. There are other drug options that, although not yet FDA-approved, are used in the treatment of MS by physicians [31]. The physician must choose the medication that best suits the patient's clinical condition since several medications have different objectives, and each patient reacts differently to the treatment. In the case of a RR patient, the goal is to reduce the frequency and severity of relapses and postpone the progressive phase of the disease, while for a SP patient, the goal is to prevent the progressive worsening [49].

Comorbidities (e.g., psychiatric and cardiovascular) and daily behaviours (e.g., smoking) are associated with increased disability, MRI changes and decreased quality of life [21]. There is no high-quality evidence to support the improvement of disease status with healthy nutrition, and vitamin D supplementation [21, 31]. However, patients should remain active, do activities that stimulate cognitive and physical function, and adopt a healthy lifestyle to relieve symptoms, promote a satisfactory quality of life, reduce comorbidities, and improve disease outcomes [31].

Table 2.3: Summary of Approved Disease-Modifying Therapies used in MS treatment [17].

Name	Indication and line of therapy	Administration	Action	Adverse effects
Ocrelizumab	RR and PP First line	Intravenous (IV) infusion, every 6 months	Reduction in annualised relapse rate (ARR) and confirmed disability progression (CDP)	Infusion-related reaction, nasopharyngitis, headache, upper respiratory tract infection, urinary tract infection, and oral herpes infection
Ofatumumab	RR First line	Subcutaneous (SC) injection, every 4 weeks	Reduction in ARR	Injection-related reaction, nasopharyngitis, headache, upper respiratory tract infection, and urinary tract infection
Natalizumab	RR Second line	IV infusion, every 4 weeks	Reduction in ARR and CDP	Fatigue and allergic reaction
Alemtuzumab	RR First line	IV infusion, once daily	Reduction in ARR	Headache, rash, nausea, and pyrexia
Mitoxantrone	RR and SP Second or third line	IV infusion, every month or 3 months	Reduction in relapses	Dose-related cardiomyopathy and promyelocytic leukemia
Fingolimod	RR Second line	Oral, once daily	Reduction in ARR	Bradycardia, atrioventricular conduction block, macular edema, elevated liver-enzyme levels, and mild hypertension
Siponimod	CIS, RR and active SP First line	Oral, once daily	Reduction in CDP	Headache, nasopharyngitis, urinary tract infection, and falls
Ozanimod	CIS, RR and active SP First line	Oral, once daily	Reduction in ARR	Headache and elevated liver aminotransferase
Dimethyl fumarate and diroximel fumarate	RR First line	Oral, twice daily	Reduction in ARR	Flushing, diarrhea, nausea, upper abdominal pain, decreased lymphocyte counts and elevated liver aminotransferase
Cladribine	RR Second or third line	Oral, 4-5 days over 2-week treatment courses	Reduction in ARR	Headache, lymphocytopenia, nasopharyngitis, upper respiratory tract infection and nausea
Teriflunomide	RR First line	Oral, once daily	Reduction in ARR	Nasopharyngitis, headache, diarrhea and alanine aminotransferase increase
Glatiramer acetate	RR First line	SC injection, once daily or 3 times weekly	Reduction in ARR	Injection-site reactions
Rebif (IFN-β-1a)	CIS and RR First line	SC injection, 3 times weekly	Reduction in ARR	Injection-site inflammation, flu-like symptoms, rhinitis, and headache
Avonex (IFN-β-1a)	CIS and RR First line	Intramuscular (IM) injection, once weekly	Reduction in CDP	Flu-like symptoms, muscle aches, asthenia, chills, and fever
Plegridy (PegIFN-β-1a)	CIS and RR First line	SC injection, every 2 weeks	Reduction in ARR	Injection-site erythema, influenza-like illness, pyrexia, and headache
Betaseron (IFN-β-1b)	CIS and RR First line	SC injection, every other day	Reduction in ARR	Lymphopenia, flu-like symptoms and injection-site reactions

Despite all these advances, it is necessary to look for new options to improve the treatment of MS. There are very effective therapies that completely control the relapsing disease. In contrast, progression treatment needs to become more effective because current therapies only partially protect against the neurodegenerative component of MS [17, 51].

2.2 Machine Learning

ML is a branch of Artificial Intelligence (AI) whose goal is to mimic human intelligence by learning from available data. To this end, mathematical models are built based on samples that allow machines to make predictions or decisions without specific computer programming. Several solutions are available, and the choice depends on the type of problem to solve. This high diversity allows ML models to be applied in several fields and to solve complex and different challenges [52].

ML algorithms are divided into three primary categories according to the type of data used for learning. In supervised learning, the input and output samples are known, and the algorithms learn to predict the output from the input data. In unsupervised learning, the data is unlabelled, and the algorithms learn from the internal structure of the input data. Semi-supervised learning combines the previous two, i.e. only a portion of the input data is labelled, and it is used to infer the unlabelled part [6].

The Figure 2.5 links together the main stages of a ML workflow. After acquisition, pre-processing and transformation, the dataset is divided into training and testing sets. The ML algorithm learns from the patterns in the training set, and this learning is applied to the testing set for prediction or classification. Finally, the model is evaluated using performance metrics [9].

This thesis focuses on supervised learning algorithms, in which labelled clinical data from patient history, i.e., collected during periodic visits, is used to generate models capable of predicting disease outcomes. This outcome is whether a patient will transition from the initial RR course to the SP form of the disease.

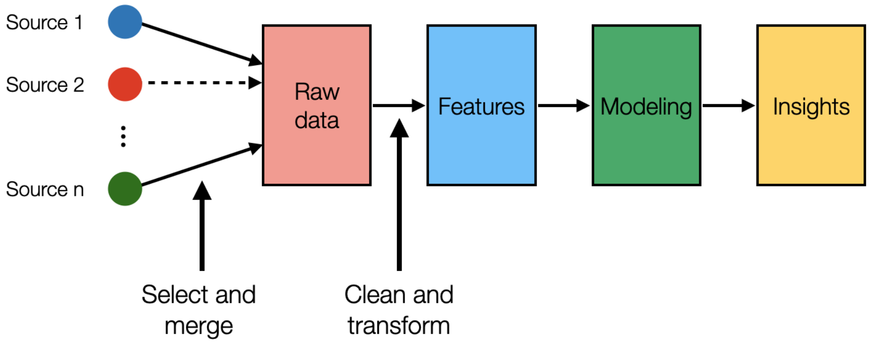


Figure 2.5: ML workflow. Adapted from [6].

2.2.1 Data preparation

Since ML algorithms learn to map input variables to output variables, the data quality influences the model performance. Data preparation is the transformation of the raw data to meet the requirements of the ML algorithms used and can be one of the most challenging steps in a ML project. The data used in the ML model should have only the most relevant and non-redundant features, so the raw data should be processed according to the problem defined beforehand [53]. This process of deriving new variables that best represent the problem from the available data is called Feature Engineering [54].

In clinical problems, the raw medical data can be medical notes, clinical lab reports, clinical images, and information from medical devices. Processing these data involves significant effort to ensure that they have the desired structure and accurately reflect clinical reality [55].

2.2.1.1 Data Cleaning

Data cleaning is usually the first step and involves denoising, identifying and correcting errors or missing values. First, it is important to identify and remove columns with the same value and duplicate rows. Moreover, model results can usually be improved after identifying and removing outliers, which are data that differ dramatically from all others [53, 56].

Regarding the existence of missing values, solving this problem is very important because most algorithms require that all samples have values for all features. The simplest approach is to eliminate the samples with missing values, which causes a loss of information that can lead to biased results. Thus, an alternative is to impute the missing values from the existing information. To deal with missing data, one must first identify the nature of the data and the mechanism leading to its lack [57, 58]:

- Missing Completely at Random (MCAR): missing data does not depend on observed or missing data.
- Missing at Random (MAR): missing data depends on observed data and does not depend on unobserved data.
- Missing Not at Random (MNAR): missing data depends on something unobserved.

There are several methods for imputing missing values using statistics or learning models. Statistical methods are the simplest approach and involve calculating a missing value from the values present. Usually, the calculated value is the mean, median, or mode of the column, and this value replaces the missing values of that column. Data imputation can also be done using models that predict missing values from all other input characteristics. For example, the k-nearest neighbours (KNN) imputation model predicts the missing value from the k nearest neighbours. Another but more complex approach is the development of iterative models. In this case, the model iteratively predicts a missing value from all features, including previously estimated and imputed values [53, 57].

2.2.1.2 Data Transformation

Data transformation is the alteration of the data type to make it suitable for the algorithm. In ML models, all input and output variables must be numeric and, consequently, categorical data (ordinal, nominal and boolean) must be encoded to numbers. Scaling data to a standard range is an essential step in pre-processing since differences between input variables scale can complicate the problem's modelling in many ML algorithms, which leads to poor performance and higher generalisation error. The two widely used techniques for scaling numerical data are [53]:

- **Normalisation:** It is the scaling of the data to a range from 0 to 1. Each value is normalised using the equation 2.1, where x is an original value and y is the normalised one. Knowing the minimum and maximum values of each feature of the training data is essential. The new data is normalised using these values.

$$y = \frac{x - min(x)}{max(x) - min(x)} \tag{2.1}$$

- **Standardisation:** It is the scaling of the data to a Gaussian distribution, i.e., with mean 0 and standard deviation 1. To apply the equation 2.2, where x is an original value and y is the standardised value, it is necessary to estimate the mean and standard deviation of each feature of the training data.

$$y = \frac{x - mean(x)}{standard\_deviation(x)} \tag{2.2}$$

2.2.1.3 Dimensionality Reduction

The dimensionality of a problem is the number of features in the input data. High dimensionality and a small number of examples can lead to problems in most algorithms, as the samples become too sparse and not representative of the space (curse of dimensionality) [53, 59]. Moreover, irrelevant and redundant variables can cause learning errors, leading to inferior performance. Removing these features has benefits such as increased knowledge of the data, identification of irrelevant variables, more efficient learning algorithms, and improved generalisability [60].

Thus, dimensionality reduction is important in eliminating irrelevant data, increasing model accuracy and improving the results' interpretation [61]. These algorithms can be divided into two distinct groups: feature extraction, which consists in creating new features from the existing ones in the input dataset, and feature selection, which involves choosing a subset of features and excluding the rest [62].

Feature Extraction

Feature extraction algorithms transform the input data into a lower dimensionality subspace by generating new features containing the most relevant information from the input dataset [53, 59]. The most popular techniques are:

- **Principal Component Analysis (PCA):** PCA is a non-supervised method that applies an orthogonal transformation to convert the original correlated variables into a set of non-linearly correlated variables called principal components. The number of components is less than or equal to the number of original variables. The principal components are calculated in descending order of importance, i.e., the first principal component has the highest possible variance, and the remaining ones have a successively lower variance [59].
- **Linear Discriminant Analysis (LDA):** LDA is a supervised algorithm that performs a linear transformation of the data. It optimises the separability of the data, i.e., maximises the inter-class distance and minimises the intra-class distance [59]. By other words, it maximises the Fisher criterion given by the equation 2.3, where S_B represents the between-class scatter matrix, and S_W represents the within-class scatter matrix. The direction W that maximises $J(W)$ is given by 2.4, and μ_1 and μ_2 are the mean vectors of the two classes [63].

$$J(W) = \frac{W^T S_B W}{W^T S_W W} \tag{2.3}$$

$$W = {S_B}^{-1}(\mu_1 - \mu_2) \tag{2.4}$$

Feature Selection

Feature selection techniques allow the selection of a subset of the most relevant input features for the problem. These techniques are divided into supervised and unsupervised, depending on whether or not they consider the target variable, respectively [64]. Supervised techniques can be further divided into three groups:

- **Filter methods:** This class of methods uses variable ranking techniques to evaluate the relationship between the input and the target variable. Each variable is assigned a score, and the ones with higher ratings are chosen, while those below a defined threshold are removed. This selection method is independent of the learning algorithm and ignores interactions between features, relying only on statistical information in the data, such as correlation, distance metrics, and consistency metrics. The filter method is computationally less demanding than the others and is preferable in high computational cost problems with large datasets. Examples of filtering methods include ANOVA, Pearson correlation, and the chi-square test [64].
- **Wrapper methods:** In this class of methods, several models are created with different subsets of features, and the selection criterion is the performance of the classifier. The selected features are the subset that leads to better performance. This method performs better than filter methods. However, it is computationally expensive, especially for large numbers of features, and may lead to overfitting. Another disadvantage is the dependence on a giving classification method. Wrapper methods can be divided into Sequential Selection Algorithms and Heuristic Search Algorithms. Sequential selection algorithms, such as Sequential Feature Selection (SFS) and Sequential Backward Selection (SBS), start with an empty or complete set of features and add or remove features, respectively until the maximum objective function is obtained. Heuristic search algorithms evaluate different subsets of features to optimise the objective function [60, 64].
- **Embedded methods:** This class of methods includes feature selection in the model fitting process, filling the gap between the filter and wrapper models. Embedded methods select multiple subsets of features during the learning process and choose the one with the best performance according to a performance metric. It is much less computationally heavy than the wrapper methods and includes the interactions with the classification model. This method includes decision trees and regularisation algorithms, such as the Least Absolute Shrinkage and Selection Operator (LASSO) and Ridge regression models. Regularisation models add penalties to different model parameters to avoid overfitting [64].

2.2.2 Classification

After pre-processing the data, the following steps include developing the classification model. It is important to analyse the problems of the dataset under study and define strategies to prevent the model from being affected by its predictions. The goal is to design a robust model to ensure reliable results. In imbalanced data, sampling techniques allow overcoming the difference between classes. To evaluate the algorithm performance it is essential to partition the data using resampling techniques. Ensemble methods can also be applied to get the most out of algorithms with good performance. In addition, several classification methods can be applied, particularly the most promising ones for the problem. Lastly, some techniques are available to find the best hyperparameters for the studied classifiers.

2.2.2.1 Sampling Methods

One of the biggest challenges in classification problems is class imbalance. In these cases, the classifier is biased towards the majority class, and this imbalance becomes more relevant when the class of interest is the minority class [65]. MS classification problems usually have balancing problems between the disease stages because MS has a slow course, which results in more records in the RR course than in the SP and PP courses [18]. To improve the classification performance, sampling techniques can be applied. These techniques change the distribution of the classes so that the data is relatively balanced [65].

Sampling approaches have been proposed, including random undersampling and random oversampling. Random undersampling involves randomly removing samples from the majority class to balance the number of examples from each class. The major disadvantage of undersampling is eliminating large amounts of data and losing information that may be relevant to classification performance. Contrarily, in random oversampling, the class balancing is done through random repetition of samples from the minority class. In this case, the major problem is the deficiency in the generalisation ability of the classifier due to overfitting [65, 66].

More advanced methods based on these simple techniques have been developed to overcome these limitations. Some of these methods use intelligence to add or remove samples or combine oversampling or undersampling to decrease information loss and avoid overfitting [65].

2.2.2.2 Partition Methods

The evaluation of the learning algorithm performance must be performed on new data, usually called test data. Otherwise, the model might be overfitted to the training data. Therefore, the dataset must be at least partitioned into two parts, one for building the model and learning, usually called the training set, and one for testing the model, called the test set [7, 67]. The split size depends on the data, but it is common to use 67% of the data for training, and the remaining 33% for testing [67].

Another alternative is the Cross-Validation (CV) technique, in which the dataset is divided into k parts, and k iterations are performed, using each time one of the k parts for testing and the rest for training, as shown in Figure 2.6. The classifier performance is given by the mean and standard deviation of the test results over k iterations. There is no rule for choosing the value of k, but it is usually 5, or 10 [7]. For imbalanced problems, Stratified CV is an alternative that ensures that each part keeps the proportion of each class in the complete data. Another variant of this method is Leave One Out Cross Validation (LOOCV), in which the number of k folds is equal to the number of instances in the data set. Overall, CV methods lead to more reliable results than train/test split since the algorithm is trained and evaluated multiple times on different data [7, 67].

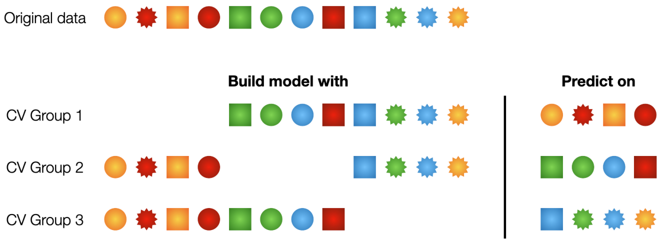


Figure 2.6: Scheme of CV with k=3. Adapted from [7].

A different approach is the Repeated Random Training/Test Splits technique presented in Figure 2.7, in which the train/test split process is repeated several times. The proportion of the data split is variable and influences the number of repetitions, i.e., the higher the percentage of training, the higher the number of repetitions should be to get stable estimates [7].

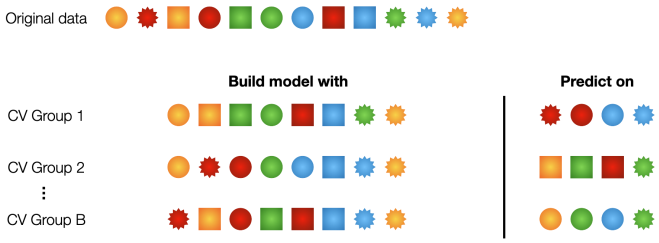


Figure 2.7: Scheme of B repeated training and test set splits. Adapted from [7].

The Bootstrap is another approach, which consists of several resamples of the same size as the original sample. Bootstrap samples are random and built with replacement, i.e., the training sample may have repeated observations as observed in Figure 2.8. Bootstrap error estimation performs well with small samples because it has a smaller variance but demands a higher computational cost [7, 68].

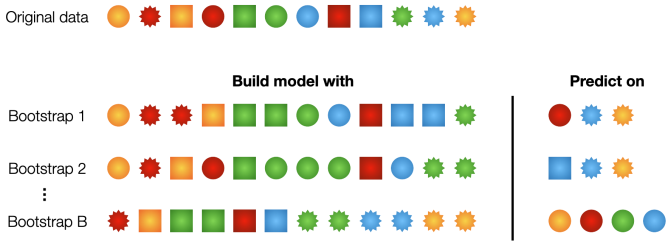


Figure 2.8: Scheme of bootstrap resampling with B subsets. Adapted from [7].

2.2.2.3 Classifiers

In supervised ML problems, the classification algorithms learn from the input data and optimise the learning for a given labelled output. Several classification methods use different learning approaches, and the most appropriate choice depends on the problem under analysis and the dataset [6]. The most well-known classification methods are the following:

1. **Logistic regression (LR):** LR is a statistical model used in classification problems that predicts the probability of a given outcome by fitting data to a logistic function. The logistic function estimates the probability $P(x_i)$ associated with the occurrence of an event, given the input variable x_i. This probability is a number between 0 and 1, which is given by the equation 2.5. The values α and β_i are unknown coefficients that can be obtained by the Maximum Likelihood Estimation (MLE) method. In binary problems, if the output is greater than 0.5, the sample is assigned to the positive class, and if

it is less than 0.5, it is assigned to the negative class [69].

$$P(\mathbf{x}) = \frac{1}{1 + e^{-(\alpha + \sum \beta_i x_i)}} \tag{2.5}$$

LR has the advantage over other algorithms of being an interpretable model that is relatively quick and easy to set up, despite being too simplistic for complex relationships between variables. It also tends to have lower performance on nonlinear problems [70].

2. ***k*-nearest neighbours (KNN):** KNN assigns the class to a sample that is the predominant one among the nearest neighbours, as shown in Figure 2.9. The value k refers to the number of nearest neighbours that the classifier will search to make the prediction. There are several metrics to calculate the distance, and their choice varies according to the problem. The most used is the Euclidean distance [9, 71].

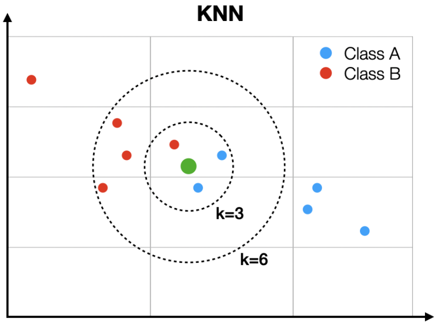


Figure 2.9: The KNN classifier. Adapted from [8].

Compared to other classifiers, KNN has the advantage of being simple to implement and easy to understand the classification result's explanation. It is a lazy learning algorithm; therefore, it memorises the training data and uses it to make predictions. It is very sensitive to the choice of parameter k and noisy data and does not work well on large datasets, and high dimensional data [70, 72].

3. **Decision tree (DT):** DTs are a hierarchical model where an unknown pattern is classified into a class using decision functions in successive steps. This sequence of recursive partitions allows simpler problems to be solved with fewer steps and features. As represented in Figure 2.10, this algorithm has a tree-like structure: nodes, branches, and leaves. The process starts at the root and at each node, after the test result, a branch is taken that leads to one of the child nodes. This procedure is repeated until a leaf is reached,

corresponding to the class result. In the case of binary decision trees, each node splits into only two branches, and the logical test is always interpreted as true (left) or false (right) [6, 9].

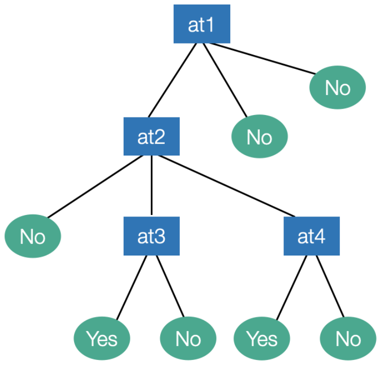


Figure 2.10: A Sample Decision Tree. Adapted from [9].

DTs have high interpretability and are easily understood by the schematic visualisation of a tree. In addition, this classifier is robust to noisy data and supports nonlinearity. On the other hand, it is prone to overfitting and is not as accurate as other classification methods [70].

4. **Naive Bayes (NB):** The NB is a classifier based on probabilistic knowledge and Bayes' theorem. This algorithm assumes that the attributes are independent of each other, given any known class. Although this assumption seems naive and simplistic, the classifier is successfully applied to multiple complex problems [73, 74].

   In training, the classifier estimates the probability distribution given the class, while in classification, the method calculates the posterior probability for each class. The test data $\hat{y}$ is classified according to the Maximum *a posteriori* (MAP) rule, which selects the class with maximum posterior probability using the equation 2.6 [73].

$$\hat{y} = \underset{y \in 1 \ldots k}{argmax} \prod_{i=1}^{n} p(x_i \mid y) P(y) \tag{2.6}$$

   where k is the number of possible classes, n is the number of attributes, $p(x_i \mid y)$ is the probability of x_i given the class y, and $P(y)$ is the prior probability of class y. The NB classifier is relatively simple, fast, easy to interpret, and robust to noise and irrelevant attributes. It works well with high dimensional input data and requires little training data. On the other hand, a limitation of NB is the assumption of independent predictors, which in real life is very uncommon. This classifier also performs relatively poorly on low dimensional data [70].

5. **Support Vector Machines (SVM):** A SVM is a binary classifier defined by a separation hyperplane that divides the feature space into two parts, and optimally each class is on different sides. For the hard-margin SVM formulation, the margin is the distance between the nearest training samples (support vectors) that belong to the different classes, as can be seen in Figure 2.12. The goal is to select the hyperplane that maximise this margin since a boundary further away from the training data will minimise the generalisation error [6, 10]. The margin is influenced by parameter C, also known as the regularisation parameter, which controls the penalty of misclassifications. This penalty is lower for lower values of C, and consequently, the margin is larger. In contrast, misclassifications are more penalised for higher values of C, and the margin is smaller [10, 75].

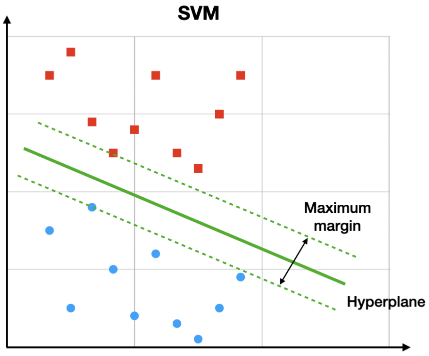


Figure 2.11: Maximum margin through SVM. Adapted from [10].

A large proportion of problems involve data not linearly separable. In these cases, the data is transformed into a higher-dimensional space using a kernel function, where it is separable by a hyperplane. The Radial Basis Function (RBF) kernel is widely used; in this case, the parameters C and gamma are set. The gamma parameter controls the distance of the influence from a training point. For low values, the influence is broader, while for higher values, the influence is more localised, and consequently, the points have to be closer to be classified in the same class [63, 75]. On the other hand, when the problem involves more than two classes, it is divided into several binary classifications [9].

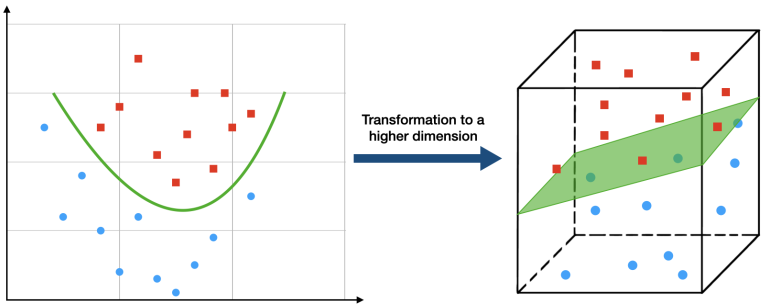


Figure 2.12: Non-linear SVM classification using kernel functions. Adapted from [10].

SVM is a robust and complex model that is considered one of the best-performing classification algorithms. The kernel function allows varying degrees of non-linearity and flexibility in the model. The main disadvantages of the SVM model are that it requires a lot of memory and processing power and that it is difficult to interpret [70].

6. **Artificial Neural Network (ANN):** ANNs are a computational model based on the connectivity of neurons in the human brain. The processing is done by neurons organised in input, hidden or output layers. This process needs an activation function, which is the mathematical function that defines and adjusts the weights along the learning process [71]. Several ANNs can be used to classify different problems for different purposes. Deep Learning is based on ANNs with many hidden layers [9].

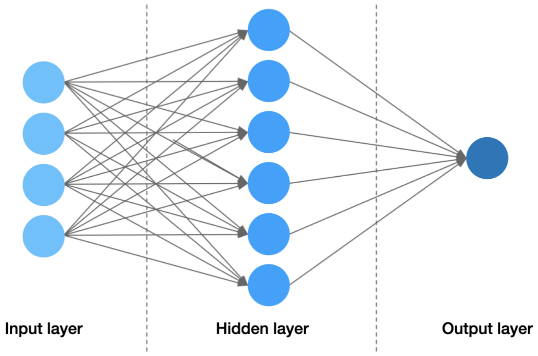


Figure 2.13: ANN architecture. Adapted from [9].

ANNs can detect complex non-linear relationships between independent and dependent variables, and possible interactions between predictor variables. In addition, several types of ANN methods can be used to train and extract knowledge. The biggest limitation of ANNs, especially in the medical field, is that they are a black-box model, and it is not always possible to identify possible data relationships that led to the solution. Training ANNs requires greater computational resources and is very time consuming, especially for complete data and networks with many hidden layers. Furthermore, they are prone to the problem of overfitting [70, 76].

The Feed-Forward Neural Network (FFNN) is the simplest Neural Network (NN) since the information is processed in a single direction (forward), and there are no cycles in the network. The ANN in Figure 2.13 represents the architecture of a FFNN with one hidden layer [77].

The opposite of a FFNN is a Recurrent Neural Network (RNN). The latter have cycles that allow information to persist. In this project, it was used a type of RNN, the Long Short-Term Memory (LSTM). These networks can learn long-term dependencies, making them popular in problems with sequential data [78]. A LSTM layer comprises connected memory cells, whose structure is represented in Figure 2.14. In this figure, h_{t-1} is the hidden state at previous timestep (short-term memory), c_{t-1} is the cell state at previous timestep (long-term memory), and x_t, h_t and c_t are the input vector, hidden state and cell state, respectively, at current timestep [11].

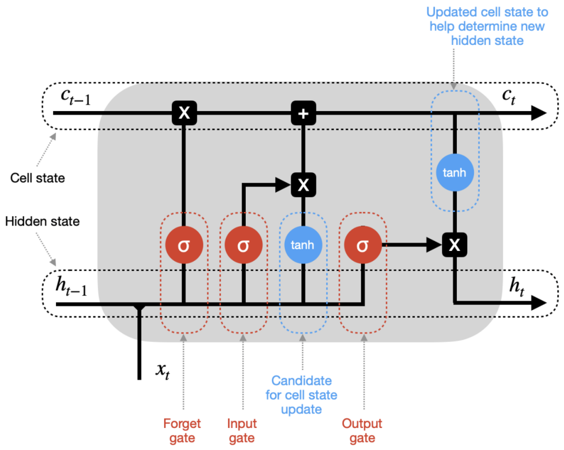


Figure 2.14: LSTM recurrent unit. Extracted from [11].

The memory unit has the cell state, which is the memory of the LSTM, and power units (gates) that regulate the flow of information in and out of memory: the forget gate controls the removal of information; the input gate controls the input of important new information; the output gate controls the information from memory that is added to the cell state. The mathematical functions define the behaviour of the cell [11, 79].

2.2.2.4 Ensemble Methods

Ensemble methods arose from the need to have techniques with strong generalisation capability. The goal of ensemble methods is to combine several learning algorithms to make the generalisation capability stronger [65]. Hence, several base learners are built from the original data, and the predictions from each one are combined [66].

Building a robust single model with good performance is a challenge, and this combination can overcome these difficulties. Thus, although ensemble models considerably increase the complexity and computational cost, they usually lead to algorithm improvements, such as increased stability, better classification performance, and reduction of the variance and bias of the model [8, 66]. Depending on how the learners are constructed, as well as how they are combined, ensemble learning methods can be divided into two types:

- **Boosting:** The main idea of Boosting is to generate several weak learners (low performing) and combine them into a single strong learner. To this end, a sequence of learners is created, and each learner tries to correct the mistakes of the previous learners in the sequence. [8, 67]. The algorithm assigns different weights to each training sample so that the samples that are harder to classify have a higher weight than the easier ones. Thus, each base learner is trained with the training samples adjusted by the output of the previous learner. The most popular Boosting algorithm is AdaBoost [8].
- **Bagging:** The Bootstrap aggregation (Bagging) algorithm involves the creation of individual and parallel base learners. This technique uses bootstrap sampling to include randomness in the training data and generate learners with less dependency, and more diversity [8, 65]. Therefore, several random samples with replacements are generated from the training group, and a base learner is built for each one. The final learner is built from the individual base learners, and the final prediction is the class most chosen by the sub-models [8].

2.2.2.5 Hyperparameter Optimisation

Hyperparameters are ML algorithm variables defined during training, whose value influences the performance of the models. Hence, it is fundamental to search for the best and most robust combination of parameters for a given problem. This search is called hyperparameter optimisation [67, 80]. They can be adjusted manually by trial and error, but there are faster and more automatic methods to optimise them. The two most widely used techniques are:

- **Grid-search:** It starts with a set of values for each hyperparameter, and a model is built and evaluated for each combination of hyperparameters. The parameters chosen are those that lead to better model performance. This method has the disadvantage of being computationally heavy due to the exhaustive search. Additionally, the number of models evaluated increases exponentially with the dimensionality of the hyperparameter space [80].
- **Random-search:** Random search is an approach that tests random combinations of hyperparameters for a fixed number of values. Like in grid-search, a model is built and evaluated for each combination of values. This method works better than grid-search when some hyperparameters are much more important than others [80].

2.2.3 Performance evaluation

When developing classification models, it is important to perform an evaluation using metrics that are adequate for the problem. Their value reflects the quality of the model and its efficiency in meeting the required requirements [67, 81].

Regarding binary classification problems, the two classes are often called positive and negative. The confusion matrix is a table that allows the calculation of most of the evaluation metrics for binary problems and has the following values [82]:

- **True Positives (TP):** samples correctly classified as positive;
- **True Negatives (TN):** samples correctly classified as negative;
- **False Positives (FP):** samples classified as positive that are negative;
- **False Negatives (FN):** samples classified as negative that are positive.

Table 2.4: Confusion Matrix for Binary Classification [8].

	Actual Positive Class	**Actual Negative Class**
Predicted Positive Class	TP	FP
Predicted Negative Class	FN	TN

The metrics for classification evaluations often computed from a confusion matrix are [82, 83]:

1. **Accuracy:** Accuracy is given by the ratio between the number of correct predictions and the total number of evaluated samples. Overall, accuracy is the most widely used metric for evaluating classification models. However, in problems with imbalanced classes, it becomes an unreliable measure of

performance since it assumes high values even when the model is not effective [81].

$$\mathit{Accuracy} = \frac{TP + TN}{TP + FP + TN + FN} \tag{2.7}$$

This measure reflects the total number of correctly classified cases, both RR and SP, but does not distinguish between the numbers of correctly classified samples from each of the classes. Thus, a high accuracy value may be related to a bad classification since the model may be classifying nearly all samples as the majority class (RR course) [84].

2. **Sensitivity:** Sensitivity, also called recall, is the proportion of positive samples that are correctly classified in relation to all positive samples. This is the True Positive Ratio (TPR) and summarises how well the positive class was classified [83].

$$\mathit{Sensitivity} = \frac{TP}{TP + FN} \tag{2.8}$$

In this problem, the sensitivity is the proportion of patients in the SP stage that have a positive result. A high sensitivity means that the classifier correctly identifies patients in the SP course.

3. **Specificity:** Specificity is the proportion of negative samples that are correctly classified in relation to all negative samples. This is the True Negative Ratio (TNR) and summarises how well the negative class was classified [83].

$$\mathit{Specificity} = \frac{TN}{TN + FP} \tag{2.9}$$

In this specific case, the sensitivity is the proportion of people in the RR stage that have a negative result. A high specificity means that the classifier correctly identifies patients in the RR course.

4. **Precision:** Precision is the ratio between correct positive predictions and the total number of positive observations predicted. This metric quantifies the quality of the positive prediction made by the model [83].

$$\mathit{Precision} = \frac{TP}{TP + FP} \tag{2.10}$$

In other words, it is the accuracy for the minority class (SP course) and quantifies the number of positive class predictions that are actually patients in the SP phase.

5. **F1-score:** F1-score is the harmonic mean of sensitivity and precision. This metric is more suitable than accuracy in problems with imbalanced data, such as MS predictions, since in these cases accuracy has high values in classifiers with poor performance for the minority class [82].

$$F1\text{-}score = \frac{2 \times precision \times recall}{precision + recall} \tag{2.11}$$

   Thus, this metric provides important information about the ML models, giving the balance between precision and recall. It is important to note that these two metrics are both important to the problem under study, and F1-score allows a trade-off between them since increasing precision generally leads to a decrease in recall value and vice-versa. By increasing recall, the chance of missing the detection of a SP patient is minimised. However, it increases the possibility of classifying RR patients as SP (FP) who will be subjected to early or more aggressive therapies with potentially troublesome impacts, not only in terms of side effects but also on patient expectations and economic management. On the other hand, precision optimisation leads to the correct prediction of SP patients, i.e., patients classified as positive are very likely to progress to SP. Despite that, the model may more often fail the prediction of SP patients, classifying them as RR (FN). In this case, a patient with worsening disease is inadequately treated [27, 85].

6. **G-mean:** This metric is the geometric mean of the sensitivity and specificity, which is given by the square root of these two metrics. This measure is particularly important in problems with imbalanced data. It allows the combination of sensitivity and specificity into a single value that reflects the balance between classification performances in both the majority and minority classes [82].

$$G\text{-}mean = \sqrt{sensitivity \times specificity} \tag{2.12}$$

7. **Area Under the Curve (AUC):** AUC is the area under the Receiver Operating Characteristic (ROC) curve, as represented in Figure 2.15. This curve is represented by TPR, or *sensitivity*, as a function of False Positive Ratio (FPR), or *1-specificity*, for different threshold values. ROC curve summarises the performance of a binary classification model for the positive class. The AUC clarify the analysis and comparison of different ROC curves. Its value ranges between 0 and 1, and the threshold between classes is 0.5. An ideal model can correctly predict each sample and has AUC=1, while a model with

AUC=0 is inversely identifying the samples [82].

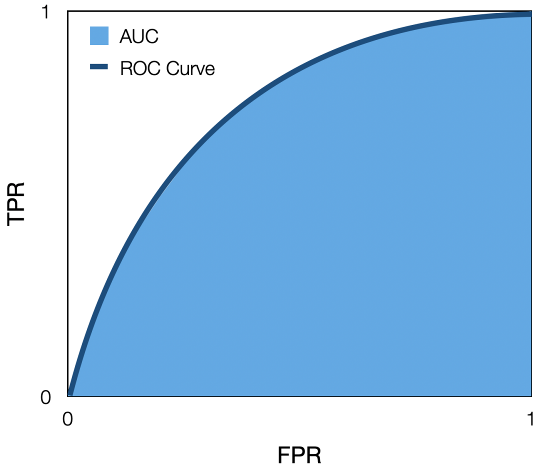


Figure 2.15: Illustration of ROC curve and AUC. Adapted from [8].

2.3 Explainability

Explainability is not a new problem in ML models, but its importance and need have grown with the increasing use of ML models in diverse and complex applications. Clinicians tend to prefer simpler models, such as linear regressions and decision trees, because these are like rule systems and, consequently, are self-explanatory and easily understood by clinicians [86].

The increasing complexity of ML models has led to the black-box problem, since the operation of these models and the reasons for their conclusions are not understood. As a result, there are problems in accepting and trusting the answers provided. There is a growing need for greater transparency in algorithm decision-making [13]. The importance of human interpretability in algorithm design was reinforced in 2018 with the publication of the General Data Protection Regulation (GDPR), which gives citizens the right to receive an explanation of automated decisions [87].

In situations where failures have a significant impact, particularly in medicine, using the performance evaluation metrics of ML models may not be sufficient to describe the problem. It becomes essential to know the reason for the decisions of ML systems [88]. Additionally, it's also important to evaluate other auxiliary criteria such as fairness, privacy, reliability, robustness, causality, usability, and trust [87].

Interpretability refers to the ability of ML models to present their decision logic in a way understandable to humans. Explainability goes a step further, being defined as the ability to summarise the reasons for the model's behaviour, and explain the reasons for the decision in non-technical terms to gain the user's trust [89].

Figure 2.16 illustrates the challenge of making trade-offs between interpretability and model performance since more precise explanations, such as NNs, may be more complex and become complicated for people to interpret; and more interpretable explanations, such as linear regression, may ignore input-output relationships and result in lower performance values [89]. However, it is important to note that the interpretability analysis is affected by aspects other than the algorithms used, such as the input data and model parameters [88].

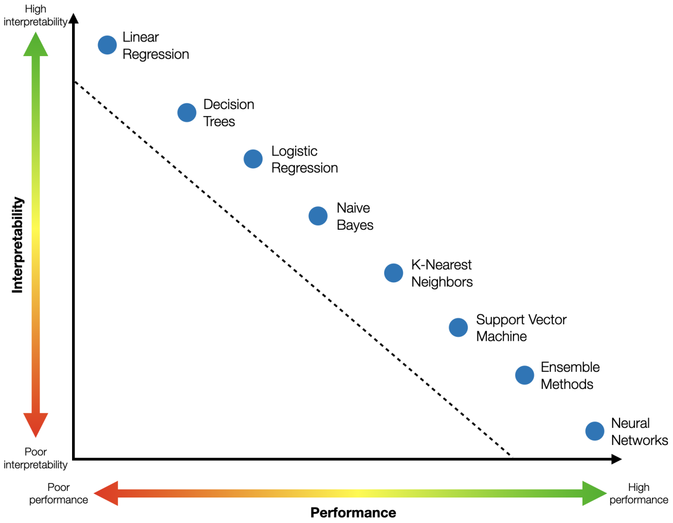


Figure 2.16: Trade-off between interpretability and model performance. Adapted from [12].

2.3.1 Explainability Methods Taxonomy

The taxonomy of explainability is still not clear or recognised, but it is essential to define criteria to allow researchers to compare and evaluate methods. The recent increase in research on the explainability of ML models has resulted in their categorisation according to multiple criteria, such as inherence, specificity, scope, and output [13, 14].

Model interpretability can be considered intrinsic if it results from simple models with constraints imposed by their complexity or post-hoc if the explanation methods analyse the model after its training phase. It is possible to distinguish whether explainability is limited to specific classes of the model, defined as model-specific, or whether it is used in any ML model, being considered model-agnostic. The classification of explainability based on scope distinguishes whether the method is local or global, that is, if it provides an explanation of a specific decision or if it gives an overview of the model [13, 14].

The distinction between explainability methods can also be based on the output obtained. Statistical and visual summaries of the features can be analysed to evaluate their impact on the model predictions. In addition, analysis of the results can provide internal model information, such as the coefficients of linear models and the structure of decision trees. Finally, some models become interpretable because they return new or existing data points that provide visual or textual explanations [14].

2.3.2 Explainability Evaluation

Unlike the performance evaluation metrics of ML models, explainability evaluation is not quantitative, and there is no consensus on the approaches used [14]. However, there are three main levels of evaluation, which are represented in Figure 2.17:

- **Application level evaluation:** It involves real people and real tasks, i.e., it evaluates how human-created explanations help other people complete a task. It provides more accurate results but involves a more costly process [13, 14].
- **Human level evaluation:** It involves real people and simpler tasks. This type of evaluation is an alternative to the previous one because it is difficult to find people who are experts in a particular domain. Thus, by being performed by non-experts, it becomes a cheaper process on a larger scale, which allows testing more general notions of the explanation [13, 14].
- **Function level evaluation:** It involves only proxy tasks without the involvement of people. Thus, the costs and time of the evaluation process are lower. The main challenge is the choice of which proxy to use [13, 14].

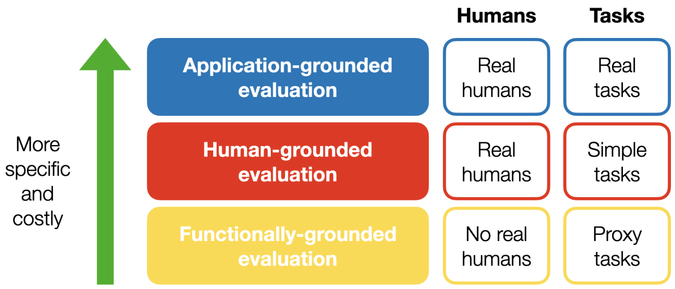


Figure 2.17: Evaluation approaches for explainability proposed by Doshi-Velez et al. [13].

2.3.3 Explainability methods

As mentioned earlier, several explainability methods are distinguished according to multiple criteria and whose application depends significantly on the purpose of the study.

Interpretability can be achieved using merely a subset of algorithms that create interpretable models, such as linear and logistic regression. On the other hand, explainability methods independent of ML models can be used, such as the agnostic models applied after the training phase. This way, greater flexibility, power, and ability to compare models can be achieved. There are also example-based methods whose explanations are obtained by selecting specific instances from the dataset [14].

Finally, specific models for interpreting NNs are also considered. These models allow the discovery of properties of the different layers of NNs, including the hidden layers, which would not be possible with non-specific models [14].

2.3.3.1 Interpretable Models

According to Mohar et al. [14], interpretability is more easily achieved by using algorithms that create interpretable models, such as those shown in Table 2.5, which have three main properties: linearity, monotonicity, and interactions.

In a linear model, like linear regression, the explanatory variables are linearly related to the response variable. Monotonicity is achieved by monotonicity constraints that ensure that the relationship between these variables continuously varies in the same way; that is, it either only increases or only decreases. Regarding interactions, decision trees include interactions between the explanatory variables to predict the response variable's value, making the model more transparent [14].

Table 2.5: Distinction between interpretable algorithms according to the type of task [14].

Algorithm	**Linear**	**Monotone**	**Interaction**	**Task**
Linear regression	✔	✔	✘	Regression
Logistic regression	✘	✔	✘	Regression
Decision tree	✘	Some	✔	Regression and classification
Naive Bayes	✘	✔	✘	Classification
k-nearest neighbours	✘	✘	✘	Regression and classification

2.3.3.2 Model-Agnostic methods

As mentioned in section 2.3.1, agnostic models consider input/output pairs and analyse how changing the input influences the output by creating feature summaries. Some of these methods are:

1. **Partial Dependence Plots (PDPs):** PDP is a global method that gives the contribution of one or two features to the output. Through equation 2.13 it is possible to calculate the partial dependence, where $g(x)$ is the output, x_S is the set of features under study, X_C is the remaining features, and g_S is the expectation of g in the marginal distribution of X_C [90].

$$g_S(x_S) = E_{X_C}[g(x_S,X_C)] = \int g(x_S,X_C)dP(x_C) \tag{2.13}$$

This method is very intuitive, easy to implement, and allows a clear interpretation of the influence of the features on the prediction. However, it has some limitations, such as only showing the average of marginal effects, assuming that features are independent, and having a maximum number of features of only two [14].

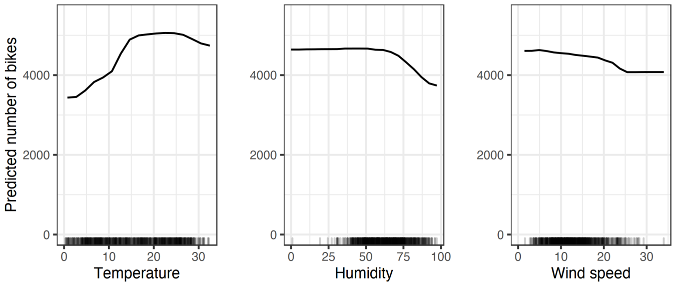


Figure 2.18: PDPs of the bicycle number prediction for the features temperature, humidity and wind speed. Extracted from [14].

In the example given by Mohnar et al. [14] regarding the prediction of the number of bicycles rented daily shown in Figure 2.13, it can be seen that overall the number of bicycles rented increases with temperature and decreases with precipitation and wind.

2. **Individual Conditional Expectation (ICE):** ICE is an extension of the previous method. While PDP shows the average influence of one or two features on the prediction, ICE gives the prediction change for each separate data instance. In other words, PDP shows the average of the curves present in the ICE plots represented in Figure 2.19. This method is more intuitive than the previous one, but it can become overcrowded in the presence of many curves

[14, 90].

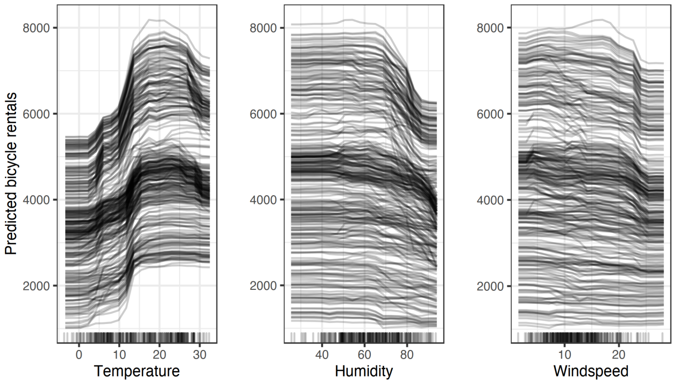


Figure 2.19: ICE plots of the bicycle number prediction for the features temperature, humidity and wind speed. Extracted from [14].

3. **Feature Interaction:** Feature interaction exists when the effect of a given feature on the prediction depends on the value of another feature. This interaction can be evaluated by Friedman's H-statistic metric, which measures the interactions between two features and between one feature and all other features. The H-statistic detects all existing types of interactions but has a very high computational cost [14].

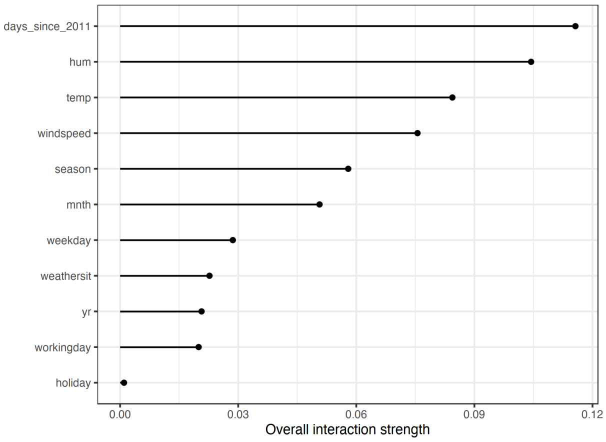


Figure 2.20: Feature interaction H-statistic metric for each feature with all others in the problem of predicting the number of bicycles. Extracted from [14].

Figure 2.20 illustrates the values of the strength of the interaction of each feature with all the others for the previous example and shows that the interaction effects between features are low.

4. **Permutation Feature Importance:** This metric allows the measurement of a feature's importance by permuting it and calculating the model's prediction

error. A feature is more important the higher the model error, and vice-versa. Although its results vary greatly (due to randomness) and can be biased by unrealistic instances, this metric gives an overview of the model's behaviour and allows comparison between different problems [14].

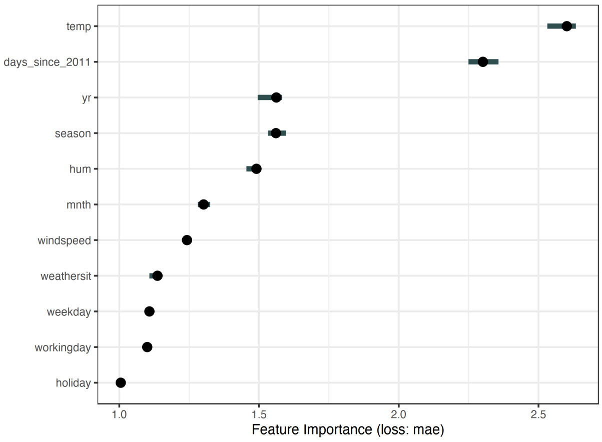


Figure 2.21: Importance of each feature in predicting the number of bicycles. Extracted from [14].

Returning to the bicycle rental example, Figure 2.21 represents the importance of each feature for the prediction problem, showing that the most important feature is temperature.

5. **Surrogate models:** Surrogate models find a simpler approximation for black-box models, which can be considered global or local, depending on whether the explanation is for the entire model or individual predictions.

   For global models, the goal is to create interpretable surrogate models that are as close as possible to the predictions of the base model. In other words, the goal is to approximate as closely as possible the base model's predictive function to the surrogate model's predictive function, with the condition that the latter is interpretable. This method is quite flexible, intuitive and easy to implement. However, it is important to note that the explanations obtained are not about the data but the model [14, 91].

   In the example of Figure 2.22, the surrogate model was trained with a decision tree to approximate the predictions of a SVM. The explanation shows that the model predicts more bicycles rented when the day is further away from 2011 and when the temperature is higher than approximately 13°C.

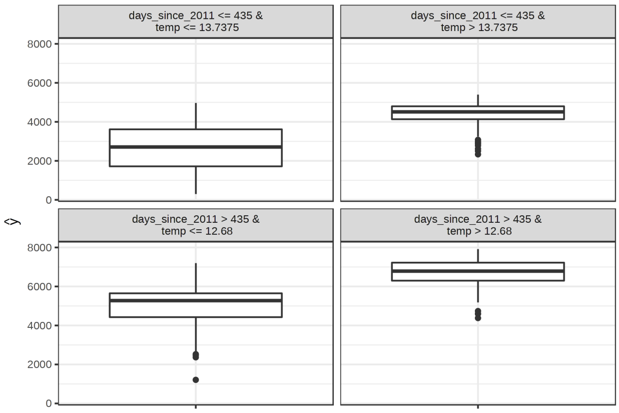


Figure 2.22: Explanations of terminal node predictions from a surrogate model with a decision tree. Extracted from [14].

On the other hand, Local Interpretable Model-Agnostic Explanations (LIME) models create surrogate models that explain the prediction of specific individual samples. This model initially generates a set of perturbed samples from the selected individual sample x and determines their predictions by the black-box model. Next, a weight is assigned to these perturbed samples according to their proximity to x, and the weight increases with proximity. These weights are calculated by a kernel function. Finally, an interpretable model is generated from the weighted sample set, which should be an excellent local approximation of the black-box model. This interpretable model provides explanations about sample x [14, 92].

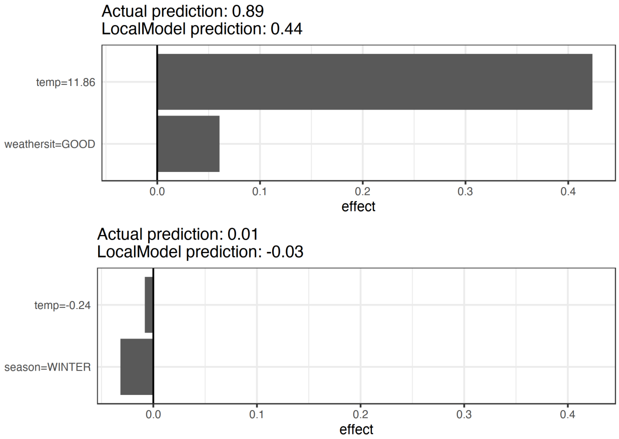


Figure 2.23: LIME explanations for two instances of the problem of predicting the number of bicycles rented. Extracted from [14].

Figure 2.23 represents the LIME explanations obtained for two features in the bicycle problem, showing that the higher temperature and the good weather have a positive impact. Besides working on tabular data, the LIME model can also be applied to text and images. One of the main problems of this model is the instability of the explanations since very close points can have significantly different explanations. [14].

6. **Shapley Values:** This local method is based on a game theory in which payouts are assigned to players according to their contribution to the total payout. In this method, each feature is a player, and the prediction is the payoff. Coalitions between features are also evaluated, and a profit is assigned to these cooperations [14].

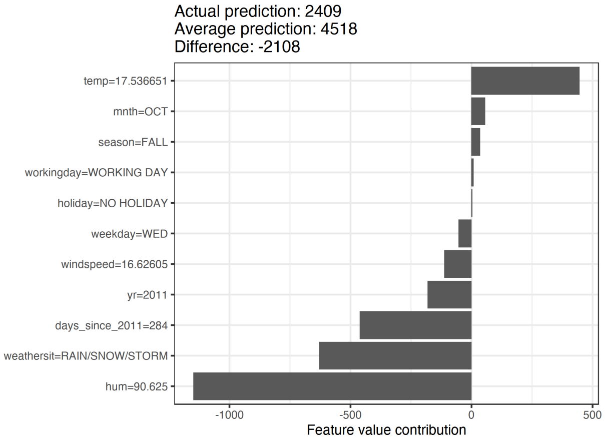


Figure 2.24: Shapley values for a day from the prediction model of the number of bicycles rented daily. Extracted from [14].

The Shapley values in Figure 2.24 show that temperature on that day had the most positive contribution and, conversely, the humidity had the most negative contribution.

2.3.3.3 Example-based methods

While agnostic methods create feature summaries, example-based methods create humanly understandable explanations by selecting instances from the dataset that allow understanding complex data distribution. These methods work best in data with a structure such as images and text. Their application to less structured data, like tabular data with multiple features, is more challenging.

The KNN method, mentioned in Section 4.2.3, is a well-known example-based method that compares the k nearest neighbours to make a prediction. Other example-based methods are:

1. **Counterfactual Explanations:** Counterfactual explanations show how the

prediction changes with the modification of a particular instance. It is based on creating instances representing hypothetical scenarios close to the original, changing as few features as possible [14]. It is usually advantageous to create multiple counterfactual explanations to illustrate the different ways to achieve various desirable outcomes [93].

This method is relatively easy to implement, and its explanations are quite clear. However, its main drawback is the existence of infinite counterfactual explanations for each case. It is only necessary to explore the ones most relevant to the outcome [14].

2. **Prototypes and Criticisms:** The prototypes are the points in the centres that represent the behaviour of the data, while the criticisms are the points in the clusters that are not well represented by the prototypes but also provide insights [14], as shown in Figure 2.25.

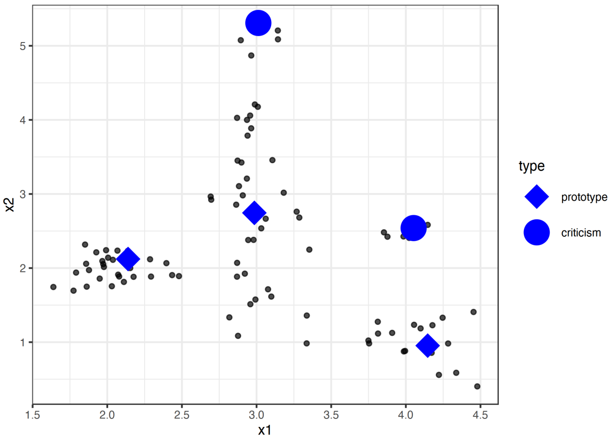


Figure 2.25: Data distribution and its prototypes and criticisms. Extracted from [14].

A method that finds prototypes and criticisms is Maximum Mean Discrepancy (MMD). By measuring the discrepancy between two distributions and the number of prototypes and criticisms chosen, this method selects the prototypes/criticisms so that their distribution is close to/distinct from the data distribution [94].

3. **Influential Instances:** Data instances whose deletion influences the model predictions are considered influential, as shown in Figure 2.26. Their identification can be performed using the deletion diagnostics method or influential functions [14].

In the first method, individual data instances are deleted, the model is retrained, and the predictions obtained are compared with the model's predictions with all instances. The second method increases the loss weight of an instance based on the gradients of the model parameters, and it is not necessary to retrain the model. This method is an excellent alternative to the previous one in models with differentiable parameters because it does not require retraining the model and, consequently, is not as computationally heavy [14, 95].

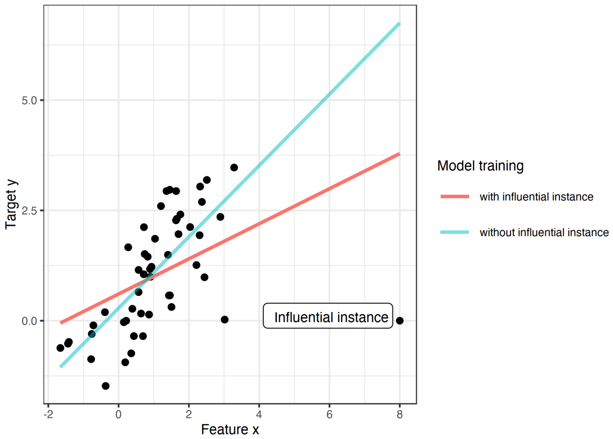


Figure 2.26: Influential instance for a linear regression model. Extracted from [14].

2.3.3.4 Neural network interpretation

The growth of the Deep Learning (DL) field has increased its application in various tasks. The architecture of NNs is very complex because of the multiple layers and parameters. Moreover, the prediction involves mathematical operations and transformations of the input in the different layers. Therefore, it becomes challenging for humans to understand the behaviour of the network from data input to prediction [14]. Although agnostic methods, such as PDPs, can be applied, it is important to consider specific explainability methods for understanding the behaviour and predictions of NNs, such as:

1. **Feature Visualisation:** Feature visualisation is a method to make learned features explicit. To do this, it finds the input that maximises the activation of a unit, and this unit can be an individual neuron, channel, or class probability neuron.

   This approach is very useful in understanding the structure and operation of NNs. However, its main problem is the illusion of interpretability, i.e., leading to the mistaken belief that the complexity of NNs is fully understood [14].

2. **Network dissection:** The network dissection method proposed by Bau et al. [96] allows the interpretability evaluation of individual units in a Convolutional Neural Network (CNN) by linking the units to human interpretable concepts.

   This interpretability evaluation has three main steps: identification of the set of visual concepts labelled by humans; measurement of the CNN channel activation for the images in the set; quantification of the alignment between the labelled concept pairs and the activations [14].

3. **Pixel Attribution:** In problems involving images, the most relevant pixels in the classification by a NN are highlighted in pixel attribution methods.

   This method is a particular case of feature assignment methods and, according to its assignment approach, can be divided into perturbation-based and Gradient-based. The first method manipulates the image to generate explanations, and the second method computes the gradient of the prediction. In both cases, to each pixel is assigned the value of its relevance in the classification [14].

2.4 Summary

MS is a chronic neurological disease that affects the central nervous system and causes the destruction of myelin, impeding adequate communication between the brain and the body. The exact cause is unknown, but it is admitted that MS may be associated with genetic, immunological, viral, bacterial, and environmental factors, among others [37]. The symptoms are heterogeneous and include motor, cognitive, and sometimes psychiatric problems [21].

During the disease course, disability can be quantified by the EDSS scale [39]. Diagnosis is challenging in the early stages of MS because symptoms can be highly variable and tend to disappear over unpredictable periods of time. Since there is no specific test for MS, clinical, imaging, and laboratory findings are combined to confirm the presence of the disease [21]. Regarding the clinical evolution, this disease can be divided into four phenotypes based on the frequency and severity of symptoms: Clinically Isolated Syndrome (CIS), Relapse Remitting (RR), Secondary Progressive (SP), and Primary Progressive (PP). RR is the most frequent course and, in most cases, evolves to SP.

Although there is no cure for MS, there are treatments available that can modify the course of the disease, reducing its activity and slowing the accumulation of disability. For a RR patient, the goal is to reduce the frequency and severity of relapses and postpone the progressive phase of the disease, while for a SP patient, the goal is to prevent progressive worsening [4, 28].

A ML model that can make an early prediction of MS progression in the first years of follow-up could significantly help physicians, leading to more appropriate treatment for each patient. In this project, the goal is to predict whether the patient will progress from the RR course to the SP form of the disease [23].

Creating a ML prediction model is a complex process that involves several steps. Initially, it is necessary to define the problem and analyse the available data. Since ML algorithms learn to map input variables to output variables, the data quality influences the model performance. This data preparation involves using or exploring the following steps: data cleaning, data transformation, and dimensionality reduction. After pre-processing the data, the next steps include the development of the classification model, exploring the use of sampling, partition, and ensemble methods. The final step is the selection of an appropriate classifier for the problem under study, using a set of different metrics to evaluate its performance [53, 67]. This imbalanced classification problem's most relevant evaluation metrics are recall, precision, and F1-score.

The question of explainability has grown with the success of more complex and opaque ML models. Explainability methods can be applied to overcome the black-box problem of ML models and increase the scientific community's acceptance and confidence in the answers. These methods can be divided into agnostic, example-based, or specific. Furthermore, it is essential to involve human experiments in the process of evaluating and validating the explainability of disease prediction models. Finally, it is important to note that much remains to be done before these methods can be reliably applied in clinical practice, but the way to do so is by being set [14].

3

State of the art

This chapter overviews the state of the art of Multiple Sclerosis (MS) progression prediction. Section 3.1 presents a review of the data type used and summarises some approaches adopted in recent years. It focuses on the study of the evolution from the Relapse Remitting (RR) course to the Secondary Progressive (SP) course. Section 3.2 is focused on the current state of the art related to the explainability of MS progression models. The chapter ends with a summary of the main ideas and future needs presented in the literature (section 3.3).

3.1 Prediction of MS progression

The treatment of diseases can be more differentiated and appropriate for each patient if the course of the disease can be predicted early. The growth of Machine Learning (ML) algorithms allows their application in the medical field, especially for classification problems. MS is the most studied autoimmune disease in the ML field [97], and the main goal of most studies is the classification, detection, and segmentation of MS lesions [98].

3.1.1 Data Used

The majority of models use clinical and Magnetic Resonance Imaging (MRI) data. Seccia et al. [18] listed some MS studies (Table 3.1) that use clinical data because this type of data has proven to be adequate in long-term prognosis.

Although Cerebrospinal fluid (CSF) represents a unique source of Central Nervous System (CNS) data and plays an important role in diagnosis, it is not routinely used in clinical practice because its collection is an invasive method [99].

MRI data is also essential in MS problems because it may clarify the pathogenic mechanisms of the disease better than purely clinical data [100, 101]. MRI is the only technique that allows non-invasive quantification and characterisation of MS lesions in space and time. Hence, there have been an increasing number of studies focusing on automated analysis of brain MRI scans, and there has been an effort to define more sophisticated MRI features that are more predictive [102]. Combining these two data types can improve discrimination between different disease courses.

In addition, the influence of other important factors such as genetics [103] and demographic characteristics [98] is also commonly studied. Another new and different approach is the collection of data using mobile devices such as cell phones, which may lead in the future to a more detailed relationship between lifestyle, disease course, and the influence of treatments [18, 104].

Table 3.1: Summary of MS prognostic studies, using ML [18].

Reference	**Problem**	**Data**	**Model**	**Most relevant features (best model)**	**Performance (best model)**
Pinto et al. 2020 [105]	RR progresses to SP in 5 years and EDSS > 3 at 6 or 10 years.	Clinical and MRI	Linear SVM, KNN, DT and LR	SP development: EDSS, CNS involvement in relapses, FS scores, age at onset Disease severity: EDSS, FS scores and CNS affected functions during relapses	RR to SP: $sensitivity = 76\%$ $specificity = 77\%$ AUC = 86% EDSS > 3 at 6y: $sensitivity = 84\%$ $specificity = 81\%$ AUC = 89% EDSS > 3 at 10y: $sensitivity = 77\%$ $specificity = 79\%$ AUC = 85%
Zhao et al. 2020 [106]	ΔEDSS ≥ 1.5 at 5 years	Clinical and MRI	Linear SVM, LR, and ensemble models	EDSS, disease course, MRI lesions, cerebellar and pyramidal function, and ambulatory index	$accuracy = 71\%$ $sensitivity = 79\%$ $specificity = 69\%$ AUC = 78%
Brichetto et al. 2020 [107]	RR progresses to SP in 4 months.	Clinical and patient reported outcomes	Linear SVM, LR KNN, and other linear classifiers	Not reported	$accuracy = 82.6\%$

Continued on next page

Table 3.1: Summary of MS prognostic studies, using ML [18].

Reference	Problem	Data	Model	Most relevant features (best model)	Performance (best model)
Seccia et al. 2020 [1]	RR progresses to SP in 0.5 to 2 years.	Clinical and MRI	Nonlinear SVM, AB, KNN, and CNN	Not studied	RR to SP in 2y (RF): $accuracy = 86.2\%$ $sensitivity = 84.1\%$ $specificity = 86.2\%$ PPV = 8.9% RR to SP in 2y (NN): $accuracy = 98\%$ $sensitivity = 67.3\%$ $specificity = 98.5\%$ PPV = 42.7%
Law et al. 2019 [108]	ΔEDSS ≥ 1 at 2 years in SP MS	Clinical and MRI	AB, RF, DT, Linear SVM, and individual and ensemble LR	EDSS, 9-Hole Peg Test, and Timed 25-Foot Walk	$sensitivity = 59\%$ $specificity = 61\%$ PPV = 32.1% NPV = 82.8%
Yoo et al. 2019 [102]	CIS progresses to MS in 2 years	Clinical and MRI	LR, RF, and CNN	Not studied	$accuracy = 75\%$ $sensitivity = 78.7\%$ $specificity = 70.4\%$ AUC = 74.6%
Zhao et al. 2017 [109]	ΔEDSS > 1.5 at 5 years	Clinical and MRI	LR and Linear SVM	Non progressive: EDSS and disease activity at 0, 6 and 12 months, brain parenchymal fraction, race, ethnicity, and family history Progressive: ΔEDSS, disease activity, active disease at baseline, T2 lesion volume, pyramidal function and its change at 1 year of follow-up	$accuracy = 67\%$ $sensitivity = 81\%$ $specificity = 59\%$
Wottschel et al. 2015 [110]	CIS progresses to MS in 1 or 3 years	Clinical and MRI	Linear SVM	CIS to MS in 1y: lesion load, type of presentation, and gender CIS to MS in 3y: age, EDSS at onset, lesion attributes	CIS to MS in 1y: $sensitivity = 77\%$ $specificity = 66\%$ CIS to MS in 3y: $sensitivity = 60\%$ $specificity = 66\%$

Continued on next page

Table 3.1: Summary of MS prognostic studies, using ML [18].

Reference	Problem	Data	Model	Most relevant features (best model)	Performance (best model)
Bejarano et al. 2011 [111]	ΔEDSS>1 + EDSS range after 2 years + relapse ocurrence	Clinical, MRI and MEP	NB, DT, LR and NN	EDSS and MEPs	ΔEDSS>1: *accuracy* = 75% *sensitivity* = 82% *specificity* = 52% AUC = 74% EDSS range: *accuracy* = 80% *sensitivity* = 92% *specificity* = 61% AUC = 76% Relapses: *accuracy* = 67% *sensitivity* = 53% *specificity* = 77% AUC = 65%

3.1.2 Clinical problems

As shown in Table 3.1, the MS study focuses on different problems using different methods. The three main classification problems are predicting conversion from Clinically Isolated Syndrome (CIS) to MS, predicting disease progression, and predicting the SP course.

3.1.2.1 Prediction of conversion from CIS to MS

Some prognostic studies using ML focus on predicting conversion from CIS to MS since an early treatment is beneficial and about 80-85% of patients progress to MS after 20 years [112, 113]. Wottschel et al. [110] combined clinical and demographic features with MRI-derived features of lesion characteristics in a Support Vector Machines (SVM) model to predict conversion of CIS to clinically-definite MS (CDMS) during one- and three-year follow-ups, obtaining the accuracy of 71.4% and 68%, respectively. More recently, Kitzler et al. [114] used advanced MRI techniques to analyse early myelin breakdown and identify an imaging biomarker associated with CIS to MS conversion through *in vivo* myelination changes.

Several studies have shown that MRI features, both the number and topography of lesions, are the main prognostic factor in early disease [102]. Both studies mentioned above confirmed that the higher the white matter lesion load in the CIS course, the higher the risk of progressing to CDMS. In these papers, the used features are associated with white matter lesions and are based on lesion masks manually created by a user, instead of using automated image analysis methods. Nowadays, theDeep Learning (DL) field has been increasingly applied to these problems. Yoo et al. [102] used Convolutional Neural Networks (CNNs) to extract latent MS lesion patterns associated with the conversion from CIS to MS disease, showing the potential advantage of this automatic extraction.

3.1.2.2 Prediction of MS progression and severity

On the other hand, some studies focus on analysing disease progression and clinical activity, such as relapses and disability. Thus, MS is classified as benign/malignant and worsening/not worsening, which are not disease phenotypes but provide an indication of the disease severity over time. This classification of disease status based on the change of the Expanded disability status scale (EDSS) value is quite common.

Law et al. [108] studied the progression of SP disability using learning models based on based on Decision trees (DTs), Logistic regression (LR), and SVMs. This was achieved by tracking patients in a clinical trial for two years and making predictions over time, based on a six-month window in advance. Thus, patients were classified as having confirmed disability progression if an increase of $\geq$1 or $\geq$0.5 was observed for EDSS $\leq$5.5 or $\geq$6, respectively. The best results were given by the DT classifier, with AUC=61.8%.

Recent studies by Zhao et al. [106, 109] have focused on worsening conditions, and patients were classified into worsening or non-worsening, according to a change greater than 1.5 in the EDSS value after five years. In the most recent study, Zhao et al. [106] used two-year clinical and longitudinal neuroimaging data to predict the patients' disability level at five years using SVM, LR, Random Forest (RF), XGBoost, Meta-L, and LightGBM. With the ensemble method LightGBM, the model achieved a sensitivity of 78%, a specificity of 68%, and accuracy of 70%.

Two of the frameworks developed by Pinto et al. [105] focus on predicting disease severity at the 6th and 10th years of follow-up, classifying a patient disease progression into benign or malignant forms. For this purpose, the patient is considered to have severe disease if the EDSS value is higher than 3. The best results were obtained for the prediction of the 6th year using clinical information from the first two years, with an Area Under the Curve (AUC) of 89%, a sensitivity of 84%, and specificity of 81%.

It is important to note that the different temporal windows of analysis limit comparison between these studies. In addition, although the EDSS scale is widely applied in these problems, it has some limitations, including the lack of consistency in the threshold value between classes [106].

3.1.2.3 Prediction of SP cases

Lastly, another approach to analyse disease progression is to predict the patients who will progress to the SP course and those who will remain in the RR course, which is the focus of this thesis.

Bergamaschi et al. [27] explored a Bayesian approach to calculate the Bayesian Risk Estimate for MS (BREMS) score in the first year of each patient's disease, which indicates the long-term risk of having the SP course. Late age at onset and polysymptomatic onset proved to be unfavourable factors, while the female gender was associated with a lower risk. This paper also suggests that, besides the number of clinical events, the characteristics of the events, such as type of onset, motor, and sphincter relapses, should also be considered.

Ion-Mărgineanu et al. [115] employed multiple binary classifiers to classify the four courses of MS defined by McDonald's criteria, combining clinical data with lesion loads and magnetic resonance metabolic features. Using the Linear Discriminant Analysis (LDA) classifier and a Non-linear SVM, it was possible to achieve a F1-score of 87% in distinguishing RR from SP, and it was concluded that the addition of metabolic features and lesion loads slightly improves the results.

Seccia et al. [1] analysed the patient's medical history to predict the RR shift for SP 180, 360, or 720 days after the last visit. This was done using the RF, SVM, k-nearest neighbours (KNN), and AdaBoost (AB) classifiers and the Long Short-Term Memory (LSTM) Neural Networks (NNs). The dataset is a time series with clinical values over several time intervals. When predicting each visit individually, the classifiers obtained recall values from 70 to 100% and precision values from 5 to 10%. When considering the data as a time series in the LSTM NN, the recall and precision values reached 67% and 42% in the 720-day prediction.

Pinto et al. [105] developed another framework to predict the SP course in RR patients. The classifier that performed best was the SVM, and the best performance for the development of SP was achieved in the 2-year model, having an AUC of 86%, a F1-score of 20%, a sensitivity of 76% and specificity of 77%. The low F1-score value commonly obtained is a limitation of the models developed since it reflects that several patients incorrectly classified as SP will be subjected to aggressive medication without actually needing it [105].

3.1.2.4 Brief description of the ML techniques commonly used

The most commonly used ML methods for disease detection and prediction are based on supervised learning. Unsupervised methods are often applied to discover and study naturally occurring patterns in data. However, supervised methods are the most widely used for detection/segmentation, therapeutic decision-making, and disease prognosis. Several classification algorithms can be used to model prediction problems [116]. These methods should be chosen and configured according to the problem under study to perform well in the classification task. Generally, a comparison is made between several classification methods using some performance metrics. The most commonly adopted classifiers in MS studies are LR, linear and non-linear SVM, DTs, RF, and NNs, especially CNN. Among the classifiers applied in the cited studies above, the performance of linear SVM, NNs, and RF stands out. The latter two can be highlighted, in comparison to SVM, due to their greater generalisation capacity that allows the detection of complex and non-linear patterns in the data [18].

The emerging role of DL

In the last decade, ML has allowed significant improvements in several areas in MS research, such as lesion detection and segmentation and prediction of disease course. The existing challenges encourage the improvement of the used methods and the exploration of new approaches. Since 2018, the DL field has grown exponentially and more advanced techniques such as CNNs and LSTMs have been applied in the study of MS [116]. DL techniques focus mainly on two areas of interest: detection and segmentation of MS lesions and prediction of disease outcome to make diagnosis more accurate and facilitate optimal clinical management of patients [117].

Therefore, the research of different ML and DL approaches is quite important for the continuous improvement of the existing models so that, in the future, the prediction of progression between RR and SP disease courses is made promptly.

3.1.3 Current methodology limitations

Data quantity

One of the main limitations of MS prognosis prediction is the quantity of available data. Data collection in the clinical environment is challenging, and without a labelled and reliable dataset, classification models are more prone to failure [118].

The building of robust models depends on adequate training with large datasets that are representative of the population. This problem is common in MS studies because there are a limited number of patients and records over the years of follow-up, the data collected are of poor quality, and there is no guarantee that a dataset is representative of the population [104]. Thus, sharing strategies can be defined, between institutes and hospitals, to obtain a greater amount of heterogeneous data while respecting ethical and data protection regulations [118].

In ML models, the implementation of Cross-Validation (CV) techniques is fundamental to minimise the risk of overfitting. Furthermore, it must be ensured that all records related to a patient are part of the same set (train, test, or validation) [18]. For the performance measure to be realistic, this test group should be independent of the others, sufficiently large and completely unknown to the model. In this case, Leave One Group Out (LOGO) CV is usually applied instead of k-fold CV, preventing the ML model from identifying specific patients rather than the data patterns [1].

In addition, the datasets used are often imbalanced since the disease has a slow course, and there are fewer records in the positive class. Some cost-sensitive strategies are adopted to minimise this problem, such as using ensemble methods, balancing the training group, and using appropriate performance metrics like F1-score and Geometric Mean [18].

Data quality

As mentioned before, ensuring the quality of the data is a big challenge, especially when it is not collected in controlled trials. There are several flaws in data recording and sometimes fields with less relevant information are ignored, resulting in datasets with misleading information and many missing values [118]. The pre-processing techniques discussed in section 2.2.1 help to work around this problem, but the quality of the datasets depends heavily on the primary data recorded. Thus, the evidence that the data provides must be reliable, and physicians should make an effort to collect complete and quality data [18]. Several measures can also be taken during the data collection process to maximise quality, such as replacing paper-based collection, which has a high error rate and involves double data processing, with a computerised collection system. These systems should be programmed with real-time automated analysis tools to identify possible failures at the time of collection [18, 113].

Generalisability and bias

ML algorithms are dependent on the available data. The training datasets should represent the population to not compromise model decisions in the real world. This dependency can affect the generalisability of the algorithm, i.e., small differences from the training conditions can result in decreased model performance [118]. An example is a study developed by Wottschel et al. [119] in which the classification accuracy of the CIS outcome at one-year follow-up was highest in datasets from each centre and reached the lowest value when all patient data from different centres were combined. This difference is because algorithms trained with MRI data collected by a specific model of an instrument sometimes cannot interpret data collected by other equipment or by the same model with a different acquisition protocol [116].

The same applies to clinical characteristics that may differ across demographics and cultures. Hence, applying a model in different regions can also lead to low performance. This inequality is also influenced by the overrepresentation of caucasian patient groups, who are socioeconomically more advantaged and have more healthcare access [18, 118]. It is important to note that the data are transformed to preserve the patients' privacy and, consequently, potentially useful data are eliminated, such as rare cases and the birthplace or ethnicity of minority groups [18].

Another challenge is the standardisation of the data to draw homogeneous conclusions between studies. In MS studies, the EDSS may behave as a noise source due to variability among neurologists in its value definition. An alternative is the automatic and objective assessment of cognitive performance [18].

Black-box models

Due to the complexity of some ML models, sometimes their logic is not easily understandable by a human and the models are considered black boxes. In these cases, clinicians do not understand the functioning and results of the models [118]. This problem has become more relevant with the increasing use of complex algorithms, such as DL, and was reinforced by the creation of the right to explanation in the General Data Protection Regulation (GDPR) 2018 [87].

Despite still being an under-explored field in MS ML models, there have been several studies that aim to create explainable models for the diagnosis and prognosis of MS and, for this, they use techniques such as Layer-wise Relevance Propagation (LRP), SHapley Additive exPlanations (SHAP), and Local Interpretable Model-Agnostic Explanations (LIME). This black-box problem influences the safety and use of the models in clinical practice due to the multiple issues in their regulatory approval [18, 118].

3.2 Explainability and MS

Artificial Intelligence (AI) techniques, along with the increasing availability of medical data, have allowed the creation of promising applications in several medical problems, such as MS prediction. However, their application in clinical practice is minimal due to the black-box problem mentioned in the previous section, which results in concerns in understanding the behaviour of the models [120]. Thus, explainability is crucial to maximise the error recognition of systems and the understanding of solutions in real-life scenarios [121]. In recent years, there have been several ML studies in different medical fields, such as Cardiology and Neurology, that include the analysis of the models developed through explainability techniques.

Eitel et al. [122] developed an explainable framework for the MS diagnosis. This framework is based on 3D CNNs and LRP models with Fluid-attenuated inversion recovery (FLAIR) imaging sequences. The CNN model obtained a balanced accuracy of 87.04% and an AUC of 96.08%. In addition, the LRP method produced heatmaps of the input images indicating the most relevant voxels for the final CNN classification result. Thus, from the heatmaps obtained, it was observed that the CNN model focused on the well-established MRI markers in MS. It was concluded that the results are consistent with clinical knowledge and that the LRP method leads to clear and intuitive explanations of the results of this model.

Lopatina et al. [123] developed a similar approach for the identification of MS patients. In this study, a CNN neural network was trained with 2D Susceptibility-weighted images (SWI) images. The attribution methods LRP and DeepLIFT were tested to visually analyse the contribution of each voxel to the classification task. The heatmaps analysis revealed specific relevant brain areas common to most patients in a class and showed that the most pertinent voxels are located in and around veins. These observations reinforce the assumed relationship between changes in the vascular system and the development of MS.

Reinhold et al. [124] developed a structural causal model that creates counterfactual MRI images for MS patients based on demographic information, MRI images, and disease covariates. The counterfactual images obtained allow modelling of disease progression in MS cases by showing how the MRI image of the brain would look if the information were changed; for example, if the lesion volume were 0 mL or if the EDSS were 5. However, this model has some limitations, such as the poor quality of the counterfactual images in the test set and the fact that the explanations obtained are not validated in a meaningful way since they are hypothetical results.

More recently, Sousa et al. [125] implemented several explainability methods, such as Permutation Feature Importance (PFI), LIME, and SHAP, in the work developed by Pinto et al. [105]. The ML model selected uses clinical information from the first two years after MS diagnosis to predict disease severity (benign vs malignant) at the sixth year of follow-up. It was concluded that the Functional System (FS)-related features are relevant in most explanations, but the EDSS feature had the highest prominence. Data scientists evaluated the explanations to determine what improvements were needed. This evaluation concluded that explanations of predictive models need to be simple and straightforward for clinicians. However, to describe the logic of the predictions from an algorithmic perspective, the data scientists suggested that the explanations should be more technical and detailed.

By extending the analysis of explainability methods to problems of prognosis of neurodegenerative diseases, such as Alzheimer's and Parkinson's, it is found that the most used method is SHAP, followed by LIME [125]. The rising popularity of SHAP can be explained by the fact that it links LIME and Shapley values and provides a global and local understanding of model predictions and the impact of features by plotting summary charts, whereas LIME only gives local explanations. A key point is its fast implementation for tree-based models [14]. Furthermore, there is growing popularity in adopting more complex classifiers such as NN and Extreme Gradient Boosting (XGBoost) due to their ability to achieve better performance [125].

In the MS domain, explainability studies focus mainly on diagnostic models using image information. Since this field of research is recent, these presented studies are only the beginning of many approaches that will be developed in the future. There is still a long way to go before clinicians are confident that a ML model can be trusted, both in MS and in healthcare in general, but that path is being paved.

3.3 Summary

In conclusion, the development of ML models for MS data follows a difficult path with multiple obstacles. Over the past few years, there has been an exponential increase in the number of ML studies focused on MS and, more recently, in the field of DL, where there have been improvements in early disease detection, prognosis, and lesion detection, and segmentation [116].

The application of complex NN shows to be capable of identifying minor differences between disease courses and is a promising approach in its prediction. There is still a need to promote the models' explainability and investigate in detail their safety and risks in clinical decision-making. In the future, explainability models will play a crucial role in both increasing clinicians' confidence in ML models and expanding the knowledge of MS disease.

Therefore, it is essential to continue progressively improving the work done by comparing the performance of various methods on different datasets and analysing inconsistencies between distinct studies. Regarding data, there is a need for techniques to build quality datasets with many representative samples. Additionally, it is crucial to explore the addition of new data sources [112, 117]. These approaches will facilitate the early accurate diagnosis, analysis of the disease progression, and treatment choice done by neurologists, consequently improving the life quality of MS patients.

4

Methodology

This chapter describes the steps used in the development of Multiple Sclerosis (MS) prediction algorithms and explainability methods. First, the dataset is presented in section 4.1. Then, the two classification scenarios (Visited-Oriented (VO) and History-Oriented (HO)) developed are described in sections 4.2 and 4.3. Finally, the explainability methods applied are detailed in section 4.4.

The general framework adopted is summarised in Figure 4.1. To predict the MS disease course, a VO approach and a HO approach were used in a Machine Learning (ML) binary classification task. These methodologies were used to predict whether a patient will pass from the Relapse Remitting (RR) phase to the Secondary Progressive (SP) phase within a given time window (180, 360, or 720 days).

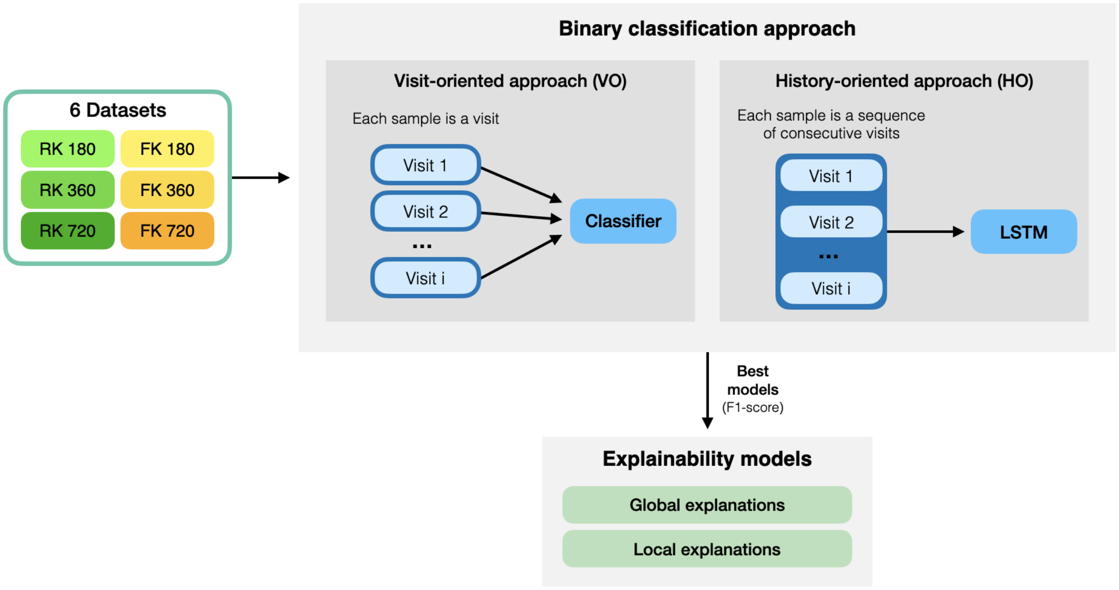


Figure 4.1: Methodology used with the datasets obtained by the Record-keeping (RK) and Feature-keeping (FK) pre-processing strategies.

In the VO approach, different classifiers were applied, and each visit was considered an isolated sample. On the other hand, in the HO approach, a Long Short-Term

Memory (LSTM) Neural Networks (NNs) were used, and each sample is the entire clinical history of a patient, i.e., it is a clinical time series. These models were built using the open-source packages scikit-learn and Keras on Python 3.6.9.

Afterwards, for each of the approaches and time windows, the model with the highest prediction performance was selected, and different explainability methods were implemented. The explanations obtained were then compared and analysed.

4.1 Dataset

The dataset used in the problem is from the MS service of the Sant'Andrea Hospital in Rome and was processed by Seccia et al. [1]. For each patient, one sample corresponds to one visit. The data include clinical status and laboratory and imaging data from the neurological examination. The features present in the dataset are listed in Table 4.1. The Magnetic Resonance Imaging (MRI) and liquor analysis features are boolean (yes/no) and have the following characteristics [1]:

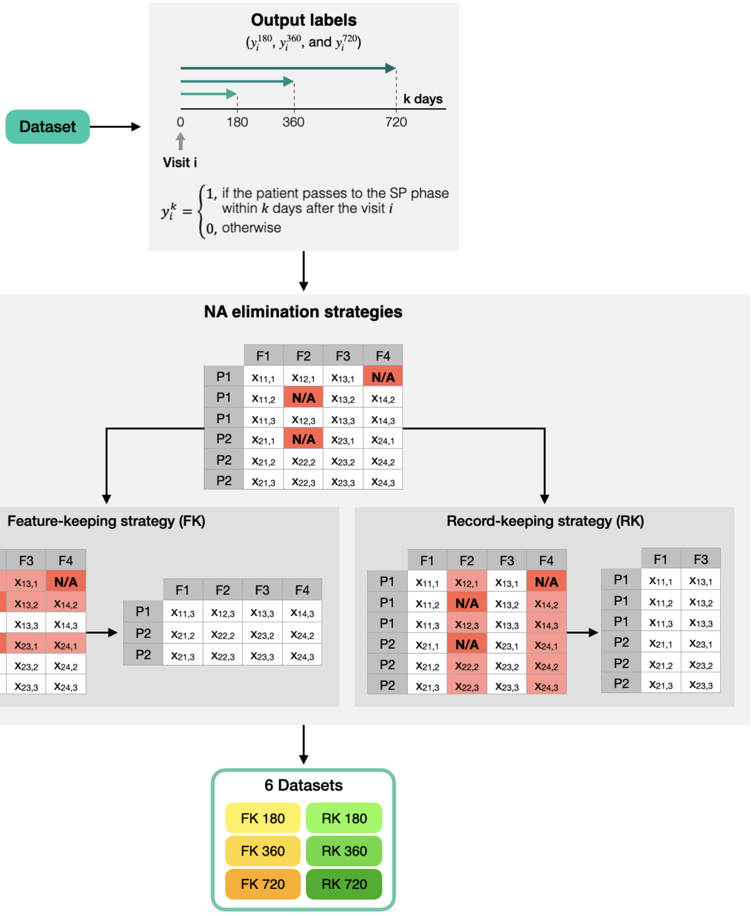
- **Status T1/T2:** Presence of gadolinium-enhancing T1/T2 lesions.
- **Oligoclonal banding:** Presence of oligoclonal bands in liquor.
- **Others:** Presence of lesions in the respective regions.

Table 4.1: Types of features present in the datasets [1].

Type	Feature
Demographic	Age at onset Gender Age at Visit
Clinical Features	EDSS Relapses from last visit Pregnancy Relapses frequency Time from last relapse
MRI and liquor	Status T1 Status T2 Oligoclonal Banding Spinal Cord Supratentorial Optic Pathway Brainstem-Cerebellum
Therapeutic treatments (drugs)	Relapse treatment drugs First line DMT Immunosuppressant MS symptomatic treatment drugs Second line DMT Other drugs

The datasets were designed to predict a patient's transition from phase RR (0) to phase SP (1) after 180, 360, and 720 days. Thus, Primary Progressive (PP) patient cases and data from visits after the transition to SP were excluded. The class label is 1 for a given sample if the patient progresses to the SP phase within 180, 360, or 720 days after that visit.

Furthermore, in the data pre-processing (see Figure 4.2), missing values were removed using two strategies: FK and RK. In the FK strategy, the dataset authors removed all samples with at least one missing value, while in the RK strategy, they eliminated the features with missing values. Thus, two different datasets were generated from the 180-, 360-, and 720-day prediction datasets, totalling the six shown in Table 4.2.


Dataset
Output labels
(y_i^{180}, y_i^{360}, and y_i^{720})
k days
0
180
360
720
Visit i
NA elimination strategies
Feature-keeping strategy (FK)
Record-keeping strategy (RK)
6 Datasets
FK 180
RK 180
FK 360
RK 360
FK 720
RK 720

$$y_i^k = \begin{cases} 1, & \text{if the patient passes to the SP phase within } k \text{ days after the visit } i \\ 0, & \text{otherwise} \end{cases}$$

NA elimination strategies

	F1	F2	F3	F4
P1	$x_{11,1}$	$x_{12,1}$	$x_{13,1}$	N/A
P1	$x_{11,2}$	N/A	$x_{13,2}$	$x_{14,2}$
P1	$x_{11,3}$	$x_{12,3}$	$x_{13,3}$	$x_{14,3}$
P2	$x_{21,1}$	N/A	$x_{23,1}$	$x_{24,1}$
P2	$x_{21,2}$	$x_{22,2}$	$x_{23,2}$	$x_{24,2}$
P2	$x_{21,3}$	$x_{22,3}$	$x_{23,3}$	$x_{24,3}$

Feature-keeping strategy (FK)

	F1	F2	F3	F4
P1	$x_{11,1}$	$x_{12,1}$	$x_{13,1}$	N/A
P1	$x_{11,2}$	N/A	$x_{13,2}$	$x_{14,2}$
P1	$x_{11,3}$	$x_{12,3}$	$x_{13,3}$	$x_{14,3}$
P2	$x_{21,1}$	N/A	$x_{23,1}$	$x_{24,1}$
P2	$x_{21,2}$	$x_{22,2}$	$x_{23,2}$	$x_{24,2}$
P2	$x_{21,3}$	$x_{22,3}$	$x_{23,3}$	$x_{24,3}$

	F1	F2	F3	F4
P1	$x_{11,3}$	$x_{12,3}$	$x_{13,3}$	$x_{14,3}$
P2	$x_{21,2}$	$x_{22,2}$	$x_{23,2}$	$x_{24,2}$
P2	$x_{21,3}$	$x_{22,3}$	$x_{23,3}$	$x_{24,3}$

Record-keeping strategy (RK)

	F1	F2	F3	F4
P1	$x_{11,1}$	$x_{12,1}$	$x_{13,1}$	N/A
P1	$x_{11,2}$	N/A	$x_{13,2}$	$x_{14,2}$
P1	$x_{11,3}$	$x_{12,3}$	$x_{13,3}$	$x_{14,3}$
P2	$x_{21,1}$	N/A	$x_{23,1}$	$x_{24,1}$
P2	$x_{21,2}$	$x_{22,2}$	$x_{23,2}$	$x_{24,2}$
P2	$x_{21,3}$	$x_{22,3}$	$x_{23,3}$	$x_{24,3}$

	F1	F3
P1	$x_{11,1}$	$x_{13,1}$
P1	$x_{11,2}$	$x_{13,2}$
P1	$x_{11,3}$	$x_{13,3}$
P2	$x_{21,1}$	$x_{23,1}$
P2	$x_{21,2}$	$x_{23,2}$
P2	$x_{21,3}$	$x_{23,3}$

Figure 4.2: Pre-processing steps of the datasets worked on by Seccia et al. [1]

Table 4.2: Characteristics of the different datasets used [1].

	Strategy	Features	Records	Patients	SP Patients	% SP Records
180 days	FK	21	4330	506	36	0.8
	RK	18	14923	1515	207	1.3
360 days	FK	21	4202	495	37	0.8
	RK	18	14238	1449	207	1.4
720 days	FK	21	3928	468	37	0.9
	RK	18	13178	1375	207	1.5

It is important to note that the authors hid the identification of most of the features in the dataset to protect the identity of the patients, and this information was not provided. This represents a significant limitation to the work of this thesis since one of the goals is to explore the explainability of the ML models developed for MS progression. It is fundamental to know the meaning of each hidden feature to evaluate which ones have the most decisive influence on the results, the degree of interaction between them, and how these interactions work. Thus, to proceed with the planned work, different strategies were explored to identify some of the dataset features.

From the number of men and women present in the datasets, it was concluded that feature F1 corresponds to gender. Additionally, by analysing the correlation matrix illustrated in the study of Seccia et al. [1] (see Figure 4.3) it was concluded that F10 and F13 correspond to the feature “Pregnancy” in the RK and FK datasets, respectively. It was also found that features F2, F3, F4, and F5 are from the MRI and liquor group and correspond to the features Spinal Cord, Supratentorial, Optic Pathway and Brainstem-Cerebellum. Still, it was not possible to identify which was which.

As mentioned before, the MRI and liquor features are boolean, so they always assume the values 0 or 1. Thus, excluding the previously identified features, it was concluded that features F7, F8 and F9 in the FK dataset correspond to Status T1, Status T2 and Oligoclonal Banding. Still, it was not possible to identify which was which. In the case of the datasets obtained by the RK approach, no more binary features are present, so it can be concluded that these were eliminated during the processing illustrated in Figure 4.2.

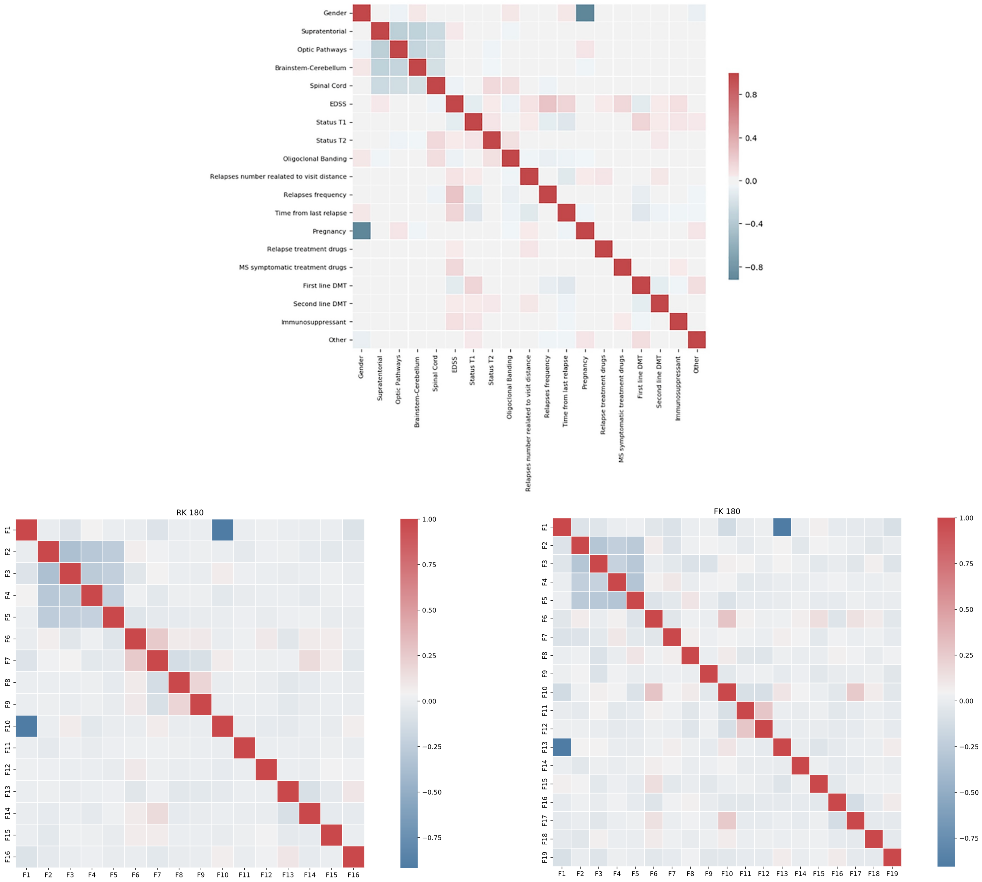


Figure 4.3: Pearson correlation matrix between the dataset features for the prediction within 180 days. At the top is the matrix presented by Seccia et al. [1] with the identification of the features, on the left is the matrix obtained by the RK dataset, and on the right is the Pearson matrix obtained by the FK dataset.

As mentioned earlier and observed in Figure 4.3, the number assigned to a given feature is not the same in both dataset types; for example, the feature Pregnancy corresponds to feature F10 in the RK datasets but corresponds to feature F13 in the FK datasets. These differences make it very difficult to compare the explanations obtained in the two problems. To work around this problem, an adjustment of the assigned names was performed.

Table 4.3 shows the identified features and the ones that could not be found (question mark). As can be seen, it was impossible to identify most of the features in the datasets. This is a huge hindrance to the analysis of the results obtained in this work since it is impossible to analyse in detail the explanations obtained and the influence of all features on the prediction.

Table 4.3: Identified features in the RK and FK datasets.

Feature	RK Datasets	FK Datasets
F1	Gender	Gender
F2	MRI and liquor*	MRI and liquor*
F3	MRI and liquor*	MRI and liquor*
F4	MRI and liquor*	MRI and liquor*
F5	MRI and liquor*	MRI and liquor*
F6	EDSS	EDSS
F7	—	MRI and liquor**
F8	—	MRI and liquor**
F9	—	MRI and liquor**
F10	?	?
F11	?	?
F12	?	?
F13	Pregnancy	Pregnancy
F14	?	?
F15	?	?
F16	?	?
F17	?	?
F18	?	?
F19	?	?

* Spinal Cord, Supratentorial, Optic Pathway or Brainstem-Cerebellum
** Status T1, Status T2 or Oligoclonal Banding

4.2 Visited-Oriented approach

In the VO approach, the predictions are made considering a single visit: a sample with information from an individual clinical record. Several ML models were trained by varying the classifiers and their parameters. Five supervised classifiers have been used: k-nearest neighbours (KNN), Random Forest (RF), AdaBoost (AB), and linear and non-linear Support Vector Machines (SVM). In short, in each iteration, one patient is selected for the testing and evaluation of classifier performance. The remaining patients represent the classifier training group. This process, represented in Figure 4.4, is repeated ten times for all patients, and the final performance is given by the average of the results of all runs. This pipeline was repeated for the best-performing model, but in this case, a step with feature selection methods was included (dashed block).

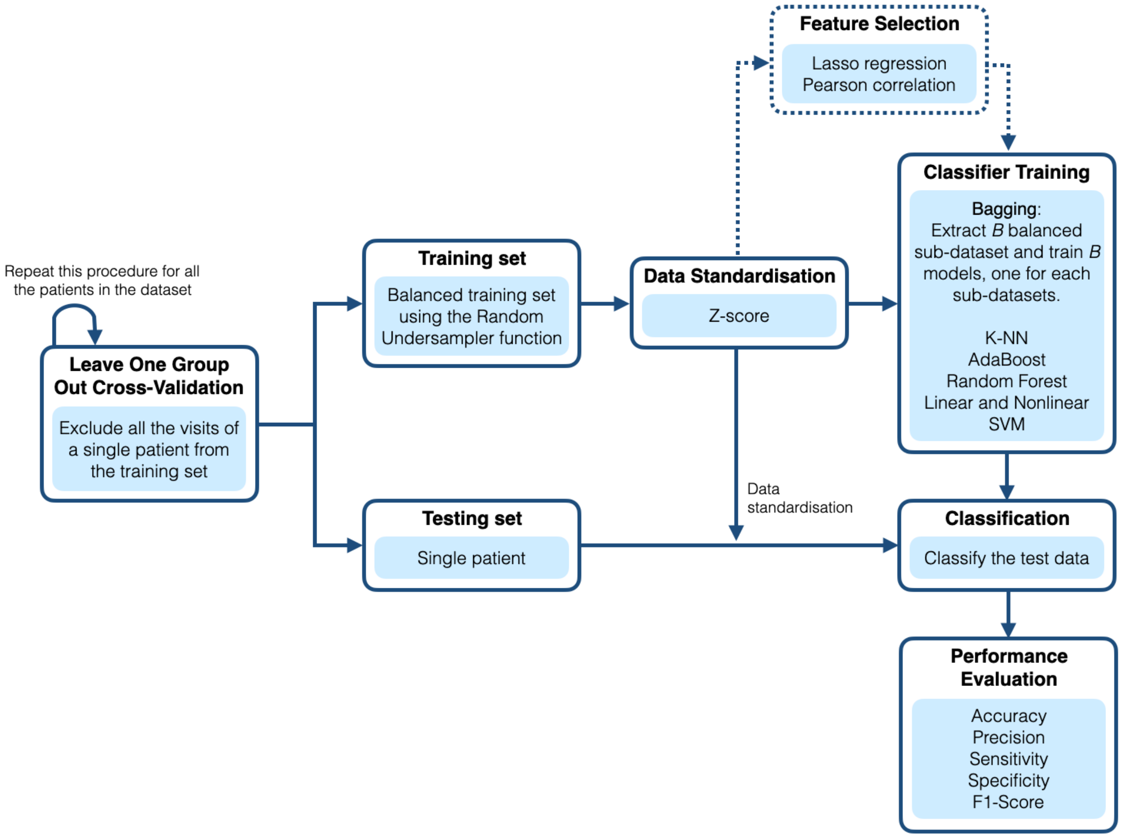


Figure 4.4: ML pipeline used for the VO approach.

Since each patient visit is considered an isolated sample, there are approximately 4000 samples in the FK datasets, of which 0.8-0.9% are SP samples, and about 14000 samples in the RK datasets with 1.3-1.5% SP samples. This big difference in the number of samples for each class reflects the imbalance of the dataset.

4.2.1 Partition and balancing methods

Due to the fact that the problem in the study is extremely imbalanced and, consequently, so is the dataset, the standard train-validation-test split strategy was not used. Instead, a Cross-Validation (CV) procedure was implemented. In the traditional CV approach, the entire dataset is partitioned in k folds, and k-1 folds are used to train, and the left one is used for the test. However, this methodology was not the most reliable in the problem in the study since the model could recognize the patient from some specific features. Thus, a Leave One Group Out (LOGO) CV procedure has been implemented. In this case, k is the number of patients, and in each iteration, all visit records of a single patient compose the test group, and the remaining patients constitute the training group. As a result, in each iteration, the model is tested on a patient who is completely new to the model. After splitting the data into train and test sets, the train set was standardised using a z-score to convert the data into a common range. The mean and standard deviation obtained from the train set were used to transform the test data.

The skew in the class distribution can influence the ML algorithms and make them ignore the minority class. One approach to address the class imbalance problem is to randomly resample the training dataset. The used technique was Random Undersampling which involves randomly selecting examples from the majority class and deleting them from the training dataset until both classes have the same number of samples.

To reduce the variance of the algorithms, the ensemble Bootstrap aggregation (Bagging) procedure was also implemented. This technique makes predictions more accurate by combining the predictions from B ML algorithms. The hyperparameters used were the default ones, which means that the number of base estimators was B=10. As shown in Figure 4.5, the B classifiers were trained on different partitions of the training data, and the combination of all the predictions defines the final prediction for the input vector.

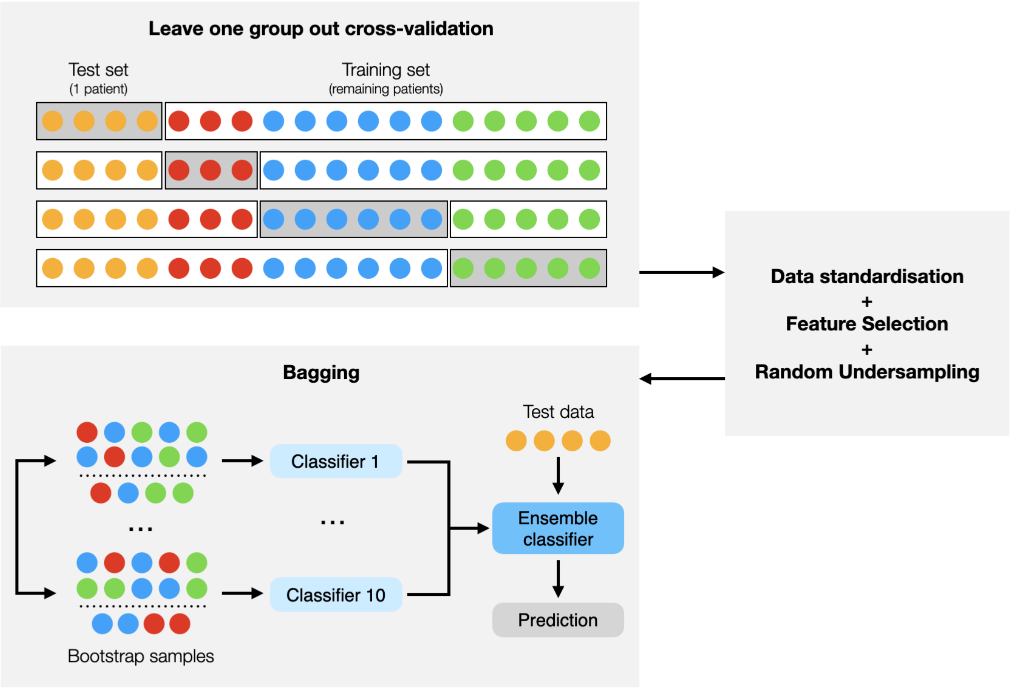


Figure 4.5: Schematic of the LOGO resampling method followed by the Bagging method.

4.2.2 Feature Selection

Although Seccia et al. [1] did not report the application of feature selection methods, after analysing the results obtained through a methodology similar to his, it was decided to test these methods and analyse their impact on the results. Therefore, feature selection methods were applied only to the best-performing model of each dataset. This choice resulted from the high computational and temporal cost associated with their application to a large number of approaches and classifiers developed so far.

This step allowed the selection of a subset that increases the generalisation ability and maximises the model performance. Moreover, the explainability of the developed models is increased by only considering the most relevant features for the prediction problem [126]. The lower the number of features, the easier it is to interpret the model [14]. The two feature selection methods applied were Pearson correlation and Least Absolute Shrinkage and Selection Operator (LASSO) regression.

Pearson's correlation method is a simple and fast filter method that calculates the correlation between each feature and the output [64]. Thus, the 3, 5 and 10 features with the highest correlation values were selected. On the other hand, the LASSO regression method is computationally more expensive. It has a regularisation process that penalises the weight of various features, reducing it to zero in some cases. Features whose weight differs from zero are selected, while the remaining ones are eliminated. In this process, the tuning parameter λ controls the influence of the penalty, and the higher its value, the greater the regularisation action, i.e., the more features are eliminated [126]. The value of the tuning parameter λ was set to 0.001, 0.003 and 0.005.

4.2.3 Classifiers and hyperparameter optimisation

Several classifiers were tested to evaluate the results using different approaches and the model performance for different approaches. Therefore, it is possible to analyse whether the model is good regardless of the classifier. The ML models considered were: KNN, RF, AB, and linear and non-linear SVM.

The grid-search technique was used to identify the optimal hyperparameters for each model from a range of values. The performance metrics of the model were calculated, and the main performance indicator for choosing the best set of hyperparameters was the F1-score. This choice is motivated by the fact that this metric provides important information about the model, giving the balance between precision and recall, as explained in detail previously in Section 2.2.3.

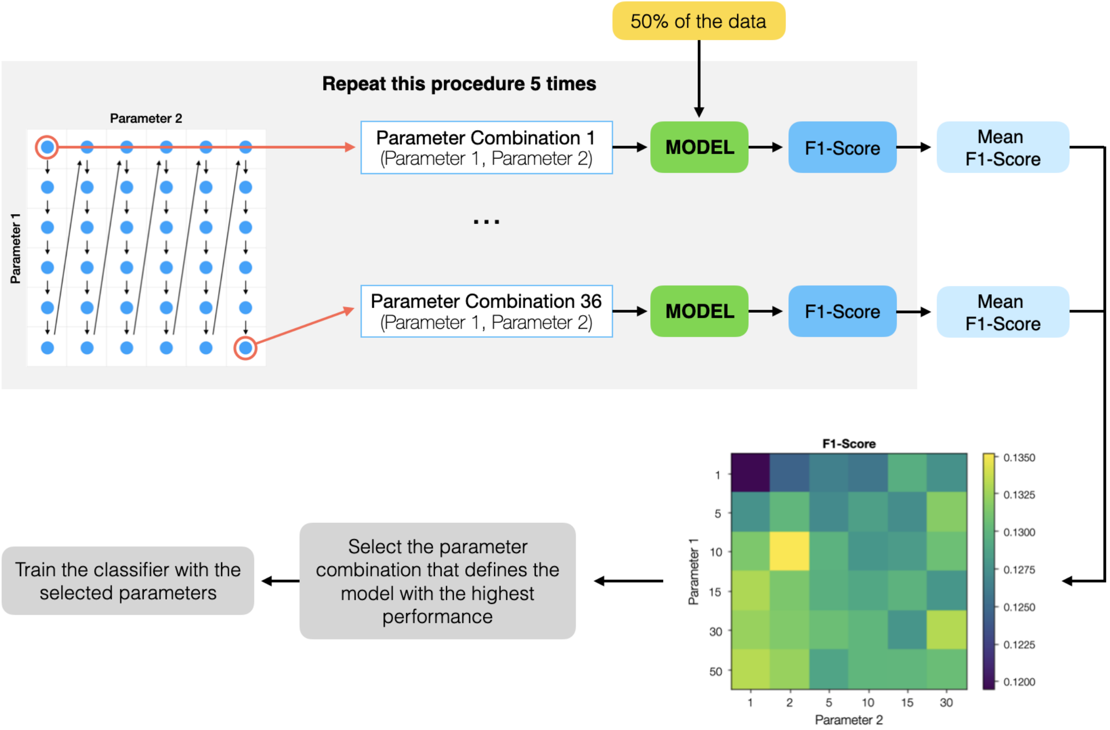


Figure 4.6: Selection process of the best training parameters.

The first classifier used was KNN, where a range of 1:2:20 values was tested for k. In the case of a RF classifier, hyperparameters include the number of decision trees and the maximum depth of each tree. It used the following number of trees and the maximum depth: {1, 2, 5, 10, 15} and {1, 2, 5, 10, 15, 30}, respectively. The AB method builds a robust classifier by combining multiple poorly performing classifiers, and, to determine the number of estimators, the model was trained with the following values: {1, 3, 5, 10, 15, 30, 50, 100}. Finally, we tested the same strategy with linear and non-linear SVMs. The first one tested the regularisation parameter C with different orders of magnitude from 10^{-5} to 10^{4}. The maximum number of iterations within the solver was also changed to 500000. The default is -1, which means there is no limit, and, in this problem, it increases the runtime significantly. In the case of the non-linear SVM, the parameters C and Gamma influence the decision boundary, and both were changed from 10^{-5} to 10^{4}.

All of these methodologies were repeated ten times, and, to speed-up the grid-search, only half of the experimental data was considered. That is, at each iteration, a subset with 50% of the data was randomly selected, keeping the class ratio of the original dataset. The mean of the performance metrics was calculated, and the chosen parameters were the ones that led to the highest value of the F1-score.

4.3 History-Oriented approach

In the HO approach, each sample of the training dataset is a sequence of consecutive visits of one patient, where each patient is considered a time series. Thus, the number of samples is the number of patients and not the total number of visits present in the dataset, i.e., it is about 500 patients in the FK datasets and 1500 patients in the RK datasets. The percentage of SP patients is the same as the previous approach, i.e., 0.8-0.9% and 1.3-1.5% for the datasets obtained by the FK and RK approaches, respectively.

Classifiers were trained to predict if patients will shift from the initial RR to the SP form, namely a NN that combined LSTM and Feed-Forward Neural Network (FFNN) layers. Figure 4.7 illustrates the ML pipeline used for the HO approach. In short, the data was divided into a test group (30%) and a training group (70%), keeping in both groups the proportions of patients evolving to SP. The Bagging algorithm was implemented with the NN, and the final performance was given by the average of the values obtained in the ten iterations performed.

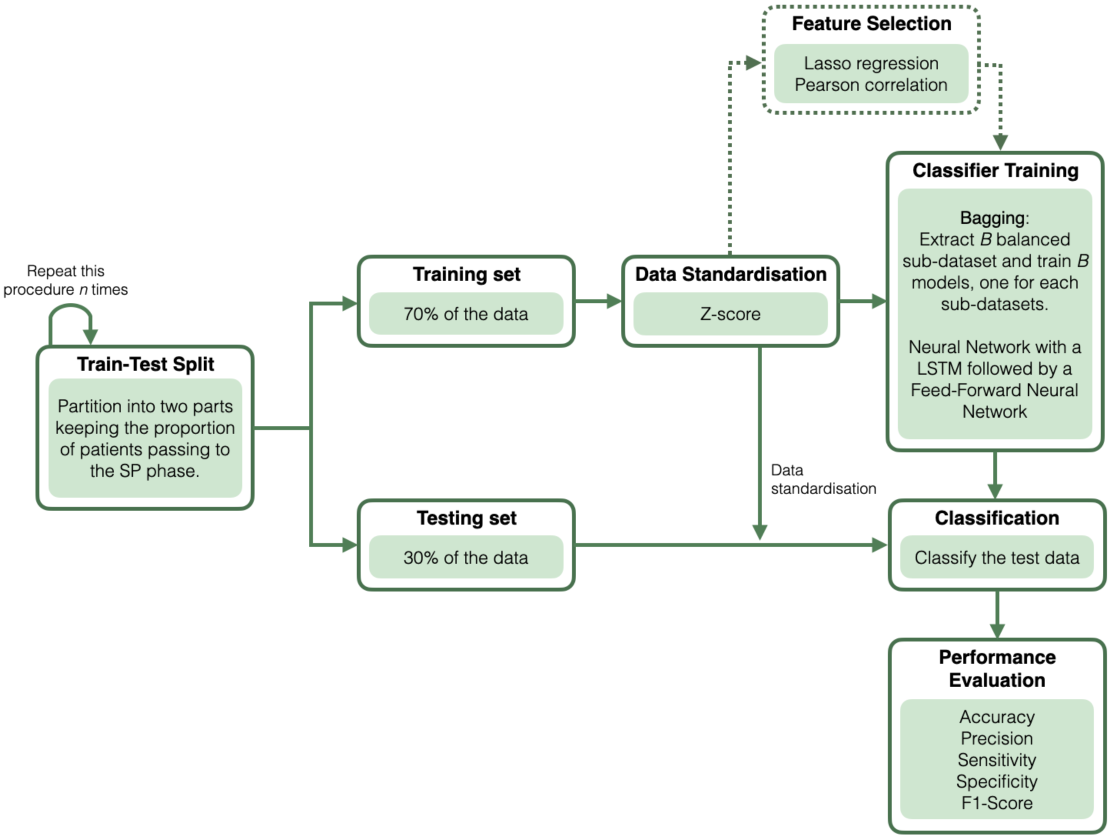


Figure 4.7: ML pipeline used for the HO approach.

4.3.1 Partition and balancing methods

Due to the computing cost and time, the LOGO procedure was not applied. Instead, the data were randomly split into training (70%) and testing (30%) sets over ten iterations ($n = 10$), with the class ratios preserved. Like in the previous approach, the data was standardised with the z-score metric, following the same methodology.

In this approach, the Bagging algorithm was also implemented, with ten bootstrap samples (default, B=10), to increase the stability of the model. The application of this algorithm in models with NN reduces forecast errors and their variation and improves short-term prediction [127].

4.3.2 Feature Selection

The feature selection process applied was the same as in the previous approach (Section 4.2.2), i.e., the Pearson correlation and the LASSO regression methods were used in the best model obtained for each problem. The same values were also chosen for the number of features selected (3, 5 and 10) and for the tuning parameter λ (0.001, 0.003 and 0.005), respectively.

4.3.3 Neural Network

The proposed NN architecture is a combination of LSTM and FFNN layers. LSTM is a form of Recurrent Neural Network (RNN) characterised by the ability to learn long-term dependencies. In addition, this network has complex layers that include feedback connections that regulate the flow of information, making these networks a proper model for time series prediction [128]. FFNN, on the other hand, has no feedback from the neurons' outputs to the inputs; that is, it has a unidirectional movement of information [129].

The proposed network can learn from the data of a time series with information collected in routine visits of MS patients. This patient's medical history constitutes an abundant source of knowledge with a lot of potential in classifying the evolution of MS [1].

Table 4.4 presents the different architectures and hyperparameters explored. Due to the computational cost associated with grid-search, the hyperparameters optimisation was done through the manual search approach, i.e., different combinations of hyperparameters were tested, as well as single-layer and stacked LSTM models. For each prediction problem, i.e., for each dataset used, the model leading to the best performance was selected. As with the previous approach, this selection was based on the value obtained for the F1-score metric.

Table 4.4: The tested NN architectures.

Model	NN Layers	No. of cells	Dropout	Function
Model 1	Masking	1	—	—
	LSTM	3	0.2	—
	Dense	1	—	Sigmoid
Model 2	Masking	1	—	—
	LSTM	6	0.2	—
	Dense	1	—	Sigmoid
Model 3	Masking	1	—	—
	LSTM	8	0.2	—
	Dense	1	—	Sigmoid
Model 4	Masking	1	—	—
	LSTM	10	0.2	—
	Dense	1	—	Sigmoid
Model 5	Masking	1	—	—
	LSTM	10	0.2	—
	LSTM	5	0.2	—
	Dense	1	—	Sigmoid
Model 6	Masking	1	—	—
	LSTM	20	0.2	—
	LSTM	10	0.2	—
	Dense	1	—	Sigmoid

To work with a LSTM network, the input sequences must have a fixed length. In this case, the length of the sequences is variable since there are patients with different numbers of visits. All samples were padded to the same length, which was defined as the maximum number of visits among all patients multiplied by the number of features. Then, a Masking layer was applied before the first LSTM layer to ignore these padded elements, as shown in Figure 4.8.

The dropout regularisation was applied to the LSTM layers to avoid the risk of overfitting and to improve model performance [130]. This method temporarily and randomly eliminates certain neurons during training to ensure they do not influence forward propagation. The proportion of neurons to be dropped out was defined as $p = 0.2$.

The network ends with a FFNN to make predictions. It uses the sigmoid activation function that produces a probability output from 0 to 1 that can be converted to class values using a threshold of 0.5. If the probability is lower than

0.5, then the output is 0; if the probability is equal to or higher than 0.5, then the output is 1.

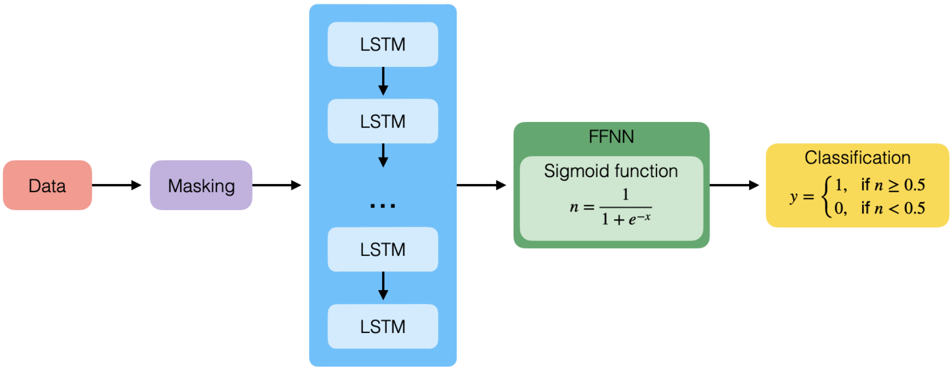


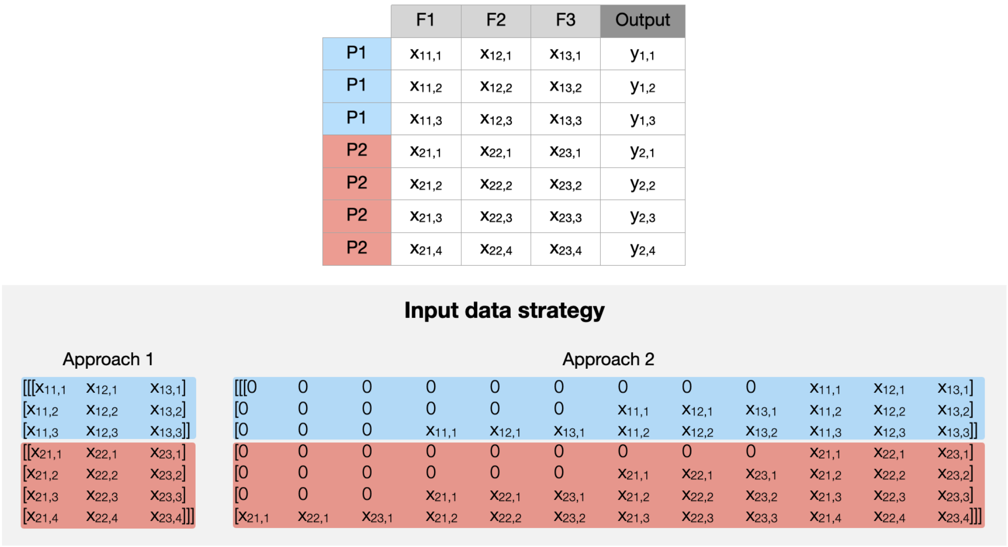
Figure 4.8: Architecture of the LSTM model.

The optimisation algorithm and the loss function significantly impact the production of optimal and fast results. The optimiser chosen was Adaptive Moment Estimation (ADAM) since it achieves good results fast, and it is considered the optimiser that performs the best on average [131]. The loss function applied was Binary Cross-entropy because it is the default loss function to use in binary classification tasks [132].

4.3.4 Input shape

Two input types were used, whose transformation is illustrated in Figure 4.9. In the first approach, the network is trained with the time series of each patient, and each record in the time series has information only from one visit. On the other hand, in the second approach, the data were transformed so that each record is considered together with all previous records. The pre-padding method was used to have all the samples of the same size since it is more efficient than post-padding in the case of LSTM [133].

These two approaches allowed exploring different ways of training LSTM networks. As mentioned before, LSTMs can store information and learn the long-term dependencies between the different time steps of the sequence data. This network processes the data from one sequence at a time and updates the state of the network so that it contains the information from all previous time steps [134]. Thus, for input 1, the information from each time step of the time sequence is presented separately. In input 2, in addition to the new information, the information of previous time steps already seen by the network is presented again.



	F1	F2	F3	Output
P1	$x_{11,1}$	$x_{12,1}$	$x_{13,1}$	$y_{1,1}$
P1	$x_{11,2}$	$x_{12,2}$	$x_{13,2}$	$y_{1,2}$
P1	$x_{11,3}$	$x_{12,3}$	$x_{13,3}$	$y_{1,3}$
P2	$x_{21,1}$	$x_{22,1}$	$x_{23,1}$	$y_{2,1}$
P2	$x_{21,2}$	$x_{22,2}$	$x_{23,2}$	$y_{2,2}$
P2	$x_{21,3}$	$x_{22,3}$	$x_{23,3}$	$y_{2,3}$
P2	$x_{21,4}$	$x_{22,4}$	$x_{23,4}$	$y_{2,4}$

Input data strategy

Approach 1

[[[$x_{11,1}$	$x_{12,1}$	$x_{13,1}$]
[$x_{11,2}$	$x_{12,2}$	$x_{13,2}$]
[$x_{11,3}$	$x_{12,3}$	$x_{13,3}$]]
[[$x_{21,1}$	$x_{22,1}$	$x_{23,1}$]
[$x_{21,2}$	$x_{22,2}$	$x_{23,2}$]
[$x_{21,3}$	$x_{22,3}$	$x_{23,3}$]
[$x_{21,4}$	$x_{22,4}$	$x_{23,4}$]]]

Approach 2

[[[0	0	0	0	0	0	0	0	0	$x_{11,1}$	$x_{12,1}$	$x_{13,1}$]
[0	0	0	0	0	0	$x_{11,1}$	$x_{12,1}$	$x_{13,1}$	$x_{11,2}$	$x_{12,2}$	$x_{13,2}$]
[0	0	0	$x_{11,1}$	$x_{12,1}$	$x_{13,1}$	$x_{11,2}$	$x_{12,2}$	$x_{13,2}$	$x_{11,3}$	$x_{12,3}$	$x_{13,3}$]]
[0	0	0	0	0	0	0	0	0	$x_{21,1}$	$x_{22,1}$	$x_{23,1}$]
[0	0	0	0	0	0	$x_{21,1}$	$x_{22,1}$	$x_{23,1}$	$x_{21,2}$	$x_{22,2}$	$x_{23,2}$]
[0	0	0	$x_{21,1}$	$x_{22,1}$	$x_{23,1}$	$x_{21,2}$	$x_{22,2}$	$x_{23,2}$	$x_{21,3}$	$x_{22,3}$	$x_{23,3}$]
[$x_{21,1}$	$x_{22,1}$	$x_{23,1}$	$x_{21,2}$	$x_{22,2}$	$x_{23,2}$	$x_{21,3}$	$x_{22,3}$	$x_{23,3}$	$x_{21,4}$	$x_{22,4}$	$x_{23,4}$]]]

Figure 4.9: Schematic of the input transformation for two patients (P1 and P2).

Furthermore, it is important to note that it was necessary to prepare the data for the LSTM network since the input to the LSTM layer must have three dimensions: samples, time steps, and features.

4.4 Explainability methods

The methods used were global and local model-agnostic interpretation methods, which stand out for their flexibility, as shown in Figure 4.10. Global methods aim to explain how the model makes predictions as a whole, while local methods aim to clarify how individual predictions are made [14].

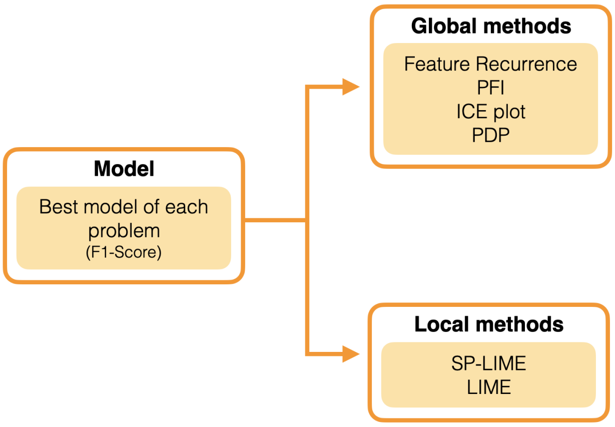


Figure 4.10: Explainability methods applied.

Thus, several explainability methods were implemented in the best-performing models created through the methodology described in the previous section that include the feature selection step. To implement these methods, the Python libraries Scikit-learn and Lime were used.

4.4.1 Global explanations

As mentioned earlier, global methods give general explanations of the model's behaviour and help to understand the logic of its results relying on features. This logic is closely related to the input data. Therefore, it is important to understand how the features contribute to the way the model makes its predictions and whether there is consistency in this learning process across different datasets [135].

To achieve this goal, four methods were applied: the determination of feature recurrence, the calculation of Permutation Feature Importance (PFI), and the visualisation of Individual Conditional Expectations (ICEs) for single features and Partial Dependence Plots (PDPs) for single features and pairs of features. For the use of these methods, in each problem, the best model was selected from all the models generated in the ten iterations, except for the feature recurrence determination in which all iterations were considered.

Initially, the recurrence of each of the selected features in the multiple models obtained throughout the CV over the ten iterations was determined. This step was performed to analyse which set of features stands out. The results were plotted in a histogram with the frequency of each selected feature.

The second method applied was PFI. This method measures the importance of each feature by analysing the reduction in model performance after its permutation, that is, after replacing the values of that feature with random noise [14]. Once again, the metric chosen to evaluate the model's performance was the F1-score, so the performance reduction is given by the difference between the F1-score achieved with the original feature and the F1-score achieved with the permuted feature. This procedure was repeated twenty times, and the result returned is the average value of the importance of each input feature. The higher this value is, the more important the feature is. However, it is important to note that the importance obtained through this method reflects the importance of each feature to the specific model and not the intrinsic predictive value of the feature itself [136].

Finally, to visualise the dependency relationship between one or more features and the prediction result, ICE plots and PDPs were produced using the *PartialDependenceDisplay* function. These plots were developed for the features and feature pairs with the highest importance values in the previous method.

4.4.2 Local explanations

Sometimes the global representation of model behaviour does not reflect the best local behaviour in the prediction process. Thus, analysing a single instance can lead to clearer, and more accurate explanations than global explanations [14, 137]. Furthermore, it is possible to analyse specific instances of correct and incorrect classifications.

Regarding local methods, the Local Interpretable Model-Agnostic Explanations (LIME) method was used to create specific explanations for the most representative samples and for well- and misclassified samples. The explanatory model was created using the *LimeTabularExplainer* function with the ML model and training data. This model was applied to samples from the test set to obtain the local explanations. Note also that the optimisation of the kernel parameter, which significantly influences the results, was not performed; instead, the default value was used. This decision was made because the lack of knowledge about the features' meaning makes it impossible to analyse whether the explanations make sense.

The Submodular Pick-LIME (SP-LIME) algorithm was used to identify the most representative samples. This algorithm proposed by Ribeiro et al. [138] combines local explanations to explain the model globally. The SP-LIME algorithm selects through a greedy approach a data set whose explanations (generated by LIME) are non-redundant and representative of the global characteristics of the model [138]. The results were visualised through a graph and table generated by the function. The number of selected instances and the number of explanations generated were set to 50 and 3, respectively.

The selection of correctly classified and misclassified samples followed the structure illustrated in Figure 4.11. This step aims to understand in more detail the behaviour of the models in both cases since, in the medical field, it is very important to understand the reasons why the model misclassifies a given sample.

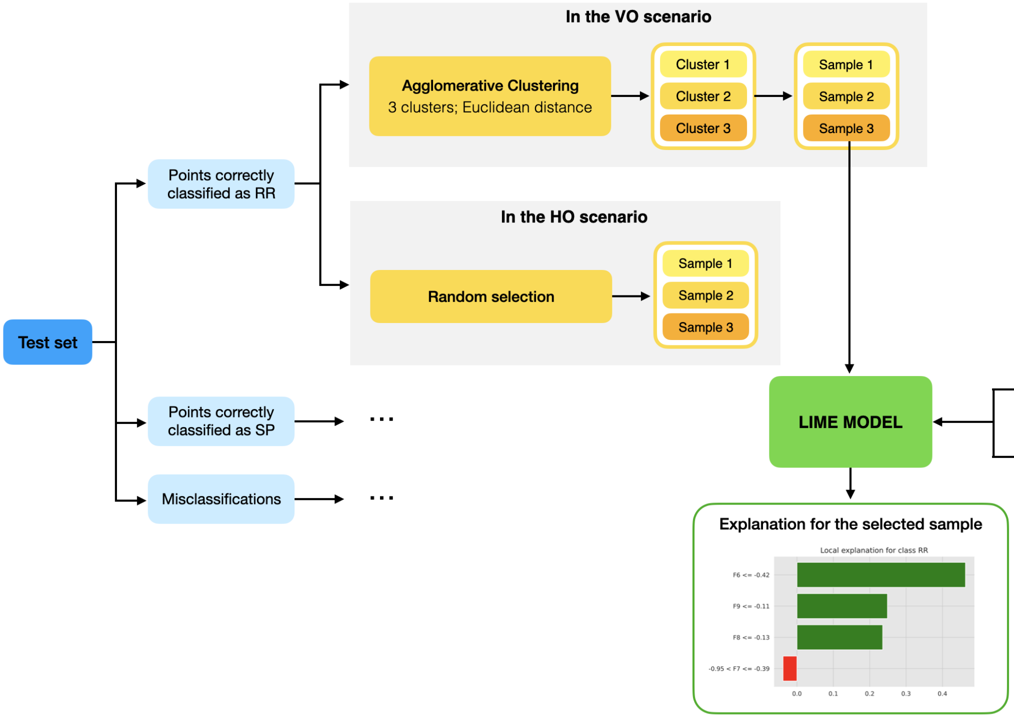


Figure 4.11: Process of selecting representative data points and applying the LIME method.

To select these samples, the test dataset was first divided into points correctly classified as progress to the SP course (class 1), points correctly classified as RR (class 0), and misclassifications.

In the VO scenario, Agglomerative clustering was applied to each of these groups. It is a bottom-up hierarchical clustering strategy in which each observation starts in its own cluster, and the two clusters with the smallest distance are recursively merged until a satisfactory final cluster is produced [139]. This step was applied to create three clusters based on the similarity between the points using Euclidean distance to compare them. In each cluster, a point was randomly selected to generate an explanation. The results are displayed in a graph and table as in the SP-LIME method.

On the other hand, in the HO scenario, it was impossible to apply clustering techniques because the NN input has a 3D shape. In this case, the three samples were randomly selected within each group.

5

Results

This chapter presents the results obtained by the methodology described in the previous chapter. In Section 5.1, the results of the different classifiers in the Visited-Oriented (VO) and History-Oriented (HO) approaches are presented. Subsequently, in Section 5.2, the explanations produced by the different methods for the prediction models are shown.

5.1 Classification

This section presents the classification results obtained in Secondary Progressive (SP) prediction at 180, 360 and 720 days for the six datasets. It starts by presenting the results obtained for the k-nearest neighbours (KNN), Random Forest (RF), AdaBoost (AB), and linear and non-linear Support Vector Machines (SVM) classifiers in the VO approach (section 5.1.1). Then, section 5.1.2 shows the results of the HO approach using Long Short-Term Memory (LSTM) models.

The initial phase of the work aimed to replicate the study presented by Seccia et al. [1] and was followed by efforts to improve the performance and explainability of the models. Regarding the performance of the models, the best results for each of the problems are in bold.

5.1.1 Visited-Oriented approach

The methodology described in the previous chapter was applied to the six datasets. Initially, the grid-search procedure was performed to select the training parameters for each one of the classifiers. This grid-search selection was based on the F1-score metric. All the chosen hyperparameters for the different classifiers in the six datasets are shown in Table 5.1. The results of the multiple classifiers with the optimal hyperparameter set are presented in Table 5.2.

Table 5.1: Optimal hyperparameters found for each classifier that produce the best value for the F1-score metric.

		FK 180	FK 360	FK 720	RK 180	RK 360	RK 720
KNN	*K* neighbors	17	19	19	19	17	19
AB	No of estimators	5	3	1	10	15	15
RF	No of trees	50	10	50	10	50	15
	Tree depth	2	2	1	5	1	2
Linear SVM	C	0.01	0.01	0.01	0.001	0.001	0.001
Non-linear SVM	C	10	100	100	1	1	1
	Gamma	0.0001	0.0001	0.0001	0.001	0.001	0.001

Table 5.2: Results of the performance obtained for the classification problem in the VO approach.

Classifier	Feature-keeping (FK)					Record-keeping (RK)				
	Accuracy	Precision	Recall	Specificity	F1-score	Accuracy	Precision	Recall	Specificity	F1-score
180 days										
KNN	90,266	5,126	61,111	90,510	9,457	88,959	8,73	73,575	89,175	15,607
AB	82,381	4,279	94,444	82,28	8,188	81,409	6,313	89,614	81,293	11,796
RF	82,277	4,212	93,333	82,184	8,061	82,417	6,627	89,179	82,322	12,338
Linear SVM	89,591	5,344	68,889	89,765	**9,917**	96,165	17,835	48,744	96,832	**26,102**
Non-linear SVM	87,492	4,994	77,778	87,573	9,384	87,708	8,77	83,575	87,766	15,873
360 days										
KNN	88,786	4,523	58,108	89,059	8,39	89,697	9,272	69,227	89,999	16,352
AB	80,873	3,92	88,108	80,809	7,506	79,673	5,961	87,826	79,552	11,163
RF	81,547	3,882	83,784	81,527	7,419	82,83	6,45	80	82,872	11,937
Linear SVM	90,074	5,341	61,351	90,329	**9,827**	97,005	23,153	45,507	97,765	**30,675**
Non-linear SVM	86,594	4,283	66,486	86,773	8,046	93,465	13,436	64,203	93,896	22,221
720 days										
KNN	91,545	5,814	52,162	91,92	10,458	90,831	10,940	67,681	91,210	18,834
AB	81,991	3,949	77,568	82,033	7,516	80,253	6,451	85,7	80,166	11,999
RF	82,432	3,697	70,27	82,548	7,024	85,011	7,678	77,44	85,132	13,97
Linear SVM	91,326	6,061	56,216	91,66	**10,939**	97,327	28,625	46,86	98,132	**35,529**
Non-linear SVM	87	4,638	65,135	87,208	8,657	94,048	15,704	63,816	94,531	25,204

As can be seen in Table 5.2, the KNN and linear and non-linear SVM classifiers achieved the best performance. Although recall assumes relatively good values, the results obtained for the F1-score metric were quite low due to low precision, which is lower in the Feature-keeping (FK) approach.

The best results were achieved using the linear SVM classifier in all approaches and time windows. This classifier leads to lower recall values than the others, but the overall F1-score achieves the highest values since it balances precision and recall.

The feature selection process was then applied to the Machine Learning (ML) models with the linear SVM classifier. Table 5.3 shows the best results achieved for each problem, and the remaining results can be found in Appendix A.1.

Table 5.3: Results of the best performance obtained for the linear SVM classifier with the feature selection step in the VO approach.

Days	Feature selection method	Accuracy	Precision	Recall	Specificity	F1-score
Feature-keeping (FK)						
180	**LASSO regression** $\lambda = 0{,}005$	91,885	6,735	68,056	92,084	12,256
360	**Pearson correlation** $n = 5$	93,137	6,99	55,135	93,474	12,402
720	**Pearson correlation** $n = 5$	93,979	8,769	57,027	94,331	15,195
Record-keeping (RK)						
180	**LASSO regression** $\lambda = 0{,}003$	96,955	21,15	43,671	97,705	28,487
360	**LASSO regression** $\lambda = 0{,}005$	97,583	27,931	41,884	98,404	33,506
720	**LASSO regression** $\lambda = 0{,}005$	97,754	33,333	42,85	98,63	37,486

Comparing the results obtained with and without the feature selection step, it can be observed that, overall, the F1-score value increases slightly when only the most relevant features are considered. This increase is a result of the improvement in the precision value.

On the other hand, when comparing the results between the FK and Record-keeping (RK) datasets, it is noticed that the values of all metrics except recall are higher for the RK dataset. This difference is most noticeable in precision and means that increasing the number of samples leads to a decrease in the number of patients misclassified as SP (False Positives (FP)). Contrarily, the recall value is lower in the RK data. In this case, there is a higher number of False Negatives (FN), i.e., cases where a patient who will transition to the SP phase is classified in the Relapse Remitting (RR) phase. A higher number of samples improves the classification of RR status but worsens the identification of patients who will progress to the SP stage in a given time window.

By analysing the performance results for the different time windows, it is observed that the F1-score increases with the increase of the time window. Focusing on the precision and recall values, it is observed that the recall value decreased with the increase of the time window while the precision value increased. These results make sense because the larger the temporal window, the more difficult it is to predict the evolution to the SP course.

5.1.2 History-Oriented approach

Two types of inputs were applied to each of the Neural Network (NN) architectures tested, as stated in Section 4.3.4. All the results of the different neuronal network architectures can be found in sections A.2.1 and A.2.2 of the Appendix. The results obtained for input strategy 2 were significantly better. Thus, the NN architecture choice was based on the model results for this approach.

The details of the chosen models are listed in Table 5.4. Table 5.5 has the performance results achieved using the selected architectures for the two types of input data structures employed.

Table 5.4: Architecture and optimal hyperparameters for the NNs that produce the best value for the F1-score metric.

Dataset	**Model**	**Details**			
		Layers	**No. of cells**	**Dropout**	**Funtion**
FK 180	Model 4	Masking	1	—	—
		LSTM	10	0.2	—
		Dense	1	—	Sigmoid
FK 360	Model 6	Masking	1	—	—
		LSTM	20	0.2	—
		LSTM	10	0.2	—
		Dense	1	—	Sigmoid
FK 720	Model 3	Masking	1	—	—
		LSTM	8	0.2	—
		Dense	1	—	Sigmoid
RK 180	Model 2	Masking	1	—	—
		LSTM	6	0.2	—
		Dense	1	—	Sigmoid
RK 360	Model 4	Masking	1	—	—
		LSTM	10	0.2	—
		Dense	1	—	Sigmoid
RK 720	Model 4	Masking	1	—	—
		LSTM	10	0.2	—
		Dense	1	—	Sigmoid

Table 5.5: Results of the performance obtained for the classification problem in the HO approach.

Days	Input strategy 1					Input strategy 2				
	Accuracy	**Precision**	**Recall**	**Specificity**	**F1-score**	**Accuracy**	**Precision**	**Recall**	**Specificity**	**F1-score**
Feature-keeping (FK)										
180	97,465	10,589	20,909	98,12	12,497	94,243	32,371	17,57	98,143	**22,019**
360	64,584	2,301	82,5	64,41	4,472	95,325	63,744	34,777	98,671	**43,623**
720	98,016	16,926	19,167	98,808	17,178	92,89	43,512	29,218	97,332	**34,312**
Record-keeping (RK)										
180	91,596	10,507	64,921	91,975	18,035	96,491	87,087	66,153	99,113	**74,655**
360	90,851	10,067	64,603	91,24	17,376	96,066	82,646	69,685	98,588	**75,461**
720	87,864	9,288	71,905	88,127	16,39	95,345	83,086	67,543	98,448	**74,399**

The results show that by considering each patient's records as a time series, it becomes very advantageous to add the information from each record to the information from previous records. Therefore, the feature selection step was applied to the models with input strategy 2, and the best results are shown in Table 5.6. The remaining results are presented in sections A.2.3 and A.2.4 of the Appendix.

Table 5.6: Results of the best performance obtained with the feature selection step in the HO approach.

Days	Feature selection method	Accuracy	Precision	Recall	Specificity	F1-score
Feature-keeping (FK)						
180	**LASSO regression** $\lambda = 0{,}005$	95,397	56,402	35,705	98,407	41,596
360	**LASSO regression** $\lambda = 0{,}005$	95,458	70,103	40,007	98,844	49,330
720	**LASSO regression** $\lambda = 0{,}005$	93,510	48,021	43,539	96,805	45,040
Record-keeping (RK)						
180	**LASSO regression** $\lambda = 0{,}001$	95,967	84,869	62,414	98,982	71,574
360	**LASSO regression** $\lambda = 0{,}003$	96,383	86,242	70,678	98,888	77,332
720	**LASSO regression** $\lambda = 0{,}003$	95,862	86,492	70,011	98,748	77,105

The performances presented in Tables 5.5 and 5.6 reflect the ability of NNs to learn and model complex relationships and generalise. In this approach, the performances were much better in all the considered problems, reaching a F1-score higher than 70% in the three RK datasets. The significant difference in the precision and F1-score values between the RK and FK datasets reinforces the need for a high number of samples. Moreover, in both data types, the performance obtained in the three time windows is very similar, although it is slightly worse in the shorter time window (180 days).

5.2 Produced explanations

This section presents the global and local explanations produced for the best model of each problem from the different approaches mentioned in the previous section. This part of the work aims to interpret the models created to understand the behaviour of the models, which features it relies on the most, and how each feature affects the final decision.

5.2.1 Visited-Oriented approach

5.2.1.1 Global explanations

The models' behaviour analysis began by determining each feature's recurrence in the different iterations, i.e., the number of times a feature was selected for the classifier input.

The same sets of features were selected in all iterations of each problem, and these sets are listed in Table 5.7. Features F6, F11 and F12 are selected in all models. On the other hand, features F8 and F9 have a strong presence in the FK approach, while in the RK approach, features F10 and F11 are always selected.

Table 5.7: Selected features of each dataset in the VO approach.

Days	**Dataset**	
	Feature-keeping (FK)	**Record-keeping (RK)**
180	F6, F11, F12	F6, F10, F11, F12
360	F6, F8, F9, F11, F12	F6, F10, F11, F12
720	F6, F8, F9, F11, F12	F6, F10, F11, F12

It is important to remind that feature F6 corresponds to the Expanded disability status scale (EDSS) value and features F8 and F9 are Magnetic Resonance Imaging (MRI) and liquor features and are not present in the RK datasets. Also, the meaning of features F10, F11 and F12 is unknown. It is only known that they can be features with clinical information or information about the treatment drugs, but these concepts are quite distinct.

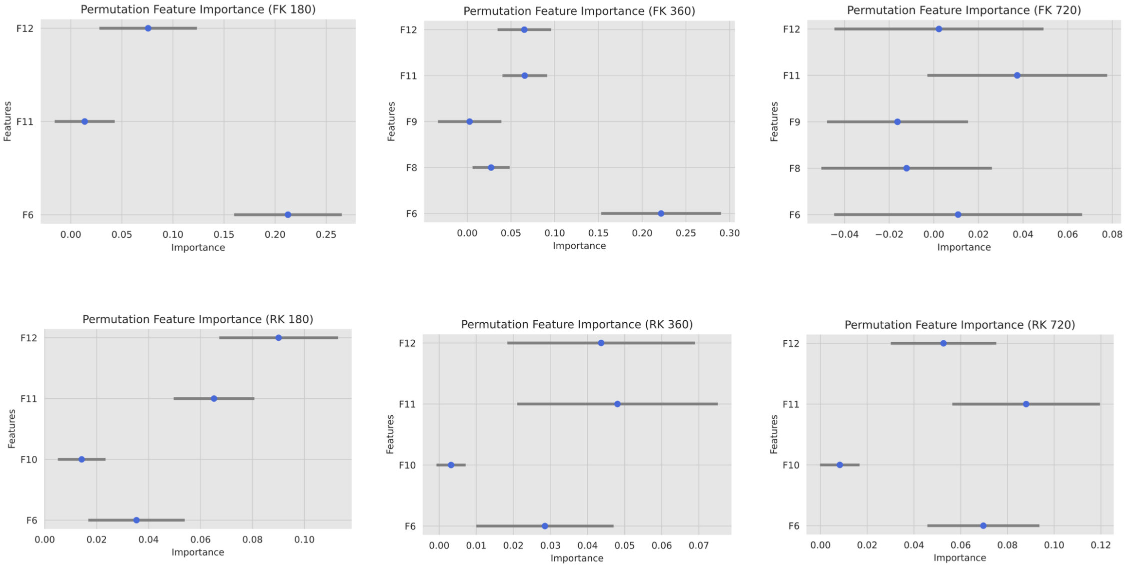


Figure 5.1: PFI for the best model of the FK (top) and RK (bottom) problems of the VO approach.

The next step in the interpretation process was calculating the Permutation Feature Importance (PFI) for each problem. Figure 5.1 shows the average values of permutation importance for each feature and their standard deviation. The interpretation of the displayed graphs is quite simple. A feature is more important the higher its calculated importance value is. In these cases, the features with the highest importance values are features F6, F11 and F12.

Some features have negative importance values, which means that in these cases, the F1-Score values obtained with the noisy data were higher than those obtained with the original data. In these cases, it can be considered that the importance of the feature is approximately zero.

The high standard deviation illustrated confirms the randomness associated with the importance calculation in the multiple iterations. The high variability causes more uncertainty in the analysis of the results and, consequently, in the conclusions drawn. Furthermore, this randomness can result in the creation of unrealistic permutation samples that significantly impact the results [14].

Despite these limitations, this algorithm shows how features contribute to the prediction in a simple and understandable way. Another major advantage is the ability to consider the interactions between features when determining importance.

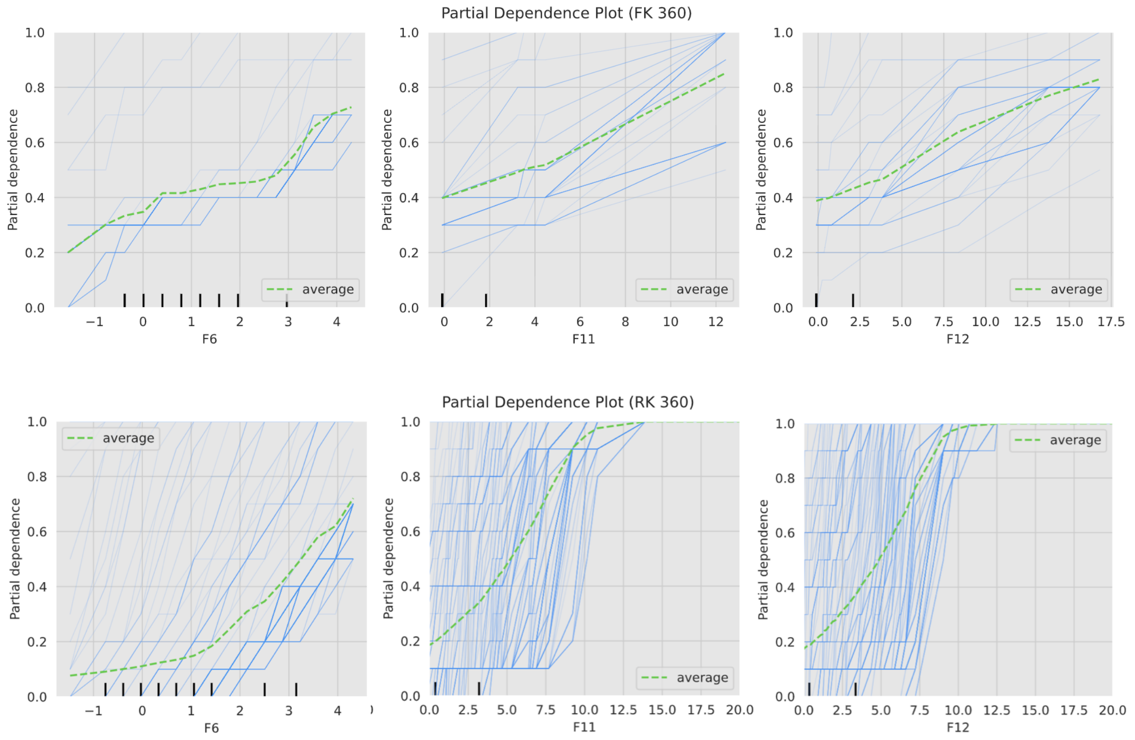


Figure 5.2: PDP (green) and ICE (blue) plots for the most important features of the FK 360 (top) and RK 360 (bottom) problems of the VO approach.

Next, Partial Dependence Plots (PDPs) and Individual Conditional Expectation (ICE) graphs were plotted to understand the marginal effect of the most important features (F6, F11, and F12) on the classification result. The dependency relationship between the target response and the different features is very similar in the three time windows of each data type, so only one example of each approach is presented. Figure 5.2 shows the PDPs of the features with the highest importance for problems FK 360 and RK 360.

The dashed green line shows the average relationship (PDP), while each blue line (ICE) is the dependence for each sample separately. Since this is a binary classification problem, the target answer is given as a probability of the positive class.

Figure 5.2 reveals a positive correlation between the features presented and the prediction value; that is, the higher its value, the higher the probability of being classified as a transition to the SP course. In the case of features F11 and F12, it is clear that there is an almost linear relationship, whose slope is significantly higher for the RK dataset. In the latter, for values of F11 and F12 greater than 10, the probability is 1.

There is some heterogeneity in the ICE lines of the different samples. This dispersion is most noticeable for features of the RK dataset. For the FK features, the presence of thicker blue lines indicates that the behaviour of the different samples is very similar between them. Overall, the behaviour and the relationship with the target response are similar to that of PDP.

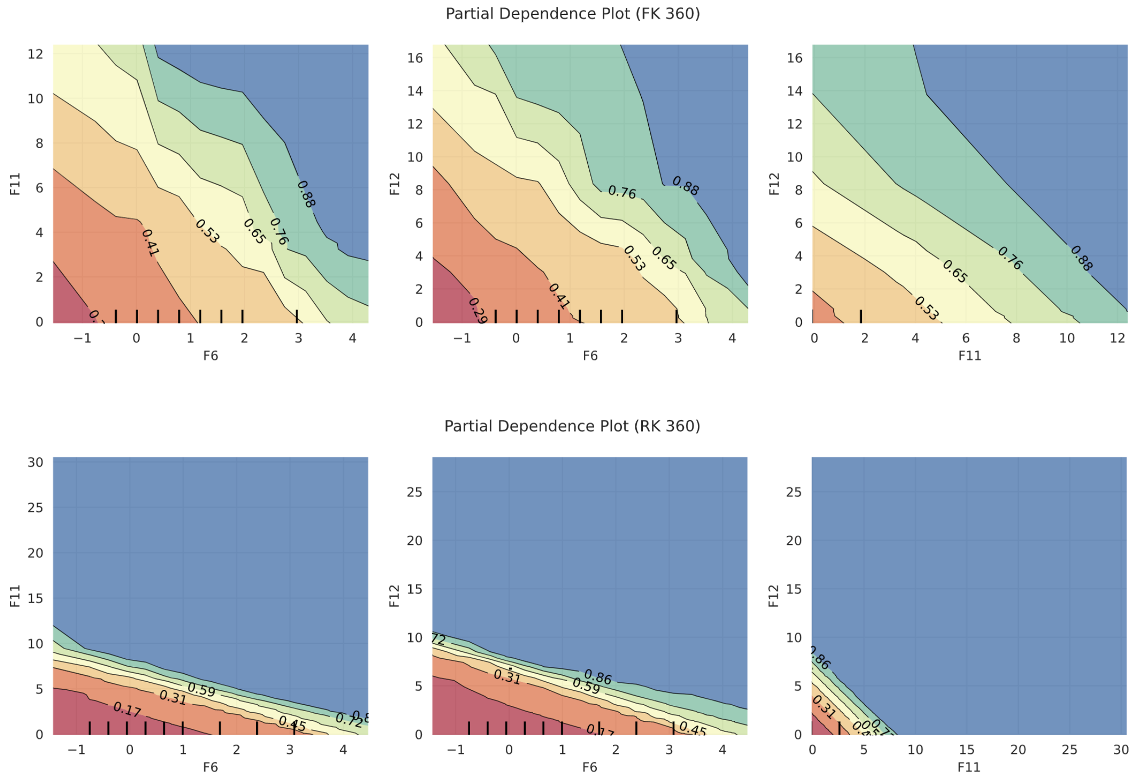


Figure 5.3: PDPs for pairs of most important features of the FK 360 (top) and RK 360 (bottom) problems of the VO approach.

Figure 5.3 shows the PDPs for the feature pairs with the highest importance and represents how the two features interact for the final prediction. The interaction between the two features has a positive linear influence on the prediction since the higher both values are, the higher the probability.

In the RK dataset, the PDPs show the strong influence of features F11 and F12 in predicting the transition to the SP course. For values higher than ten, the classification is practically independent of F6.

The analysis of PDPs is intuitive and quickly understandable. It is possible to analyse the effect of a given feature on the predicted outcome, compare the behaviour of different models and evaluate their consistency. Comparing the representation of the impact of features on the target outcome to current clinical knowledge helps to determine whether the behaviour of features is the expected one. In addition, the inclusion of ICE curves allows the exposure of heterogeneous relationships [14].

The biggest problem of these methods is the independence assumption, that is, the assumption that features are not correlated with others in the partial dependency calculation. If this is not the case, the dependency points created may be invalid. Furthermore, it is impossible to interpret the partial dependency between more than two features [14].

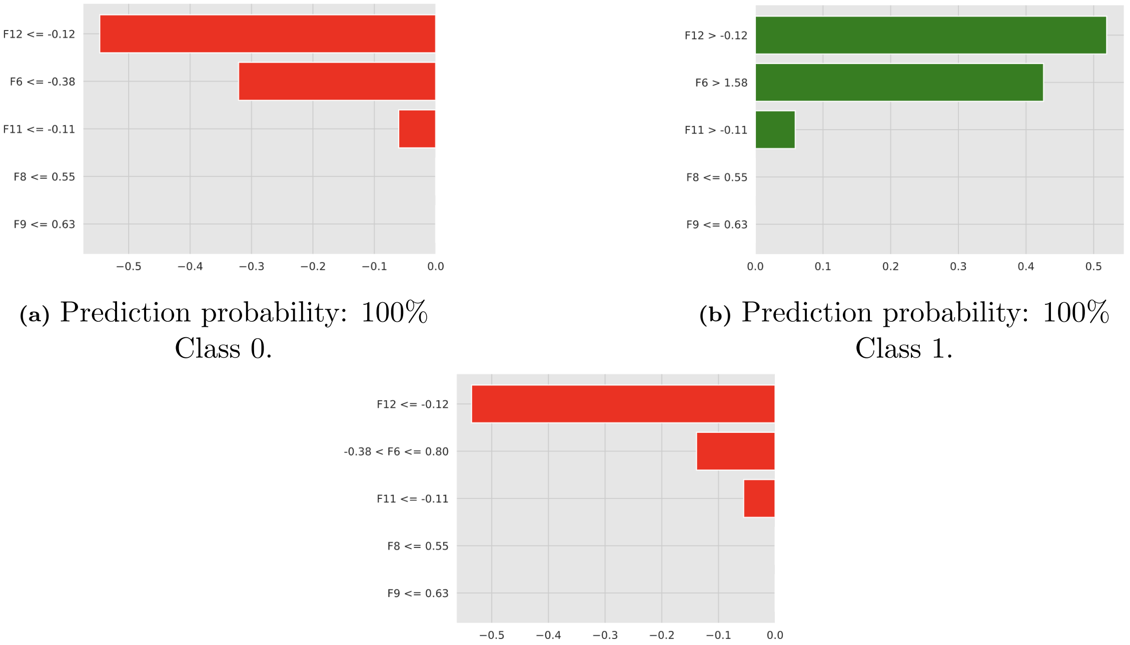
5.2.1.2 Local explanations

The results obtained for the global methods presented earlier helped to get an idea of the overall behaviour of the models. The Submodular Pick-LIME (SP-LIME) algorithm was used to select the three most relevant instances of each problem and generate the Local Interpretable Model-Agnostic Explanations (LIME) explanations.

The work was developed with six different datasets that have a high amount of samples with redundant information. Only the explanations for problems FK 720 and RK 720 will be presented since they showed the best classification results in each data type (Table 5.3).

Figures 5.4 and 5.5 show the sets of explanations chosen by the SP-LIME for both problems FK 720 and RK 720. The bar charts explain the reason for the classification, representing the weight of each feature in the result; the higher its value, the longer the bar. Positive weights (green) indicate that the features promote the evolution to the SP course (class 1), while negative weights (red) mean that the features influenced the prognosis for the Multiple Sclerosis (MS) RR course (class 0).

Explanations a) and b) of Figures 5.4 and 5.5 show extreme cases in which the model is 100% confident that the classes are 0 and 1, respectively. In these cases, the features that contributed most to increasing the probability of the predicted class were F6, F11, and F12. In problem FK 720, it is observed that features F8 and F9 did not influence the model prediction in any of the three cases. Contrarily, in problem RK 720, overall, all features somehow influence the prediction probability. In the case of Figure 5.5c, features F11 and F12 contributed to the prediction being class 0, although the remaining features voted for class 1.

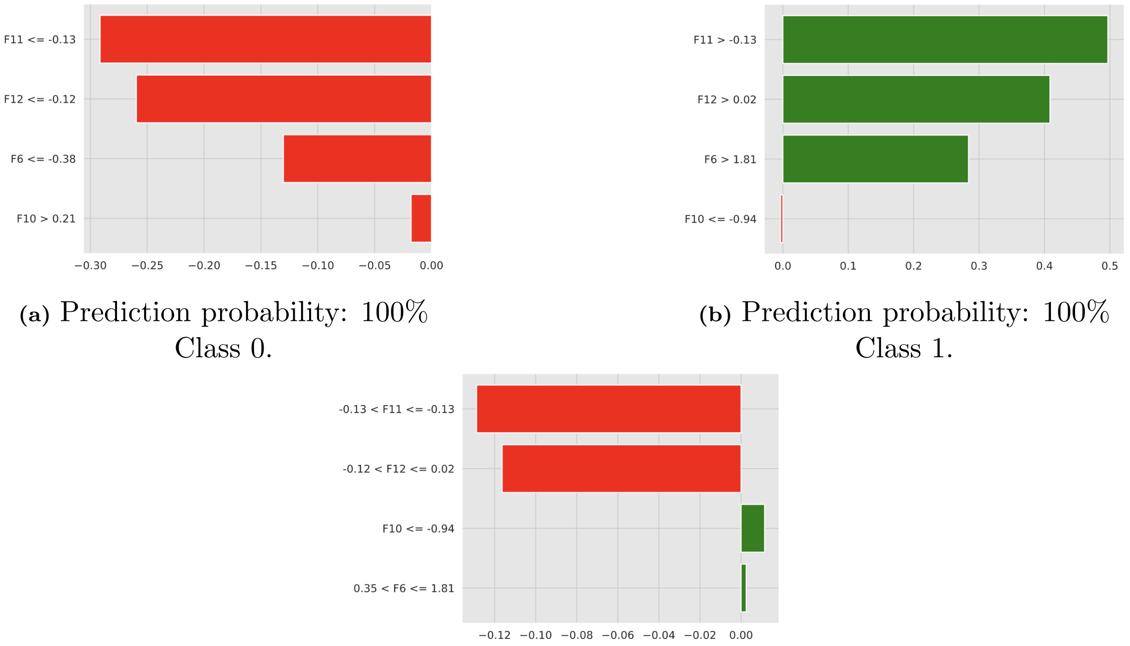


(a) Prediction probability: 100%
Class 0.

(b) Prediction probability: 100%
Class 1.

(c) Prediction probability: 90%
Class 0.

Figure 5.4: LIME explanations generated by the SP-LIME method with the best model from the FK 720 problem of the VO approach.



(a) Prediction probability: 100%
Class 0.

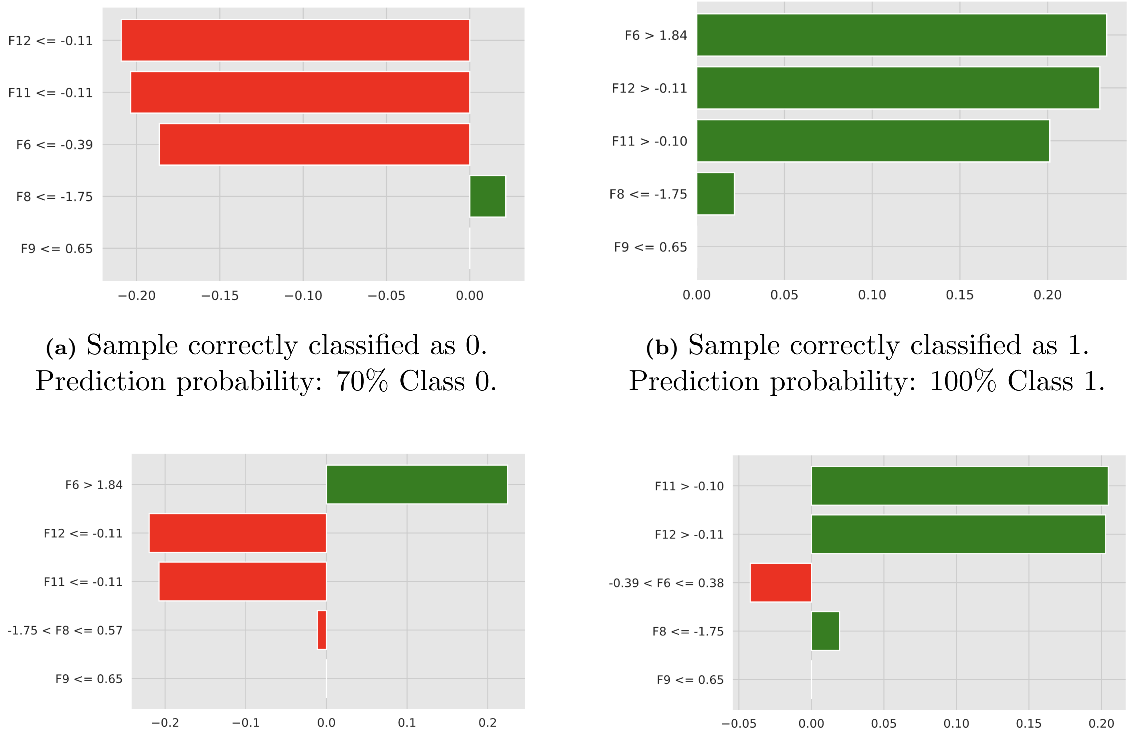
(b) Prediction probability: 100%
Class 1.

(c) Prediction probability: 80%
Class 0.

Figure 5.5: LIME explanations generated by the SP-LIME method with the best model from the RK 720 problem of the VO approach.

First, for the FK 720 problem (Figure 5.6), it can be seen that values of F11 and F12 less than -0.11 strongly support the prediction of the samples as being class 0, and conversely, for higher values, these features weigh on the classification of class 1. Misclassifications are based on the dominance of these features. In these cases, although feature F6 supports the prediction of the sample for the correct class, it is not as dominant as the other features. Furthermore, it is observed that feature F9 does not influence the prediction.

Similar situations are observed in problem RK 720 (Figure 5.7); that is, the weights assigned to features F11 and F12 underlie the outcome of the classifications, and feature F6 always supports the correct class. In this case, feature F10 supports the classification very little.



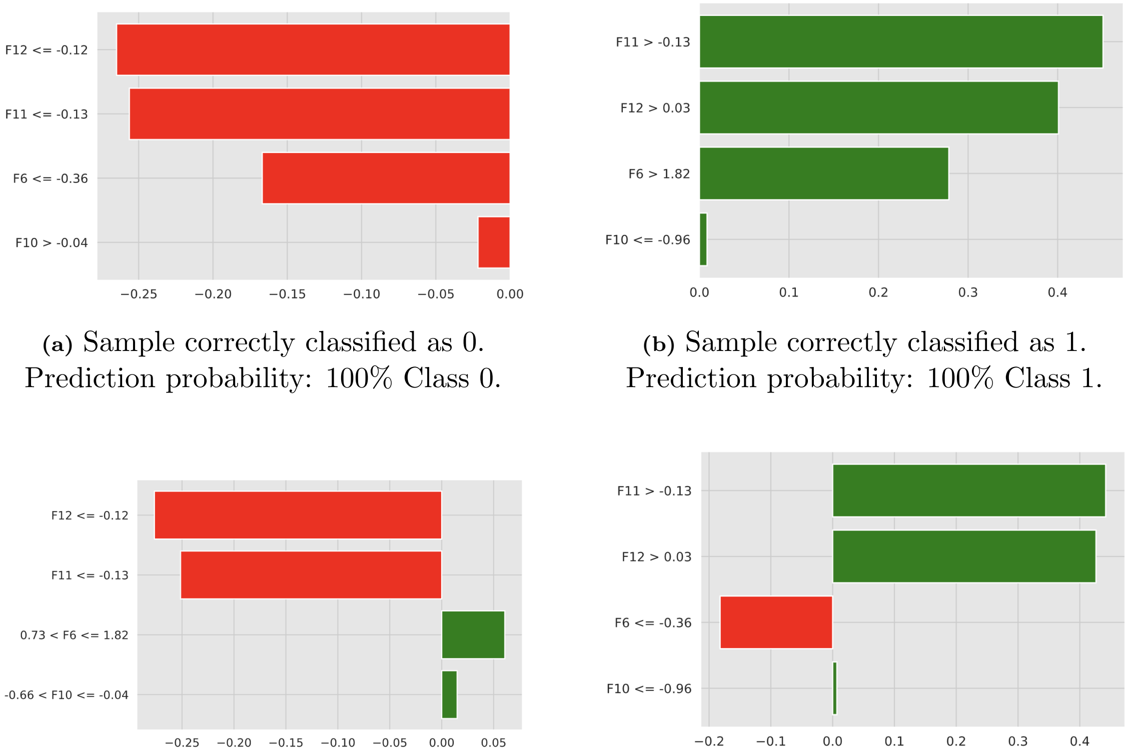
(a) Sample correctly classified as 0. Prediction probability: 70% Class 0.

(b) Sample correctly classified as 1. Prediction probability: 100% Class 1.

(c) Misclassified sample. Prediction probability: 60% Class 0.

(d) Misclassified sample. Prediction probability: 90% Class 1.

Figure 5.6: LIME explanations for examples of the three classification groups from the best model of the FK 720 problem of the VO approach.



(a) Sample correctly classified as 0. Prediction probability: 100% Class 0.

(b) Sample correctly classified as 1. Prediction probability: 100% Class 1.

(c) Misclassified sample. Prediction probability: 85% Class 0.

(d) Misclassified sample. Prediction probability: 90% Class 1.

Figure 5.7: LIME explanations for examples of the three classification groups from the best model of the RK 720 problem of the VO approach.

These local explanations produced are simple and do not influence model performance. Thus, through LIME explanations, the clinician can analyse model behaviour and understand the local impact of a particular feature when the others are in a specific range.

In this case, from the instances selected for analysis, it is possible to understand the behaviour of the best models of the RK 720 and FK 720 problems and to see that the samples with higher values of F6, F11, and F12 tend to be classified as cases that will evolve to SP in the 720-day time window.

However, it is important to note that even though LIME helps increase the explainability of the model, it is still a developing method with several limitations. The main obstacles are the instability of the explanations and the difficulty of the method in defining the local neighbourhood of the selected sample [14, 140]. In addition, improvements in the user experience are needed. An example of a problem is the incomplete explanation of the relationship between the prediction probability and the feature probabilities graph presented [140].

5.2.2 History-Oriented approach

5.2.2.1 Global explanations

In the HO approach, the same methodology was followed as in the previous approach. Initially, the recurrence of each feature over the ten runs in the different problems was determined. In this case, the selected features differ between iterations, and their frequency is illustrated in Figure 5.8.

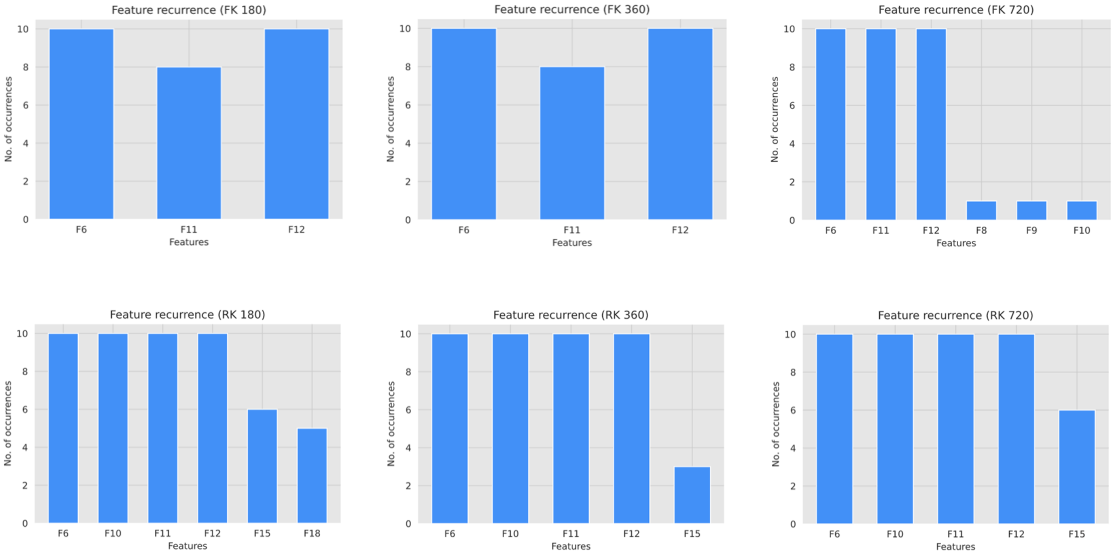


Figure 5.8: Feature recurrence for each problem of the HO approach.

As with the VO approach, the variable recurrence results show that the feature sets [F6, F11, F12] and [F6, F10, F11, F12] are the most recurrent in the FK and RK datasets, respectively. The best models were obtained with the feature sets present in Table 5.8 and will be the ones used to generate the remaining explanations.

Table 5.8: Selected features of each dataset in the HO approach.

Days	Dataset	
	Feature-keeping (FK)	**Record-keeping (RK)**
180	F6, F11, F12	F6, F10, F11, F12, F15, F18
360	F6, F11, F12	F6, F10, F11, F12
720	F6, F11, F12	F6, F10, F11, F12, F15

It is observed that the features F6, F11, and F12 are always selected in the best models. As mentioned earlier, feature F6 is the EDSS value, and the exact meaning of features F10, F11, F12, F15, and F18 is not known; these could be clinical information or information about drug therapies.

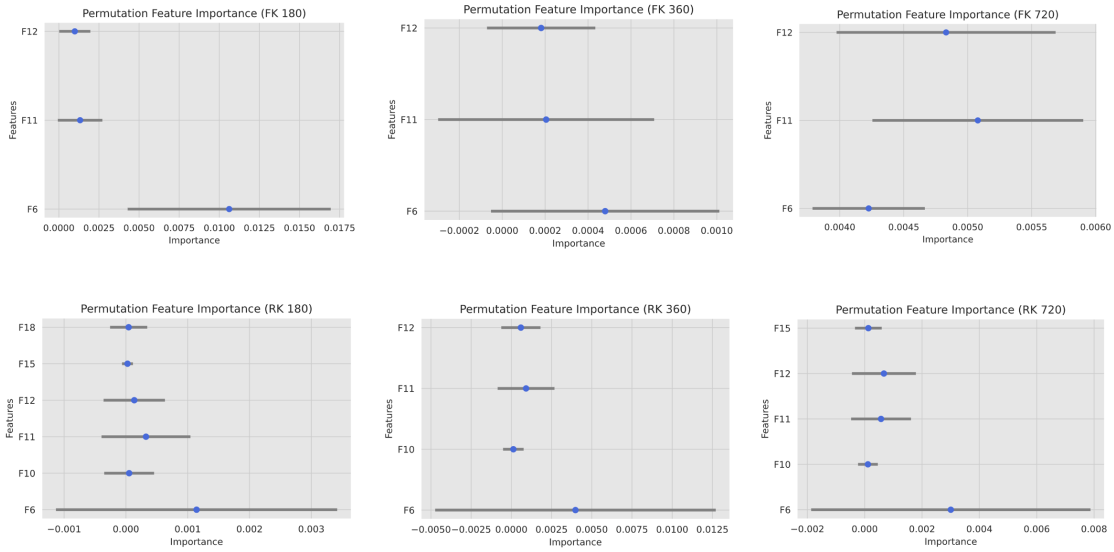


Figure 5.9: PFI for the best model of the FK (top) and RK (bottom) problems of the HO approach.

Figure 5.9 shows the permutation importance of the features used in the best models and the associated uncertainty. Again, the presence of features F6, F11, and F12 stands out, and for the RK datasets, these three features have higher importance values than the others. In this approach, the feature with the highest uncertainty is F6.

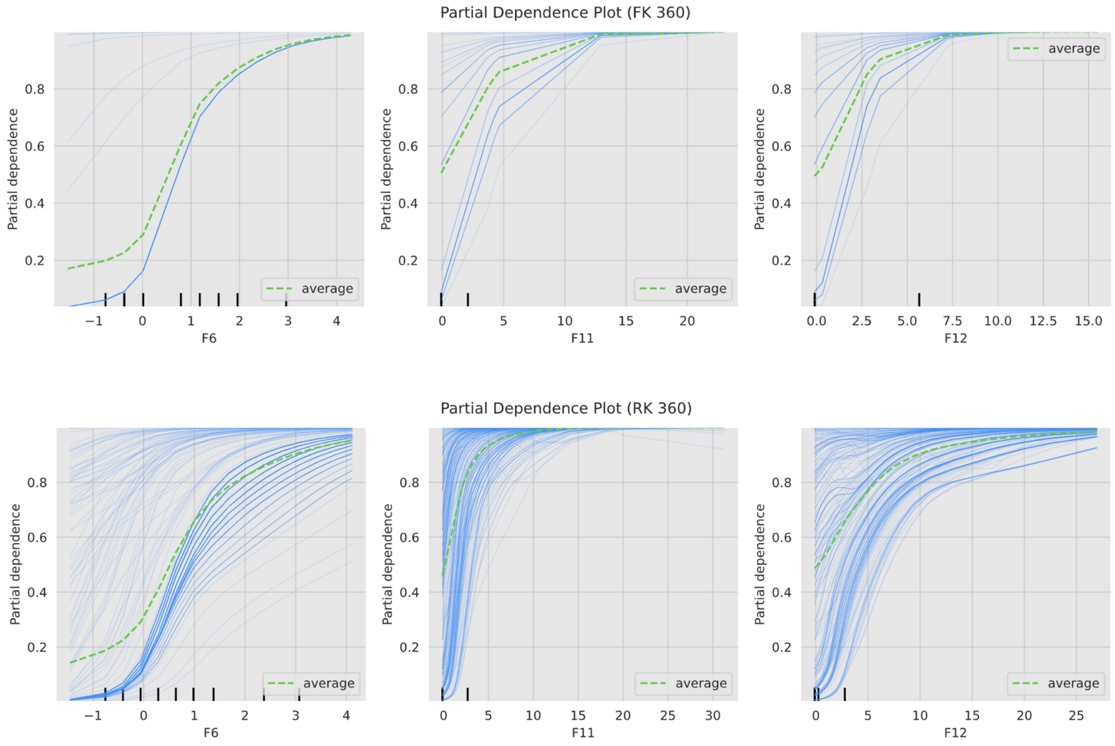
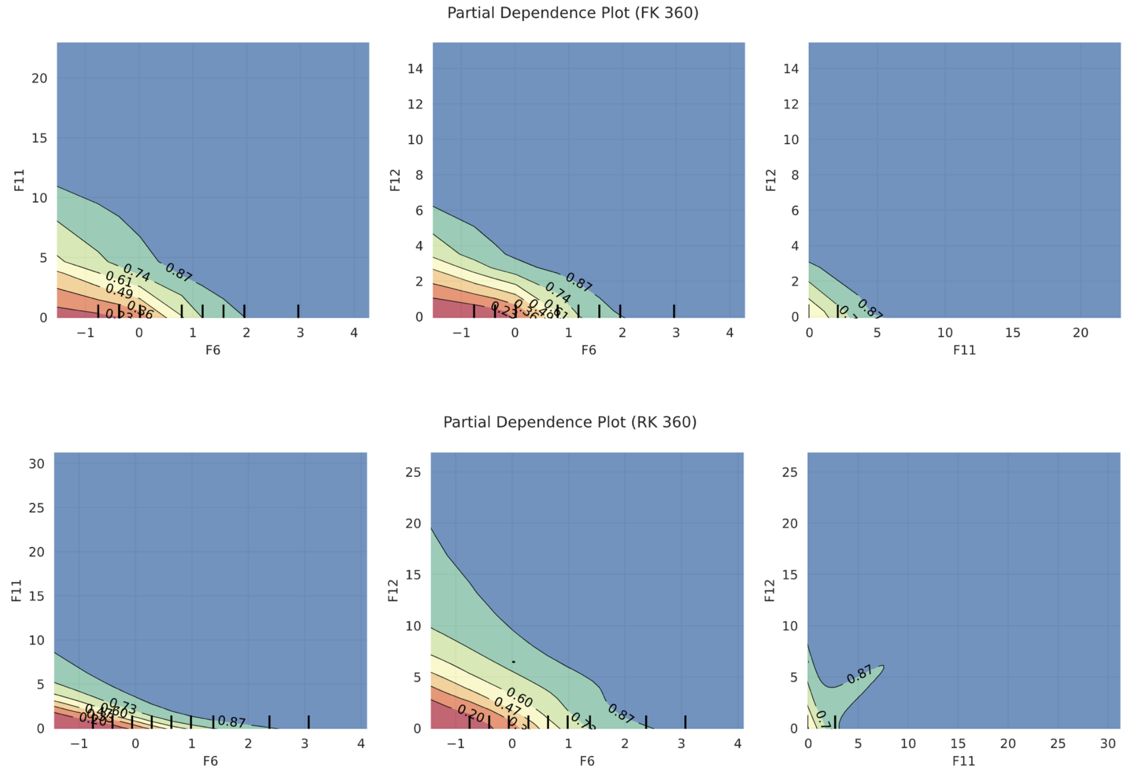


Figure 5.10: PDP (green) and ICE (blue) plots for the most important features of the FK 360 (top) and RK 360 (bottom) problems of the HO approach.

The PDPs and ICEs were then plotted for the different problems. Figure 5.10 shows the plots for the features with the highest permutation importance for the FK 360 and FK 360 problems. The relationship between feature value and dependency is similar in both problems and across features. However, for the RK datasets, there is a bigger dispersion of the ICE lines from the different samples. When observing the PDP, it becomes evident that high sample values generally indicate progression to the SP state.


Partial Dependence Plot (FK 360)

Partial Dependence Plot (RK 360)

Figure 5.11: PDPs for pairs of most important features of the FK 360 (top) and RK 360 (bottom) problems of the HO approach.

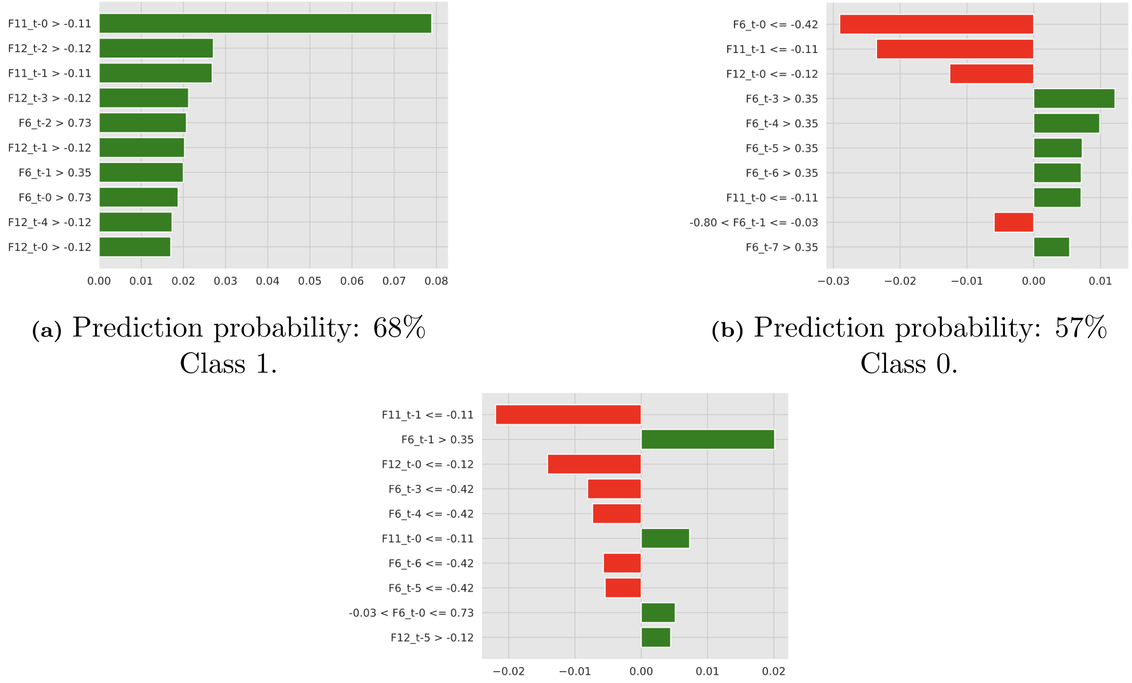
In the plots of Figure 5.11, it can be seen that the first values are able to influence the prediction result, and changing the values in a later step no longer changes the model's confidence.

5.2.2.2 Local explanations

The local explanations presented in this section are related to the problems with the best classification results, which correspond to the 360-day time window. In this step, the three most relevant instances were selected, and their explanations were generated using the SP-LIME algorithm to show how the model behaves globally.

The top ten features used in each explanation are shown in Figures 5.12 and

5.13. In the HO approach, each sample corresponds to a time series with the visits of one patient. Thus, each explanation presented considers the history of a selected patient. It can be seen that, in the different explanations, a diverse set of visits from the time series is chosen.



(a) Prediction probability: 68% Class 1.

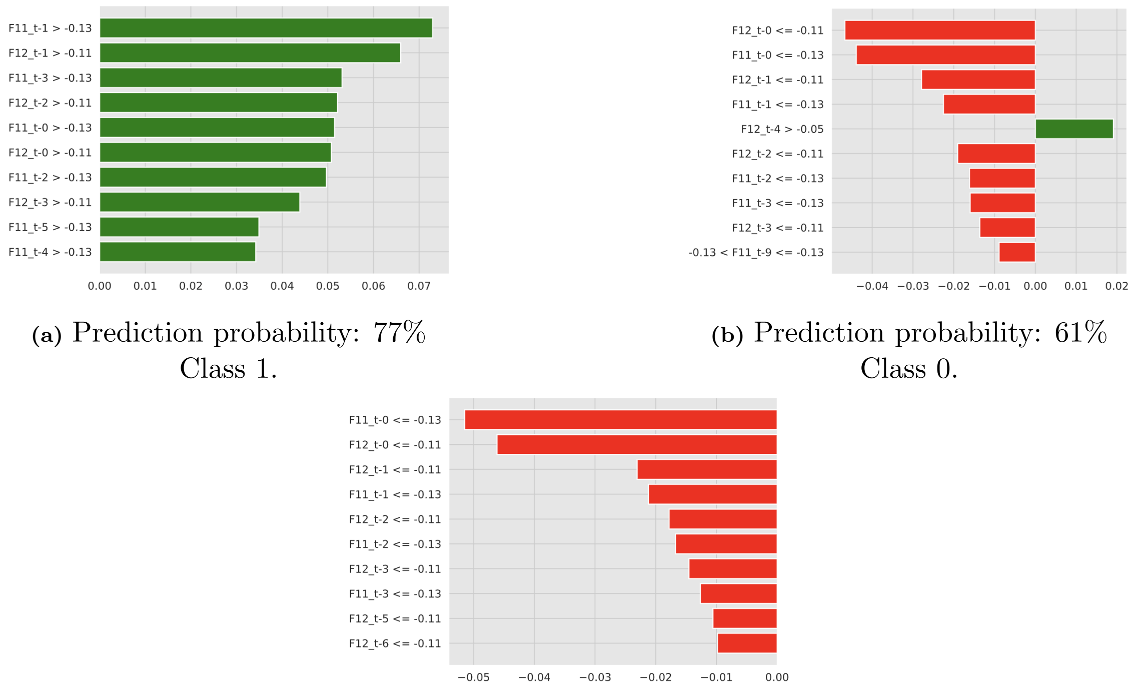
(b) Prediction probability: 57% Class 0.

(c) Prediction probability: 62% Class 0.

Figure 5.12: LIME explanations generated by the SP-LIME method with the best model from the FK 360 problem of the HO approach.

In problem FK 360 (Figure 5.12), the main common features of the three explanations are the values of F6, F11, and F12 from the last two visits (t-0 and t-1). In general, data of up to seven visits are considered. This shows that the most recent values have a greater impact on the classification task but the time series analysis influences the prediction process. Overall, values of F6, F11, and F12 greater than 0.35, -0.11, and -0.12, respectively, support class 1.

In the RK 360 problem (Figure 5.13), the most important features only include features F11 and F12 at different visits (t-0 to t-6). The impact of these features is very similar to the one of the previous problem. The values that support class 1 are $F11 > -0.11$ and $F12 > -0.13$.



(a) Prediction probability: 77% Class 1.

(b) Prediction probability: 61% Class 0.

(c) Prediction probability: 70% Class 0.

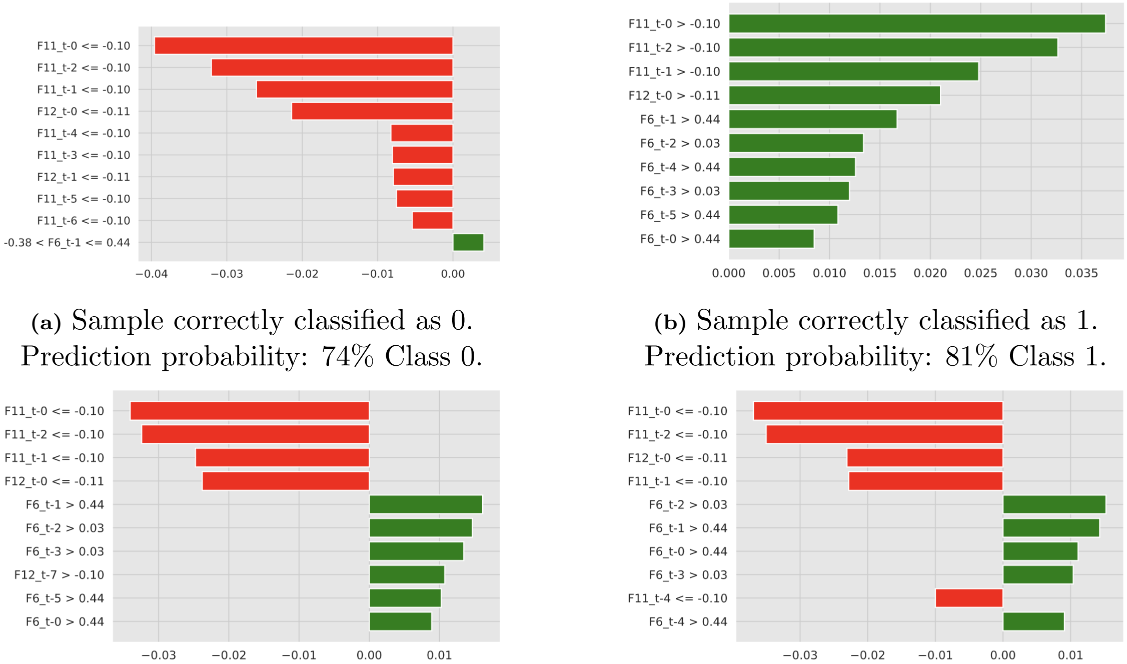
Figure 5.13: LIME explanations generated by the SP-LIME method with the best model from the RK 360 problem of the HO approach.

The second step was the creation of explanations for the specific samples. Regarding the selection of correctly and wrongly classified samples, it was not possible to apply the clustering step since the clustering algorithms of the *scikit-learn* library are not compatible with the 3D input data of the model. Thus, the samples were randomly selected from each classification group (class 1, class 0, and misclassifications).

Figures 5.14 and 5.15 show the high weight of features F11 and F12 for t-0, t-1, and t-2. In figures a) and b), the predictions are quite reliable, and practically all ten features weigh in the classification of the correct class. The model predicts classes 0 and 1 with relatively high confidence values of approximately 70%, reaching 81% in the explanation 5.14b.

Images 5.14c and 5.14d show situations where the model of the problem FK 360 classified the samples into class 0, but the actual class was 1. It is noticeable that this misclassification was based on the F11 values of the last three visits and the F12 value of the last visit. In contrast, the feature F6 supports class 1.

For problem RK 360, the explanation 5.15d shows that the most recent visits (t-0) voted wrongly for class 1 and the other features contributed to the model prediction being class 0.



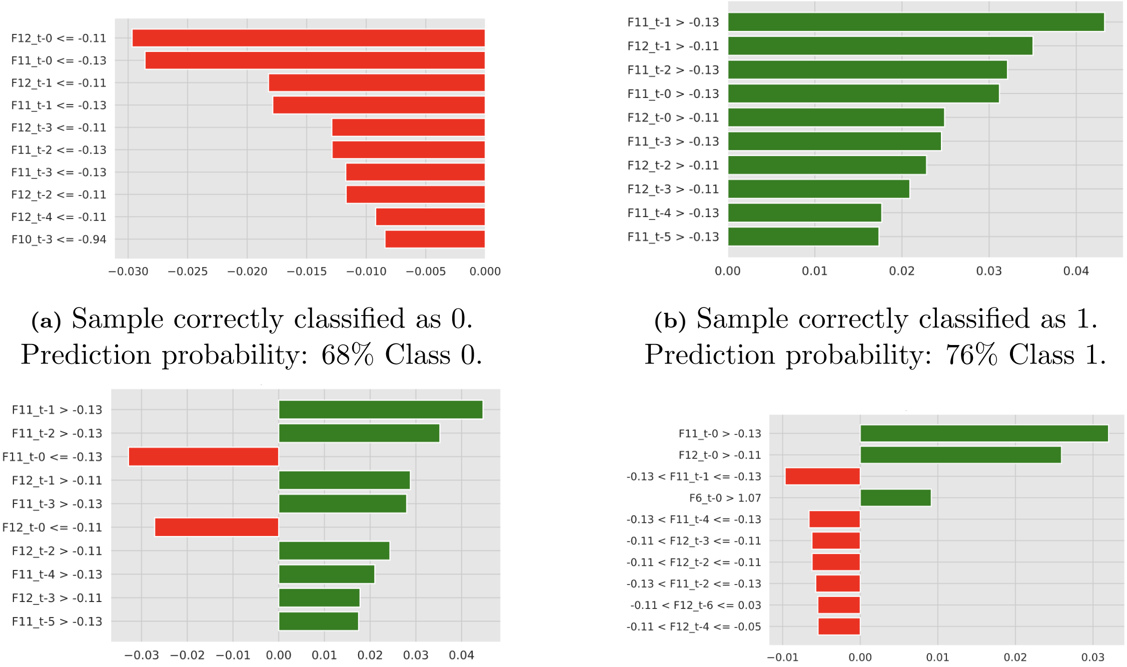
(a) Sample correctly classified as 0. Prediction probability: 74% Class 0.

(b) Sample correctly classified as 1. Prediction probability: 81% Class 1.

(c) Misclassified sample. Prediction probability: 55% Class 0.

(d) Misclassified sample. Prediction probability: 63% Class 0.

Figure 5.14: LIME explanations for examples of the three classification groups from the best model of the FK 360 problem of the HO approach.



(a) Sample correctly classified as 0. Prediction probability: 68% Class 0.

(b) Sample correctly classified as 1. Prediction probability: 76% Class 1.

(c) Misclassified sample. Prediction probability: 58% Class 1.

(d) Misclassified sample. Prediction probability: 52% Class 1.

Figure 5.15: LIME explanations for examples of the three classification groups from the best model of the RK 360 problem of the HO approach.

6 Discussion

This chapter discusses several points related to the approach used in the experimental procedure and the analysis of the results obtained. Firstly, in section 6.1, an overview of the dataset is presented. Section 6.2 is focused on the classification process developed, while section 6.3 concentrates on the explanations produced.

6.1 Dataset description

A significant limitation of this master's thesis was the dataset used. The small number of patients and samples per patient and the high number of missing data in some fields made it impossible to use the Centro Hospitalar e Universitário de Coimbra (CHUC) dataset in this project, that was previosly used in Pinto et al. [105], Oliveira et al. [141], and Sousa et al. [125]. The transformation of this dataset into time windows led to datasets with a reduced number of patients.

Therefore, it was decided to use the datasets provided by Seccia et al. [1]. These datasets have the advantage of having data that is easily collected routinely in clinical practice and includes results of neurological and imaging examinations. The use of clinical data acquired in routine visits, representing a real scenario, may lead to an easier and improved replication of this proposed methodology with datasets from different clinical centres. However, a big problem with the dataset used was that the authors removed the feature labels to protect the patients' privacy. This action was a significant limitation for the explainability step, discussed later in this chapter.

The dataset is the pillar of Machine Learning (ML) models, and its preparation is a crucial step. Different pre-processing approaches can lead to different outcomes. One of the limitations found in using this dataset is that it is already extremely preprocessed, and it is not possible to explore different strategies, such as alternative missing data imputation methods and other prediction time windows.

The pre-processing performed by the authors reduced the problem dimensionality from 200 features to 21 features. Furthermore, as mentioned before, for the elimination of missing values, the authors considered two strategies: the elimination of all records with at least one missing value while persevering the features (Feature-keeping (FK)) and the elimination of features with missing values while keeping all patient samples (Record-keeping (RK)). Different methods could be addressed in this step, such as the imputation of missing values using existing data in order to keep as many samples as possible.

The datasets are significantly imbalanced, which is typical in medical Multiple Sclerosis (MS) databases. In this case, only approximately 1% of the visits belong to patients who have developed Secondary Progressive (SP). Although the datasets have a high number of patients, about 40% of the patients contain less than five visits, and 13% have information from only one visit. This small number of samples limits the performance of the classifiers, especially in the History-Oriented (HO) approach, where a patient's visits are considered a time series, and the Long Short-Term Memory (LSTM) Neural Network (NN) needs more data than is available.

Furthermore, although the FK and RK datasets contain information regarding approximately 500 and 1500 patients, respectively, it is considered a small dataset for the complexity of this problem since the more complex the relationships, the more data is required. Thus, in this study, there was always the risk of exhausting the information present in the datasets used.

6.2 Classification

Since the problem is highly imbalanced, there was a special need to adopt specific techniques to address this challenge. The data balancing step to equalise the number of samples of each class in the training group stands out. The Bootstrap aggregation (Bagging) algorithm assumed a key role in the classification process due to its impact on improving robustness and reliability in model performance. Several metrics were also calculated using the confusion matrix to evaluate the performances and have an overview of the model's behaviour in different aspects.

The feature selection step not only decreased the training time, whose difference was most noticeable in the HO approach, but also improved the model performance. Optimising the feature selection hyperparameters made it possible to find the best subset of features in each problem.

Regarding the Visited-Oriented (VO) approach, five different classifiers were used to have a higher degree of confidence and explore the data in-depth. The selection of the classifiers was based on their properties and performance. The grid-search step for each classifier-dataset pair was fundamental to extract the classifier's highest capacity and improve the model's global performance. For all three time windows (180, 360 and 720 days), the best results were achieved with the linear Support Vector Machines (SVM) classifier.

The F1-score value is quite low, especially for the FK data, suggesting that the classifier is misclassifying many patients who will remain in the Relapse Remitting (RR) stage as progressing to SP. Using this algorithm without medical confirmation would result in the administration of more aggressive drugs unnecessarily, which could cause negative effects on patients. However, it is important to emphasize that the purpose of this ML model is not to make decisions by clinicians in isolation but to support decision-making and help in complex and challenging situations.

Looking at the HO approach, multiple NN architectures were designed and tested with the different datasets to bring out the best architecture for each problem and achieve better classification performance. The simplest tested NN did not learn the relationship of the data (underfitting), while the more complex ones adapted too well to the data (overfitting).

As mentioned earlier, a training set was selected to train the model and another set to test. While in the VO approach, the Leave One Group Out (LOGO) data partitioning method was used, in the HO approach, the simpler train-test split strategy was used (70% training and 30% testing). Thus, in each one of the ten runs, random and different sets of patients were selected for training and testing. This change was due to the time and computational limitations of applying the LOGO procedure in NN models and possibly negatively affected the performance of the LSTM model.

Despite having a small sample of patients that developed the SP course, the results obtained for the RK approach are quite satisfactory, with a F1-score higher than 70% in all three-time windows. These results support the choice of this approach in predicting the transition of the disease to the SP course, from the shortest time period (half a year), up to about two years. Detecting this transition two years in advance is very useful to adapt therapies and reverse this evolution, but it may not be enough since the effects of some therapies are quite slow [1].

Finally, comparing the results of the two different types of datasets (FK and RK), it is noticeable that it is more important to have a higher number of visits than the additional features F7, F8, and F9, whose information is about results of liquor and Magnetic Resonance Imaging (MRI) scans. However, the recall values of the VO approach show that these imaging data help identify the cases of patients who will transition to the SP course. These two different pre-processing approaches performed by Seccia allowed the evaluation of the behaviour of the models in relation to the amount of data and features available.

6.2.1 Comparative analysis with other studies

The results obtained can be compared with the performance of the papers mentioned in Section 3.1.2.3. When comparing our results with Ion-Măgineanu et al. [115], who developed a classification model for RR and SP cases using Fisher Linear Discriminant Analysis (LDA) and SVM-RBF classifiers, our model had a lower performance in terms of F1-score. Concerning Pinto et al. [105], who predicted the SP course in patients who appear to have RR MS, the results achieved are better in terms of F1-score for both approaches with the RK datasets. Nevertheless, it is important to note that these comparisons are limited since the problem approaches are quite different, as are the datasets and their time window of analysis.

On the other side, it is possible to make a direct comparison with the study by Seccia et al. [1] because the work developed in this master thesis was built on this study and used the same dataset. Tables 6.1 and 6.2 show the results presented by Seccia on each classification problem.

The short description of the procedures performed made it impossible to exactly replicate the model developed by Seccia as a starting point. However, the efforts to replicate this study, and to improve the performance of the models presented, were successful and resulted in higher F1-score values.

The main performance indicator used by Seccia to choose the best set of hyperparameters and evaluate the models was the recall metric. This leads to higher recall values than those obtained in this work for the VO approach. Differently, in this project, the key performance indicator chosen was the F1-score metric. As mentioned before, the F1-score value gives a realistic view of the model's behaviour, especially in problems of this type with a highly imbalanced dataset.

In terms of classification, the state-of-the-art was essential for selecting the classifiers used. In addition to the classifiers used by Seccia et al. [1], different classifiers that are commonly found in the literature were also tested, namely Logistic regression (LR), linear SVM, and Decision trees (DTs). It was decided to only include the linear SVM results in this dissertation since it was the only classifier that overperformed the others. Besides, it is noteworthy that the linear SVM provided the best performance results in all datasets.

Focusing on the NN models, the strategy adopted in the shape of the input data allowed the exploration of a new problem approach that significantly improved the results. By looking at each visit in the time series that includes information from previous visits, the model recalls the previously observed information stored in the memory cells. However, it is important to note that with this approach, we may be squeezing as much as possible from the data and the model. The way Seccia worked out the input data is unclear and this change may be at the root of the observed differences in performance.

Table 6.1: Comparison between the best performance values of this study and the study by Seccia et al. [1] for the classification problem in the VO approach.

Study and	Classifier	Feature-keeping (FK)					Record-keeping (RK)				
		Accuracy	Precision	Recall	Specificity	F1-score	Accuracy	Precision	Recall	Specificity	F1-score
180 days											
Seccia	**KNN**	72,6	2,4	80,6	72,6	4,7	85,6	7,4	81,2	85,7	13,6
	AB	86,1	5,6	100,0	86,0	10,6	85,4	7,9	88,9	85,3	14,5
	RF	86,5	5,8	100,0	86,4	10,9	85,1	7,9	90,8	85,0	14,6
	Non-linear SVM	86,4	5,5	94,4	86,4	10,4	87,2	8,6	85,0	87,2	15,6
This study	**KNN**	90,118	5,195	63,056	90,345	9,597	88,959	8,73	73,575	89,175	15,607
	AB	82,381	4,279	94,444	82,28	8,188	81,409	6,313	89,614	81,293	11,796
	RF	82,277	4,212	93,333	82,184	8,061	82,417	6,627	89,179	82,322	12,338
	Linear SVM	91,885	6,735	68,056	92,084	12,256	96,955	21,15	43,671	97,705	28,487
	Non-linear SVM	87,492	4,994	77,778	87,573	9,384	87,708	8,77	83,575	87,766	15,873
360 days											
Seccia	**KNN**	71,2	2,0	67,6	71,2	3,9	85,0	7,1	77,3	85,1	13,0
	AB	85,5	4,9	83,8	85,5	9,3	83,6	7,3	88,4	83,5	13,5
	RF	87,3	5,9	89,2	87,3	11,1	83,2	7,2	88,4	83,1	13,3
	Non-linear SVM	85,1	4,9	86,5	85,1	9,3	86,6	8,2	80,7	86,7	14,9
This study	**KNN**	88,786	4,523	58,108	89,059	8,39	89,697	9,272	69,227	89,999	16,352
	AB	80,873	3,92	88,108	80,809	7,506	79,673	5,961	87,826	79,552	11,163
	RF	81,547	3,882	83,784	81,527	7,419	82,83	6,45	80	82,872	11,937
	Linear SVM	93,137	6,99	55,135	93,474	12,402	97,583	27,931	41,884	98,404	33,506
	Non-linear SVM	86,594	4,283	66,486	86,773	8,046	93,465	13,436	64,203	93,896	22,221
720 days											
Seccia	**KNN**	75,1	2,4	64,9	75,2	4,6	85,2	7,6	75,8	85,4	13,5
	AB	86,9	4,9	70,3	87,1	9,2	85,0	8,3	84,5	85,1	15,1
	RF	86,2	5,2	78,4	86,3	9,8	86,2	8,9	84,1	86,2	16,1
	Non-linear SVM	84,8	4,8	81,1	84,8	9,1	87,8	9,3	77,3	87,9	16,6
This study	**KNN**	91,545	5,814	52,162	91,92	10,458	90,719	10,838	67,874	91,084	18,69
	AB	81,991	3,949	77,568	82,033	7,516	80,253	6,451	85,7	80,166	11,999
	RF	82,432	3,697	70,27	82,548	7,024	85,011	7,678	77,44	85,132	13,97
	Linear SVM	93,979	8,769	57,027	94,331	15,195	97,754	33,333	42,85	98,63	37,486
	Non-linear SVM	87	4,638	65,135	87,208	8,657	94,048	15,704	63,816	94,531	25,204

Table 6.2: Comparison between the best performance values of this study and the study by Seccia et al. [1] for the classification problem in the HO approach.

Days	Feature-keeping (FK)					Record-keeping (RK)				
	Accuracy	Precision	Recall	Specificity	F1-score	Accuracy	Precision	Recall	Specificity	F1-score
180 days										
Seccia	96,1	10,5	44,4	96,6	16,9	98,0	30,8	38,5	98,8	34,2
This study	95,397	56,402	35,705	98,407	41,596	95,967	84,869	62,414	98,982	71,574
360 days										
Seccia	97,0	14,8	40,0	97,6	21,6	97,5	29,5	50,0	98,2	37,1
This study	95,458	70,103	40,007	98,844	49,330	96,383	86,242	70,678	98,888	77,332
720 days										
Seccia	97,1	20,7	60,0	97,5	30,8	98,0	42,7	67,3	98,5	52,3
This study	93,510	48,021	43,539	96,805	45,040	95,862	86,492	70,011	98,748	77,105

To conclude this section, it is important to note that, unlike Seccia, the hyperparameters of the classifiers and the different NN architectures were adjusted to each dataset and time window to extract the maximum capacity of the models with the different data used.

6.3 Explainability methods

The diversity of explainability methods applied made it possible to generate several explanations that provide information about the relationship between the data, the model and the outcome. These explanations evaluate different characteristics of the models and allow the analysis of the consistencies between results and comparison of the approaches developed. In addition to increasing confidence in predictive models, in the future, the explanations will make it possible to identify the safest and most reliable models for clinical application.

6.3.1 Methods and produced explanations

Initially, for each approach (VO and HO), the iteration model with the best performance of each problem was identified. This choice was made since several classifiers were studied, and it is impossible to analyse all the models' explanations. Thus, the best model allows the analysis of the predictions with greater confidence. Nevertheless, the conclusions obtained for each model cannot be generalised because the results change from model to model.

Next, global and local explainability methods that complement each other were applied. These methods are essential for the future validation of ML models in the clinical domain.

Global explanations

The global explanations can provide information about the global dynamics of the models and even about the MS disease. The feature recurrence analysis allowed the study of the best set of features for the problems under investigation. The features F6, F11 and F12 were always selected in the different iterations of the various problems. This high selection as input highlights their significant role in disease classification.

Determining the Permutation Feature Importance (PFI) of the selected features provided a clear view of the overall behaviour of the model, considering the interactions between the features. The inclusion of these interactions is crucial because it expands the knowledge about the relationships between model variables and the target. For example, the combination of features can significantly impact the prediction process, which is not seen using the features individually. The representation of the mean and standard deviation of the PFI measure allows the easy comparison of the importance of each feature in the different problems. However, given the randomness demonstrated, it might be interesting to explore other metrics and approaches to analyse the importance results of the various iterations.

The last global methods applied were the Individual Conditional Expectation (ICE) and Partial Dependence Plot (PDP) plots. Due to a large amount of information to interpret, only the plots of the most important features of the best model were included in this thesis. These plots provide a simple way to visualise the influence of different features on the prediction outcome and determine whether their behaviour follows the expected clinical knowledge. In this case, only feature F6 is known to be the Expanded disability status scale (EDSS) value. It is observed that it follows theoretical knowledge since the higher its value, the higher the probability of a prediction of transition to the SP course.

Combining the different global methods results provided more insight and a better understanding of the reason for the prediction. In summary, the recurrence analysis gave general information about the features that stand out the most in iterations, the PFI allowed the study of interactions between features, and the PDPs represented the dependency between the target and a particular feature of interest.

Local explanations

The Local Interpretable Model-Agnostic Explanations (LIME) method was used to analyse sample-specific explanations and understand how models use features in decisions. Two procedures were applied to select the instances: the Submodular Pick-LIME (SP-LIME) method and the application of the LIME method on selected samples that were correctly and wrongly classified. It is important to highlight that the range of values shown in the LIME plots corresponds to the normalised values. Moreover, all selected samples and their explanations are different, so it is not possible to make a direct comparison.

The application of the SP-LIME method resulted in the representation, and consequent analysis, of a set of explanations that simulate the global understanding of the model.

In the other approach, the selection of the samples for local explanations was based on a clustering technique to ensure that the set of selected examples was distinct and representative of the data. In addition, separating the samples into the three groups of correctly classified (class 1 or class 0) and misclassified and the consequent analysis of the explanations of each type was an important step towards a more comprehensive understanding of model behaviour. To apply these methods in clinical settings, it is important to understand what led the model to classify correctly and mainly the reason for a misclassification.

Through the selected instances in both approaches, it was understood how the models think, the local impact of each feature on the final decision when the others are in a particular range, and, in the case of the HO approach, which is the most appropriate time step. Moreover, the prediction probability for each sample given by the LIME method is also valuable because it helps to understand the model's confidence. These factors are essential for the clinician to understand the model's behaviour, compare it with existing medical knowledge, and validate the explanations and models.

Overall analysis

The results of all problems indicate that the most influential features are F6, F11, and F12; F6 is the EDSS value and the meaning of the last two is unknown. Both the LIME values of the represented samples and the PDPs showed that, as expected, low values of EDSS promote continuity on the RR course, while high values promote development towards the SP course.

Nevertheless, in most explanations, the EDSS value is not the one that has a greater weight in the classification. This fact highlights the complexity of MS and the challenge of its prognosis due to the heterogeneity of features.

6.3.2 Results constraints

The lack of knowledge about the dataset features strongly affected the explainability step. Despite the efforts to discover the meaning of the features, the information found was insufficient for a solid analysis of the model behaviour, the impact of the different features and the consequent comparison with the theoretical knowledge. Moreover, using a dataset is not enough to select the best model for each problem and ensure its reliability in the complex context of predicting the evolution of MS.

Regarding the methods, several techniques were applied and resulted in a large number of explanations. Only the most relevant results for the best model were included in this thesis. This decision was made since the abundance of information may confuse the information receiver. This problem can be compounded by the different results between methods that result from their different logical frameworks in determining the impact of features on prediction. For example, in the RK 720 problem of the VO approach, although feature F10 has a high recurrence, it has relatively low permutation importance and almost no impact on local explanations. Still, this comparison between methods is essential to understand which are the most suitable to support disease prediction models and which complement each other.

The implementation of other methods could lead to improvements in explanations. For example, Sousa et al. [125] explored counterfactual explanations in a MS progression problem. Although the number of features that could be directly changed was small, Sousa showed that this method is promising. In the present study, it was impossible to explore this method due to the lack of knowledge of the meaning of different features. Furthermore, applying the famous SHapley Additive exPlanations (SHAP) explainer was impossible because it does not support the Bagging model.

At a local level, it would be interesting to analyse instances incorrectly predicted by clinicians and inconclusive instances for clinicians that the models correctly classify since these are the cases in which the existence of a ML model to support decision-making is most important. These approaches would require clinicians' involvement in sample identification but would be a key step towards validation.

Finally, it is noticeable that this study lacks the rigorous evaluation of the explanations by data scientists, as done by Sousa et al. [125], but especially by MS experts. Although quite complex, this evaluation by MS experts is fundamental for the models to be validated and applied safely and reliably in medical practice. Thus, it is expected that the information presented to physicians will support decision-making and make diagnoses more accurate after future adaptations and refinements of this model. Additionally, these explanations could be an additional tool that physicians can use to explain to patients the challenges involved in MS prognosis.

7
Conclusion

This project aimed to predict whether a patient will pass from the Relapse Remitting (RR) course to the Secondary Progressive (SP) course in a given time window and to explore different explanations for the Machine Learning (ML) models. It also aims to provide explanations to better understand the models' decision-making and disease dynamics. This was done using the six datasets processed by Seccia et al. [1] from the Multiple Sclerosis (MS) service of Sant'Andrea Hospital.

Initially, two different prediction approaches (Visited-Oriented (VO) and History-Oriented (HO)) were developed based on the study by Seccia et al. [1] that led to the construction and evaluation of seven alternative ML models for each dataset. These algorithms were used to classify each patient as either not transitioning or transitioning to the SP course. It was then possible to compare the performances of the models. Overall, all the proposed models outperformed the results obtained by Seccia et al. [1] for the F1-score metric. The best results for the VO approach were obtained using the Record-keeping (RK) 720 dataset and the linear Support Vector Machines (SVM) classifier. They achieved an accuracy of 97.75%, a precision of 33.33%, a sensitivity of 42.85%, a specificity of 98.63% and a F1-score of 37.49%. In particular, this work's best performing classification problem was obtained by the HO approach with the RK 360 dataset, with an accuracy of 96.38%, a precision of 86.24%, and a sensitivity of 70.68%, a specificity of 98.89% and a F1-score of 77.33%. In performance evaluation, it is fundamental to consider different metrics to evaluate the classifier's performance in detail, with the F1-score metric assuming greater relevance for the MS problem under study.

It is concluded that the best classification approach is the HO scenario, with quite satisfactory F1-score values. Long Short-Term Memory (LSTM) Neural Networks (NNs) are very promising for short- and long-term forecasts, but using datasets with larger visits is essential. Therefore, testing the methodology presented in different datasets is encouraged to validate the conclusions drawn. In addition, an effort by physicians to register clinical data of MS patients is called for to increase the number of quality datasets available.

Besides predicting progression for the SP state, the logic of the models in decision-making through explainability methods was also studied. The combination of different global and local methods proved to be very useful for increasing the comprehension of the models. The Expanded disability status scale (EDSS) feature was often relevant in the different explanations. This observation gives a certain degree of confidence to the results as it is consistent with medical knowledge. It is worth noting that a significant limitation of this study was the fact that the name of the features was hidden, and it was not possible to identify more than half of them. Consequently, the methodology of explainability was immediately compromised, and achieving the desired depth of analysis was impossible.

In the future, it is essential to apply different complete datasets, conduct an evaluation by experts, refine the model and explore other explainability techniques. These steps are essential to ensure that explanations are secure and reliable and help MS experts understand the models and discover hidden patterns in the data.

Bibliography

[1] R. Seccia, D. Gammelli, F. Dominici, S. Romano, A. C. Landi, M. Salvetti, A. Tacchella, A. Zaccaria, A. Crisanti, F. Grassi, *et al.*, "Considering patient clinical history impacts performance of machine learning models in predicting course of multiple sclerosis," *PloS one*, vol. 15, no. 3, p. e0230219, 2020.

[2] *Atlas of MS.* The Multiple Sclerosis International Federation (MSIF), 3 ed., 2020.

[3] J. F. Kurtzke, "Rating neurologic impairment in multiple sclerosis: an expanded disability status scale (edss)," *Neurology*, vol. 33, no. 11, pp. 1444–1452, 1983.

[4] C. Confavreux and S. Vukusic, "The clinical course of multiple sclerosis," *Handbook of clinical neurology*, vol. 122, pp. 343–369, 2014.

[5] S. Klineova and F. D. Lublin, "Clinical course of multiple sclerosis," *Cold Spring Harbor perspectives in medicine*, vol. 8, no. 9, 2018.

[6] B. Mahesh, "Machine learning algorithms - a review," *International Journal of Science and Research (IJSR)*, vol. 9, pp. 381–386, 2020.

[7] M. Kuhn, K. Johnson, *et al.*, *Applied predictive modeling*, vol. 26. Springer, 2013.

[8] Z.-H. Zhou, *Machine Learning.* Springer, 2021.

[9] I. Muhammad and Z. Yan, "Supervised machine learning approaches: a survey.," *ICTACT Journal on Soft Computing*, vol. 5, no. 3, 2015.

[10] A. Ben-Hur and J. Weston, "A user's guide to support vector machines," in *Data mining techniques for the life sciences*, pp. 223–239, Springer, 2010.

[11] S. Dobilas, "Lstm recurrent neural networks-how to teach a network to remember the past," Mar 2022.

[12] "Model explainability with aws artificial intelligence and machine learning solutions," 2021.

[13] F. Doshi-Velez and B. Kim, "Towards a rigorous science of interpretable machine learning," *arXiv preprint arXiv:1702.08608*, 2017.

[14] C. Molnar, *Interpretable machine learning.* Lulu. com, 2020.

[15] A. Thompson, B. Banwell, F. Barkhof, W. Carroll, T. Coetzee, G. Comi, J. Correale, F. Fazekas, M. Filippi, M. Freedman, S. Galetta, H. P. Hartung, L. Kappos, F. Lublin, R. A. Marrie, A. Miller, D. Miller, X. Montalban, E. Mowry, S. Sorensen, M. Tintoré, A. Traboulsee, M. Trojano, B. Uitdehaag, S. Vukusic, E. Waubant, B. Weinshenker, S. Reingold, and J. Cohen, "Diagnosis of multiple sclerosis: 2017 revisions of the mcdonald criteria," *The Lancet Neurology*, vol. 17, p. 162–173, 2017.

[16] F. D. Lublin, "New multiple sclerosis phenotypic classification," *European Neurology*, vol. 72, no. 1, pp. 1–5, 2014.

[17] S. L. Hauser and B. A. C. Cree, "Treatment of multiple sclerosis: A review.," *The American journal of medicine*, vol. 133, no. 12, p. 1380–1390, 2020.

[18] R. Seccia, S. Romano, M. Salvetti, A. Crisanti, L. Palagi, and F. Grassi, "Machine learning use for prognostic purposes in multiple sclerosis," *Life*, vol. 11, no. 2, p. 122, 2021.

[19] T. Chitnis, "Role of puberty in multiple sclerosis risk and course," *Clinical Immunology*, vol. 149, no. 2, pp. 192–200, 2013.

[20] V. Saccà, A. Sarica, F. Novellino, S. Barone, T. Tallarico, E. Filippelli, A. Granata, C. Chiriaco, R. B. Bossio, P. Valentino, *et al.*, "Evaluation of machine learning algorithms performance for the prediction of early multiple sclerosis from resting-state fmri connectivity data," *Brain imaging and behavior*, vol. 13, no. 4, pp. 1103–1114, 2019.

[21] M. McGinley, C. Goldschmidt, and A. Rae-Grant, "Diagnosis and treatment of multiple sclerosis: A review," *JAMA*, vol. 325, no. 8, p. 765–779, 2021.

[22] T. Grzegorski and J. Losy, "Multiple sclerosis–the remarkable story of a baffling disease," *Reviews in the Neurosciences*, vol. 30, no. 5, pp. 511–526, 2019.

[23] G. Giovannoni, H. Butzkueven, S. Dhib-Jalbut, J. Hobart, G. Kobelt, G. Pepper, M. P. Sormani, C. Thalheim, A. Traboulsee, and T. Vollmer, "Brain

health: time matters in multiple sclerosis," *Multiple sclerosis and related disorders*, vol. 9, pp. S5–S48, 2016.

[24] R. Maguire, B. McKeague, N. Kóka, L. Coffey, P. Maguire, and D. Desmond, "The role of expectations and future-oriented cognitions in quality of life of people with multiple sclerosis: A systematic review," *Multiple Sclerosis and Related Disorders*, vol. 56, p. 103293, 2021.

[25] A. Ochoa-Morales, T. Hernández-Mojica, F. Paz-Rodríguez, A. Jara-Prado, Z. T.-D. L. Santos, M. Sánchez-Guzmán, J. Guerrero-Camacho, T. Corona-Vázquez, J. Flores, A. Camacho-Molina, V. Rivas-Alonso, and D. J. D.-O. de Montellano, "Quality of life in patients with multiple sclerosis and its association with depressive symptoms and physical disability," *Multiple sclerosis and related disorders*, vol. 36, 2019.

[26] R. Bergamaschi, "Can we predict the evolution of an unpredictable disease like multiple sclerosis?," *Eur. J. Neurol*, vol. 20, no. 7, pp. 995–996, 2013.

[27] R. Bergamaschi, S. Quaglini, M. Trojano, M. P. Amato, E. Tavazzi, D. Paolicelli, V. Zipoli, A. Romani, A. Fuiani, E. Portaccio, *et al.*, "Early prediction of the long term evolution of multiple sclerosis: the bayesian risk estimate for multiple sclerosis (brems) score," *Journal of Neurology, Neurosurgery & Psychiatry*, vol. 78, no. 7, pp. 757–759, 2007.

[28] F. D. Lublin, S. C. Reingold, J. A. Cohen, G. R. Cutter, P. S. Sørensen, A. J. Thompson, J. S. Wolinsky, L. J. Balcer, B. Banwell, F. Barkhof, B. Bebo Jr, P. A. Calabresi, M. Clanet, G. Comi, R. J. Fox, M. S. Freedman, A. D. Goodman, M. Inglese, L. Kappos, B. C. Kieseier, J. A. Lincoln, C. Lubetzki, A. E. Miller, X. Montalban, P. W. O'Connor, J. Petkau, C. Pozzilli, R. A. Rudick, M. P. Sormani, O. Stüve, E. Waubant, and C. H. Polman, "Defining the clinical course of multiple sclerosis: the 2013 revisions," *Neurology*, vol. 83, no. 3, p. 278–286, 2014.

[29] A. Manouchehrinia, F. Zhu, D. Piani-Meier, M. Lange, D. G. Silva, R. Carruthers, A. Glaser, E. Kingwell, H. Tremlett, and J. Hillert, "Predicting risk of secondary progression in multiple sclerosis: a nomogram," *Multiple Sclerosis Journal*, vol. 25, no. 8, pp. 1102–1112, 2019.

[30] J. Oh, A. Vidal-Jordana, and X. Montalban, "Multiple sclerosis: clinical aspects," *Current opinion in neurology*, vol. 31, no. 6, pp. 752–759, 2018.

[31] M. M. Goldenberg, “Multiple sclerosis review,” *PT: a peer-reviewed journal for formulary management*, vol. 37, no. 3, p. 175–184, 2012.

[32] J. D. Haines, M. Inglese, and P. Casaccia, “Axonal damage in multiple sclerosis,” *The Mount Sinai journal of medicine*, vol. 78, no. 2, pp. 231–243, 2011.

[33] T. Olsson, L. F. Barcellos, and L. Alfredsson, “Interactions between genetic, lifestyle and environmental risk factors for multiple sclerosis,” *Nature reviews: Neurology*, vol. 13, no. 1, pp. 25–36, 2017.

[34] B. Nourbakhsh and E. M. Mowry, “Multiple sclerosis risk factors and pathogenesis,” *Multiple Sclerosis and other CNS Inflammatory Diseases*, vol. 25, no. 3, pp. 596–610, 2019.

[35] E. Koutsouraki, V. Costa, and S. Baloyannis, “Epidemiology of multiple sclerosis in europe: a review,” *International review of psychiatry*, vol. 22, no. 1, pp. 2–13, 2010.

[36] M. B. Sintzel, M. Rametta, , and A. T. Reder, “Vitamin d and multiple sclerosis: A comprehensive review,” *Neurology and therapy*, vol. 7, no. 1, pp. 59–85, 2018.

[37] L. Alfredsson and T. Olsson, “Lifestyle and environmental factors in multiple sclerosis,” *Cold Spring Harbor perspectives in medicine*, vol. 9, no. 4, 2019.

[38] H.-P. Hartung, J. Graf, O. Aktas, J. Mares, and M. H. Barnett, “Diagnosis of multiple sclerosis: revisions of the mcdonald criteria 2017 - continuity and change.,” *Current opinion in neurology*, vol. 32, no. 3, pp. 327–337, 2019.

[39] M. Gaspari, G. Roveda, C. Scandellari, and S. Stecchi, “An expert system for the evaluation of edss in multiple sclerosis,” *Artificial intelligence in medicine*, vol. 25, no. 2, pp. 187–210, 2002.

[40] B. Sharrack, R. Hughes, S. Soudain, and G. Dunn, “The psychometric properties of clinical rating scales used in multiple sclerosis,” *Brain: a journal of neurology*, vol. 122, no. 1, p. 141–159, 1999.

[41] J. Kragt, A. Thompson, X. Montalban, M. Tintore, J. Rio, C. Polman, and B. Uitdehaag, “Responsiveness and predictive value of edss and msfc in primary progressive ms,” *Neurology*, vol. 70, no. 13 Part 2, pp. 1084–1091, 2008.

[42] F. D. Lublin and S. C. Reingold, “Defining the clinical course of multiple sclerosis: results of an international survey.,” *Neurology*, vol. 46, no. 4, p. 907–911, 1996.

[43] B. Lo Sasso, L. Agnello, G. Bivona, C. Bellia, and M. Ciaccio, “Cerebrospinal fluid analysis in multiple sclerosis diagnosis: An update,” *Medicina*, vol. 55, no. 6, p. 245, 2019.

[44] F. D. Lublin, T. Coetzee, J. A. Cohen, R. A. Marrie, and A. J. Thompson, “The 2013 clinical course descriptors for multiple sclerosis: A clarification.,” *Neurology*, vol. 94, no. 24, p. 1088–1092, 2020.

[45] S. Vukusic and C. Confavreux, “Primary and secondary progressive multiple sclerosis,” *Journal of the neurological sciences*, vol. 206, no. 2, pp. 153–155, 2003.

[46] K. S. Sullivan and G. McDonnell, “Benign multiple sclerosis? clinical course, long term follow up, and assessment of prognostic factors.,” *Journal of neurology, neurosurgery, and psychiatry*, vol. 67, no. 2, pp. 148–152, 1999.

[47] G. S. M. Ramsaransing and J. D. Keyser, “Benign course in multiple sclerosis: a review.,” *Acta neurologica Scandinavica*, vol. 113, no. 6, p. 359–369, 2006.

[48] T. J. Murray, “Diagnosis and treatment of multiple sclerosis.,” *BMJ*, vol. 332, no. 7540, p. 525–527, 2006.

[49] C. H. Polman and B. M. J. Uitdehaag, “Drug treatment of multiple sclerosis.,” *The Western journal of medicine*, vol. 173, no. 6, p. 398–402, 2000.

[50] A. Gajofatto and M. D. Benedetti, “Treatment strategies for multiple sclerosis: when to start, when to change, when to stop?,” *World Journal of Clinical Cases: WJCC*, vol. 3, no. 7, p. 545, 2015.

[51] M. Tintore, A. Vidal-Jordana, and J. Sastre-Garriga, “Treatment of multiple sclerosis - success from bench to bedside.,” *Nature reviews*, vol. 15, no. 1, p. 53–58, 2019.

[52] I. El Naqa and M. J. Murphy, “What is machine learning?,” in *machine learning in radiation oncology*, pp. 3–11, Springer, 2015.

[53] J. Brownlee, *Data Preparation for Machine Learning: Data Cleaning, Feature Selection, and Data Transforms in Python.* Machine Learning Mastery, 2020.

[54] C. Reid Turner, A. Fuggetta, L. Lavazza, and A. L. Wolf, “A conceptual basis for feature engineering,” *Journal of Systems and Software*, vol. 49, no. 1, pp. 3–15, 1999.

[55] K. Y. Ngiam and I. W. Khor, “Big data and machine learning algorithms

for health-care delivery," *The Lancet Oncology*, vol. 20, no. 5, pp. e262–e273, 2019.

[56] I. H. Witten, E. Frank, and M. A. Hall, *Data Mining: Practical Machine Learning Tools and Techniques.* Morgan Kaufmann, 3 ed., 2011.

[57] A. R. Donders, van der Heijden, G. J., T. Stijnen, and K. G. Moons, "Review: a gentle introduction to imputation of missing values," *Journal of clinical epidemiology*, vol. 59, no. 10, p. 1087–1091, 2006.

[58] P. D. Allison, "Multiple imputation for missing data: A cautionary tale," *Sociological Methods Research*, vol. 28, no. 3, p. 301–309, 2000.

[59] H. S. Obaid, S. A. Dheyab, and S. S. Sabry, "The impact of data pre-processing techniques and dimensionality reduction on the accuracy of machine learning," in *2019 9th Annual Information Technology, Electromechanical Engineering and Microelectronics Conference (IEMECON)*, pp. 279–283, 2019.

[60] G. Chandrashekar and F. Sahin, "A survey on feature selection methods," *Computers Electrical Engineering*, vol. 40, no. 1, pp. 16–28, 2014.

[61] S. Khalid, T. Khalil, and S. Nasreen, "A survey of feature selection and feature extraction techniques in machine learning," in *2014 science and information conference*, pp. 372–378, IEEE, 2014.

[62] S. Velliangiri, S. Alagumuthukrishnan, *et al.*, "A review of dimensionality reduction techniques for efficient computation," *Procedia Computer Science*, vol. 165, pp. 104–111, 2019.

[63] J. M. De Sa, *Pattern recognition: concepts, methods, and applications.* Springer Science & Business Media, 2001.

[64] J. Tang, S. Alelyani, and H. Liu, *Feature selection for classification: A review*, pp. 37–64. CRC Press, 2014.

[65] H. He and Y. Ma, *Imbalanced learning: foundations, algorithms, and applications.* Wiley-Blackwell, 2013.

[66] A. Fernández, S. García, M. Galar, R. C. Prati, B. Krawczyk, and F. Herrera, *Learning from imbalanced data sets*, vol. 10. Springer, 2018.

[67] J. Brownlee, *Machine learning mastery with Python: understand your data, create accurate models, and work projects end-to-end.* Machine Learning Mastery, 2016.

[68] J.-H. Kim, “Estimating classification error rate: Repeated cross-validation, repeated hold-out and bootstrap,” *Computational Statistics Data Analysis*, vol. 53, no. 11, pp. 3735–3745, 2009.

[69] D. G. Kleinbaum and M. Klein, “Introduction to logistic regression,” in *Logistic regression*, pp. 1–39, Springer, 2010.

[70] A. Chang, “Intelligence based medicine,” *Artificial Intelligence and Human Cognition in Clinical Medicine and Healthcare*, pp. 397–412, 2020.

[71] H. O. Alanazi, A. H. Abdullah, and K. N. Qureshi, “A critical review for developing accurate and dynamic predictive models using machine learning methods in medicine and health care,” *Journal of medical systems*, vol. 41, no. 4, p. 69, 2017.

[72] K. Taunk, S. De, S. Verma, and A. Swetapadma, “A brief review of nearest neighbor algorithm for learning and classification,” in *2019 International Conference on Intelligent Computing and Control Systems (ICCS)*, pp. 1255–1260, IEEE, 2019.

[73] D. Berrar, “Bayes’ theorem and naive bayes classifier,” *Encyclopedia of Bioinformatics and Computational Biology: ABC of Bioinformatics; Elsevier Science Publisher: Amsterdam, The Netherlands*, pp. 403–412, 2018.

[74] V. Nasteski, “An overview of the supervised machine learning methods,” *Horizons*, vol. 4, pp. 51–62, 2017.

[75] S. Yildirim, “Hyperparameter tuning for support vector machines: C and gamma parameters,” Jun 2020.

[76] L. J. Lancashire, C. Lemetre, and G. R. Ball, “An introduction to artificial neural networks in bioinformatics,” *Briefings in bioinformatics*, vol. 10, no. 3, pp. 315–329, 2009.

[77] A. Moldovan, A. Caţaron, and R. Andonie, “Learning in feedforward neural networks accelerated by transfer entropy,” *Entropy*, vol. 22, no. 1, p. 102, 2020.

[78] P. T. Yamak, L. Yujian, and P. K. Gadosey, “A comparison between arima, lstm, and gru for time series forecasting,” in *Proceedings of the 2019 2nd International Conference on Algorithms, Computing and Artificial Intelligence*, pp. 49–55, 2019.

[79] H. Sak, A. W. Senior, and F. Beaufays, “Long short-term memory recurrent neural network architectures for large scale acoustic modeling,” 2014.

[80] M. Feurer and F. Hutter, "Hyperparameter optimization," in *Automated machine learning*, pp. 3–33, Springer, Cham, 2019.

[81] R. P. Espíndola and N. F. Ebecken, "On extending f-measure and g-mean metrics to multi-class problems," *WIT Transactions on Information and Communication Technologies*, vol. 35, 2005.

[82] M. Bekkar, H. K. Djemaa, and T. A. Alitouche, "Evaluation measures for models assessment over imbalanced data sets," *J Inf Eng Appl*, vol. 3, no. 10, 2013.

[83] M. Hossin and M. N. Sulaiman, "A review on evaluation metrics for data classification evaluations," *International journal of data mining & knowledge management process*, vol. 5, no. 2, p. 1, 2015.

[84] M. Galar, A. Fernandez, E. Barrenechea, H. Bustince, and F. Herrera, "A review on ensembles for the class imbalance problem: bagging-, boosting-, and hybrid-based approaches," *IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews)*, vol. 42, no. 4, pp. 463–484, 2011.

[85] A. Kumarl, "Accuracy, precision, recall amp; f1-score - python examples," Jan 2022.

[86] S. Tonekaboni, S. Joshi, M. D. McCradden, and A. Goldenberg, "What clinicians want: contextualizing explainable machine learning for clinical end use," in *Machine learning for healthcare conference*, pp. 359–380, PMLR, 2019.

[87] B. Goodman and S. Flaxman, "European union regulations on algorithmic decision-making and a "right to explanation"," *AI magazine*, vol. 38, no. 3, pp. 50–57, 2017.

[88] M. A. Ahmad, C. Eckert, and A. Teredesai, "Interpretable machine learning in healthcare," in *Proceedings of the 2018 ACM international conference on bioinformatics, computational biology, and health informatics*, pp. 559–560, 2018.

[89] L. H. Gilpin, D. Bau, B. Z. Yuan, A. Bajwa, M. Specter, and L. Kagal, "Explaining explanations: An overview of interpretability of machine learning," in *2018 IEEE 5th International Conference on data science and advanced analytics (DSAA)*, pp. 80–89, IEEE, 2018.

[90] Q. Zhao and T. Hastie, "Causal interpretations of black-box models," *Journal of Business & Economic Statistics*, vol. 39, no. 1, pp. 272–281, 2021.

[91] R. Elshawi, M. H. Al-Mallah, and S. Sakr, "On the interpretability of machine learning-based model for predicting hypertension," *BMC medical informatics and decision making*, vol. 19, no. 1, pp. 1–32, 2019.

[92] G. Cid, "Understanding lime in 5 steps," Mar 2022.

[93] S. Wachter, B. Mittelstadt, and C. Russell, "Counterfactual explanations without opening the black box: Automated decisions and the gdpr," *Harv. JL & Tech.*, vol. 31, p. 841, 2017.

[94] K. S. Gurumoorthy, A. Dhurandhar, G. Cecchi, and C. Aggarwal, "Efficient data representation by selecting prototypes with importance weights," in *2019 IEEE International Conference on Data Mining (ICDM)*, pp. 260–269, IEEE, 2019.

[95] P. W. Koh and P. Liang, "Understanding black-box predictions via influence functions," in *International conference on machine learning*, pp. 1885–1894, PMLR, 2017.

[96] D. Bau, B. Zhou, A. Khosla, A. Oliva, and A. Torralba, "Network dissection: Quantifying interpretability of deep visual representations," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 6541–6549, 2017.

[97] I. Stafford, M. Kellermann, E. Mossotto, R. Beattie, B. MacArthur, and S. Ennis, "A systematic review of the applications of artificial intelligence and machine learning in autoimmune diseases," *NPJ digital medicine*, vol. 3, no. 1, pp. 1–11, 2020.

[98] K. Sakai and K. Yamada, "Machine learning studies on major brain diseases: 5-year trends of 2014–2018," *Japanese journal of radiology*, vol. 37, no. 1, pp. 34–72, 2019.

[99] S. Toscano and F. Patti, "Csf biomarkers in multiple sclerosis: Beyond neuroinflammation," *Neuroimmunology and Neuroinflammation*, vol. 8, no. 1, pp. 14–41, 2021.

[100] A. Eshaghi, A. L. Young, P. A. Wijeratne, F. Prados, D. L. Arnold, S. Narayanan, C. R. Guttmann, F. Barkhof, D. C. Alexander, A. J. Thomp-

son, *et al.*, “Identifying multiple sclerosis subtypes using unsupervised machine learning and mri data,” *Nature communications*, vol. 12, no. 1, pp. 1–12, 2021.

[101] M. Inglese and M. Petracca, “Mri in multiple sclerosis: clinical and research update,” *Current opinion in neurology*, vol. 31, no. 3, pp. 249–255, 2018.

[102] Y. Yoo, L. Y. Tang, D. K. Li, L. Metz, S. Kolind, A. L. Traboulsee, and R. C. Tam, “Deep learning of brain lesion patterns and user-defined clinical and mri features for predicting conversion to multiple sclerosis from clinically isolated syndrome,” *Computer Methods in Biomechanics and Biomedical Engineering: Imaging & Visualization*, vol. 7, no. 3, pp. 250–259, 2019.

[103] K. C. Jackson, K. Sun, C. Barbour, D. Hernandez, P. Kosa, M. Tanigawa, A. M. Weideman, and B. Bielekova, “Genetic model of ms severity predicts future accumulation of disability,” *Annals of human genetics*, vol. 84, no. 1, pp. 1–10, 2020.

[104] P. M. Matthews, V. J. Block, and L. Leocani, “E-health and multiple sclerosis,” *Current opinion in neurology*, vol. 33, no. 3, pp. 271–276, 2020.

[105] M. F. Pinto, H. Oliveira, S. Batista, L. Cruz, M. Pinto, I. Correia, P. Martins, and C. Teixeira, “Prediction of disease progression and outcomes in multiple sclerosis with machine learning,” *Scientific reports*, vol. 10, no. 1, pp. 1–13, 2020.

[106] Y. Zhao, T. Wang, R. Bove, B. Cree, R. Henry, H. Lokhande, M. Polgar-Turcsanyi, M. Anderson, R. Bakshi, H. L. Weiner, *et al.*, “Ensemble learning predicts multiple sclerosis disease course in the summit study,” *NPJ digital medicine*, vol. 3, no. 1, pp. 1–8, 2020.

[107] G. Brichetto, M. M. Bragadin, S. Fiorini, M. A. Battaglia, G. Konrad, M. Ponzio, L. Pedulla, A. Verri, A. Barla, and A. Tacchino, “The hidden information in patient-reported outcomes and clinician-assessed outcomes: multiple sclerosis as a proof of concept of a machine learning approach,” *Neurological sciences*, vol. 41, no. 2, pp. 459–462, 2020.

[108] M. T. Law, A. L. Traboulsee, D. K. Li, R. L. Carruthers, M. S. Freedman, S. H. Kolind, and R. Tam, “Machine learning in secondary progressive multiple sclerosis: an improved predictive model for short-term disability progression,” *Multiple Sclerosis Journal–Experimental, Translational and Clinical*, vol. 5, no. 4, p. 2055217319885983, 2019.

[109] Y. Zhao, B. C. Healy, D. Rotstein, C. R. Guttmann, R. Bakshi, H. L. Weiner,

C. E. Brodley, and T. Chitnis, “Exploration of machine learning techniques in predicting multiple sclerosis disease course,” *PloS one*, vol. 12, no. 4, p. e0174866, 2017.

[110] V. Wottschel, D. Alexander, P. Kwok, D. Chard, M. Stromillo, N. De Stefano, A. Thompson, D. Miller, and O. Ciccarelli, “Predicting outcome in clinically isolated syndrome using machine learning,” *NeuroImage: Clinical*, vol. 7, pp. 281–287, 2015.

[111] B. Bejarano, M. Bianco, D. Gonzalez-Moron, J. Sepulcre, J. Goñi, J. Arcocha, O. Soto, U. Del Carro, G. Comi, L. Leocani, *et al.*, “Computational classifiers for predicting the short-term course of multiple sclerosis,” *BMC neurology*, vol. 11, no. 1, pp. 1–9, 2011.

[112] H. Zhang, E. Alberts, V. Pongratz, M. Mühlau, C. Zimmer, B. Wiestler, and P. Eichinger, “Predicting conversion from clinically isolated syndrome to multiple sclerosis–an imaging-based machine learning approach,” *NeuroImage: Clinical*, vol. 21, p. 101593, 2019.

[113] M. Trojano, M. Tintore, X. Montalban, J. Hillert, T. Kalincik, P. Iaffaldano, T. Spelman, M. P. Sormani, and H. Butzkueven, “Treatment decisions in multiple sclerosis—insights from real-world observational studies,” *Nature Reviews Neurology*, vol. 13, no. 2, pp. 105–118, 2017.

[114] H. H. Kitzler, H. Wahl, J. C. Eisele, M. Kuhn, H. Schmitz-Peiffer, S. Kern, B. K. Rutt, S. C. Deoni, T. Ziemssen, and J. Linn, “Multi-component relaxation in clinically isolated syndrome: Lesion myelination may predict multiple sclerosis conversion,” *NeuroImage: Clinical*, vol. 20, pp. 61–70, 2018.

[115] A. Ion-Mărgineanu, G. Kocevar, C. Stamile, D. M. Sima, F. Durand-Dubief, S. Van Huffel, and D. Sappey-Marinier, “Machine learning approach for classifying multiple sclerosis courses by combining clinical data with lesion loads and magnetic resonance metabolic features,” *Frontiers in neuroscience*, vol. 11, p. 398, 2017.

[116] H. R. Afzal, S. Luo, S. Ramadan, and J. Lechner-Scott, “The emerging role of artificial intelligence in multiple sclerosis imaging,” *Multiple Sclerosis Journal*, p. 1352458520966298, 2020.

[117] E. E. Kontopodis, E. Papadaki, E. Trivzakis, T. G. Maris, P. Simos, G. Z. Papadakis, A. Tsatsakis, D. A. Spandidos, A. Karantanas, and K. Marias, “Emerging deep learning techniques using magnetic resonance imaging data

applied in multiple sclerosis and clinical isolated syndrome patients," *Experimental and Therapeutic Medicine*, vol. 22, no. 4, pp. 1–17, 2021.

[118] S. D. Auger, B. M. Jacobs, R. Dobson, C. R. Marshall, and A. J. Noyce, "Big data, machine learning and artificial intelligence: a neurologist's guide," *Practical Neurology*, vol. 21, no. 1, pp. 4–11, 2021.

[119] V. Wottschel, D. T. Chard, C. Enzinger, M. Filippi, J. L. Frederiksen, C. Gasperini, A. Giorgio, M. A. Rocca, A. Rovira, N. De Stefano, *et al.*, "Svm recursive feature elimination analyses of structural brain mri predicts near-term relapses in patients with clinically isolated syndromes suggestive of multiple sclerosis," *NeuroImage: Clinical*, vol. 24, p. 102011, 2019.

[120] P. Chakraborty, B. C. Kwon, S. Dey, A. Dhurandhar, D. Gruen, K. Ng, D. Sow, and K. R. Varshney, "Tutorial on human-centered explainability for healthcare," in *Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, pp. 3547–3548, 2020.

[121] L. Arbelaez Ossa, G. Starke, G. Lorenzini, J. E. Vogt, D. M. Shaw, and B. S. Elger, "Re-focusing explainability in medicine," *DIGITAL HEALTH*, vol. 8, p. 20552076221074488, 2022.

[122] F. Eitel, E. Soehler, J. Bellmann-Strobl, A. U. Brandt, K. Ruprecht, R. M. Giess, J. Kuchling, S. Asseyer, M. Weygandt, J.-D. Haynes, *et al.*, "Uncovering convolutional neural network decisions for diagnosing multiple sclerosis on conventional mri using layer-wise relevance propagation," *NeuroImage: Clinical*, vol. 24, p. 102003, 2019.

[123] A. Lopatina, S. Ropele, R. Sibgatulin, J. R. Reichenbach, and D. Güllmar, "Investigation of deep-learning-driven identification of multiple sclerosis patients based on susceptibility-weighted images using relevance analysis," *Frontiers in neuroscience*, vol. 14, p. 609468, 2020.

[124] J. C. Reinhold, A. Carass, and J. L. Prince, "A structural causal model for mr images of multiple sclerosis," in *International Conference on Medical Image Computing and Computer-Assisted Intervention*, pp. 782–792, Springer, 2021.

[125] M. J. C. d. Sousa, "On the explainability of multiple sclerosis disease progression models," Master's thesis, Universidade de Coimbra, 2021.

[126] V. Fonti and E. Belitser, "Feature selection using lasso," *VU Amsterdam research paper in business analytics*, vol. 30, pp. 1–25, 2017.

[127] A. Khwaja, M. Naeem, A. Anpalagan, A. Venetsanopoulos, and B. Venkatesh, "Improved short-term load forecasting using bagged neural networks," *Electric Power Systems Research*, vol. 125, pp. 109–115, 2015.

[128] Y. Yu, X. Si, C. Hu, and J. Zhang, "A review of recurrent neural networks: Lstm cells and network architectures," *Neural computation*, vol. 31, no. 7, pp. 1235–1270, 2019.

[129] M. H. Sazli, "A brief review of feed-forward neural networks," *Communications Faculty of Sciences University of Ankara Series A2-A3 Physical Sciences and Engineering*, vol. 50, no. 01, 2006.

[130] N. Srivastava, G. Hinton, A. Krizhevsky, I. Sutskever, and R. Salakhutdinov, "Dropout: a simple way to prevent neural networks from overfitting," *The journal of machine learning research*, vol. 15, no. 1, pp. 1929–1958, 2014.

[131] J. Brownlee, "How to choose an optimization algorithm," 2020. Last accessed 04 November 2021.

[132] J. Brownlee, "Loss and loss functions for training deep learning neural networks," 2019. Last accessed 04 November 2021.

[133] M. Dwarampudi and N. Reddy, "Effects of padding on lstms and cnns," *arXiv preprint arXiv:1903.07288*, 2019.

[134] "Time series forecasting using deep learning," 2022.

[135] G. Stiglic, P. Kocbek, N. Fijacko, M. Zitnik, K. Verbert, and L. Cilar, "Interpretability of machine learning-based prediction models in healthcare," *Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery*, vol. 10, no. 5, p. e1379, 2020.

[136] "Permutation feature importance," 2020.

[137] T. Botari, R. Izbicki, and A. C. de Carvalho, "Local interpretation methods to machine learning using the domain of the feature space," in *Joint European conference on machine learning and knowledge discovery in databases*, pp. 241–252, Springer, 2019.

[138] M. T. Ribeiro, S. Singh, and C. Guestrin, "" why should i trust you?" explaining the predictions of any classifier," in *Proceedings of the 22nd ACM SIGKDD international conference on knowledge discovery and data mining*, pp. 1135–1144, 2016.

[139] L. Rokach and O. Maimon, “Clustering methods,” in *Data mining and knowledge discovery handbook*, pp. 321–352, Springer, 2005.

[140] J. Dieber and S. Kirrane, “Why model why? assessing the strengths and limitations of lime,” *arXiv preprint arXiv:2012.00093*, 2020.

[141] H. D. Oliveira, “Evaluation and prediction of multiple sclerosis disease progression,” Master’s thesis, Universidade de Coimbra, 2020.

Appendices

A

Supplementary results

A.1 Hyperparameter optimisation in the VO scenario

A.1.1 Feature selection in the FK datasets

Table A.1: Results of the feature selection techniques for the dataset Feature-keeping (FK) 180 in the VO scenario.

Method	Hyperparameters	Accuracy	Precision	Recall	Specificity	F1-score
Without reduction	—	89,591	5,344	68,889	89,765	9,917
Pearson correlation	$n = 10$	90,095	5,664	69,444	90,268	10,472
	$n = 5$	90,887	5,904	66,667	91,09	10,846
	$n = 3$	91,771	6,487	66,111	91,986	11,814
LASSO regression	$\lambda = 0{,}001$	90,076	5,519	67,778	90,263	10,206
	$\lambda = 0{,}003$	91,252	6,104	66,111	91,463	11,173
	$\lambda = 0{,}005$	91,885	6,735	68,056	92,084	12,256

Table A.2: Results of the feature selection techniques for the dataset FK 360 in the VO scenario.

Method	Hyperparameters	Accuracy	Precision	Recall	Specificity	F1-score
Without reduction	—	90,074	5,341	61,351	90,329	9,827
Pearson correlation	$n = 10$	91,283	5,789	58,108	91,577	10,525
	$n = 5$	993,137	6,99	55,135	93,474	12,402
	$n = 3$	93,329	6,734	51,081	93,705	11,894
LASSO regression	$\lambda = 0{,}001$	91,461	5,899	57,838	91,76	10,701
	$\lambda = 0{,}003$	92,989	6,683	53,514	93,34	11,879
	$\lambda = 0{,}005$	93,256	6,961	53,784	93,606	12,323

Table A.3: Results of the feature selection techniques for the dataset FK 720 in the VO scenario.

Method	Hyperparameters	Accuracy	Precision	Recall	Specificity	F1-score
Without reduction	—	91,326	6,061	56,216	91,66	10,939
Pearson correlation	$n = 10$	93,293	7,703	55,405	93,654	13,519
	$n = 5$	93,979	8,769	57,027	94,331	15,195
	$n = 3$	94,252	8,107	49,189	94,681	13,91
LASSO regression	$\lambda = 0{,}001$	92,429	6,887	55,946	92,777	12,257
	$\lambda = 0{,}003$	94,007	8,239	52,703	94,4	14,244
	$\lambda = 0{,}005$	93,938	7,773	48,919	94,367	13,397

A.1.2 Feature selection in the RK datasets

Table A.4: Results of the feature selection techniques for the dataset RK 180 in the VO scenario.

Method	Hyperparameters	Accuracy	Precision	Recall	Specificity	F1-score
Without reduction	—	96,165	17,835	48,744	96,832	26,102
Pearson correlation	$n = 10$	96,404	18,656	47,295	97,095	26,752
	$n = 5$	96,716	19,531	44,589	97,463	26,964
	$n = 3$	96,681	19,047	42,802	97,439	26,358
LASSO regression	$\lambda = 0{,}001$	96,715	20,345	46,763	97,418	28,341
	$\lambda = 0{,}003$	96,955	21,150	43,671	97,705	28,487
	$\lambda = 0{,}005$	96,948	21,044	43,478	97,700	28,348

Table A.5: Results of the feature selection techniques for the dataset RK 360 in the VO scenario.

Method	Hyperparameters	Accuracy	Precision	Recall	Specificity	F1-score
Without reduction	—	97,005	23,153	45,507	97,765	30,675
Pearson correlation	$n = 10$	97,094	23,482	44,106	97,875	30,635
	$n = 5$	97,382	25,994	43,188	98,182	32,443
	$n = 3$	97,469	26,43	41,401	98,296	32,251
LASSO regression	$\lambda = 0{,}001$	97,297	25,135	43,382	98,092	31,824
	$\lambda = 0{,}003$	97,541	27,18	41,159	98,373	32,739
	$\lambda = 0{,}005$	97,583	27,931	41,884	98,404	33,506

Table A.6: Results of the feature selection techniques for the dataset RK 720 in the VO scenario.

Method	Hyperparameters	Accuracy	Precision	Recall	Specificity	F1-score
Without reduction	—	97,327	28,625	46,86	98,132	35,529
Pearson correlation	$n = 10$	97,478	30,208	46,039	98,299	36,468
	$n = 5$	97,738	33,121	43,092	98,61	37,447
	$n = 3$	97,76	33,258	42,174	98,647	37,177
LASSO regression	$\lambda = 0{,}001$	97,557	30,853	44,638	98,401	36,479
	$\lambda = 0{,}003$	97,704	32,322	42,126	98,591	36,571
	$\lambda = 0{,}005$	97,754	33,333	42,85	98,63	37,486

A.2 Hyperparameter optimisation in the HO scenario

A.2.1 Classification in the FK datasets

Table A.7: Results of the different NN architectures tested for the FK 180 dataset.

NN	Input strategy 1					Input strategy 2				
Model	**Accuracy**	**Precision**	**Recall**	**Specificity**	**F1-score**	**Accuracy**	**Precision**	**Recall**	**Specificity**	**F1-score**
Model 1	99,081	33,333	8,333	99,857	13,333	94,135	21,374	7,968	98,512	11,411
Model 2	98,935	22,628	12,727	99,674	15,507	94,093	31,768	15,878	98,197	20,954
Model 3	98,704	21,27	19,091	99,382	19,504	94,204	28,151	15,607	98,095	19,209
Model 4	97,465	10,589	20,909	98,12	12,497	94,243	32,371	17,57	98,143	22,019
Model 5	89,347	4,751	59,091	89,605	8,775	95,648	44,459	5,673	99,526	9,151
Model 6	71,767	2,282	75,455	71,734	4,423	95,247	41,734	12,833	99,082	18,148

Table A.8: Results of the different NN architectures tested for the FK 360 dataset.

NN	Input strategy 1					Input strategy 2				
Model	**Accuracy**	**Precision**	**Recall**	**Specificity**	**F1-score**	**Accuracy**	**Precision**	**Recall**	**Specificity**	**F1-score**
Model 1	98,960	20,011	9,091	99,731	12,511	94,163	20,635	9,081	98,308	12,376
Model 2	98,642	18,175	12,5	99,434	13,68	94,008	42,112	24,34	98,02	30,024
Model 3	97,481	13,534	22,5	98,201	15,705	93,604	36,323	25,766	97,444	29,757
Model 4	96,673	12,317	29,167	97,334	15,491	94,181	39,587	23,004	98,14	28,312
Model 5	87,626	4,631	56,667	87,922	8,509	95,083	66,042	28,561	98,823	37,904
Model 6	64,584	2,301	82,5	64,41	4,472	95,325	63,744	34,777	98,671	43,623

Table A.9: Results of the different NN architectures tested for the FK 720 dataset.

NN	Input strategy 1					Input strategy 2				
Model	Accuracy	Precision	Recall	Specificity	F1-score	Accuracy	Precision	Recall	Specificity	F1-score
Model 1	92,87	38,436	11,713	98,565	16,636	91,855	22,947	11,038	97,326	14,696
Model 2	98,486	20,903	12,5	99,366	14,09	91,985	31,427	24,212	96,547	26,934
Model 3	98,016	16,926	19,167	98,808	17,178	92,89	43,512	29,218	97,332	34,312
Model 4	96,732	11,762	30,833	97,399	16,669	92,676	32,593	29,622	96,622	29,416
Model 5	81,762	3,704	63,333	81,956	6,98	93,963	59,368	11,269	99,442	18,634
Model 6	50,783	1,86	80,833	50,46	3,629	93,996	52,239	25,742	98,401	33,679

A.2.2 Classification in the RK datasets

Table A.10: Results of the different NN architectures tested for the RK 180 dataset.

NN	Input strategy 1					Input strategy 2				
Model	Accuracy	Precision	Recall	Specificity	F1-score	Accuracy	Precision	Recall	Specificity	F1-score
Model 1	92,623	11,587	63,492	93,04	19,554	94,495	92,996	37,639	99,725	53,201
Model 2	91,596	10,507	64,921	91,975	18,035	96,491	87,087	66,153	99,113	74,655
Model 3	91,165	10,021	65,873	91,528	17,379	95,943	88,807	60,594	99,294	71,725
Model 4	90,727	9,457	64,762	91,095	16,463	96,504	88,97	63,798	99,316	73,707
Model 5	83,587	6,679	77,937	83,666	12,257	94,738	86,09	Ê44,413	99,342	58,113
Model 6	80,932	5,782	81,746	80,921	10,785	95,175	73,99	65,084	97,905	68,868

Table A.11: Results of the different NN architectures tested for the RK 360 dataset.

NN	Input strategy 1					Input strategy 2				
Model	Accuracy	Precision	Recall	Specificity	F1-score	Accuracy	Precision	Recall	Specificity	F1-score
Model 1	90,618	10,323	68,095	90,955	17,884	95,065	89,936	49,921	99,44	63,401
Model 2	91,527	10,626	62,381	91,963	18,13	95,842	87,709	63,992	99,078	73,435
Model 3	90,999	10,198	64,444	91,391	17,574	96,169	85,906	67,393	98,94	75,432
Model 4	90,851	10,067	64,603	91,24	17,376	96,066	82,646	69,685	98,588	75,461
Model 5	83,465	6,705	77,778	83,55	12,324	94,347	83,122	46,045	99,074	58,168
Model 6	76,671	5,255	80,952	76,609	9,832	95,738	78,235	71,827	98,054	74,693

Table A.12: Results of the different NN architectures tested for the RK 720 dataset.

NN	Input strategy 1					Input strategy 2				
Model	Accuracy	Precision	Recall	Specificity	F1-score	Accuracy	Precision	Recall	Specificity	F1-score
Model 1	90,761	10,768	64,603	91,174	18,391	93,587	86,882	44,067	99,233	57,884
Model 2	89,854	9,953	66,032	90,237	17,263	95,229	86,32	62,651	98,869	72,115
Model 3	988,803	9,444	69,524	89,115	16,615	95,249	83,58	65,376	98,576	73,208
Model 4	87,864	9,288	71,905	88,127	16,39	95,345	83,086	67,543	98,448	74,399
Model 5	77,454	5,595	81,111	77,393	10,449	94,014	78,304	53,172	98,377	62,459
Model 6	69,082	4,419	86,032	68,81	8,387	94,333	76,567	64,989	97,693	69,655

A.2.3 Feature selection in the FK datasets

Table A.13: Results of the feature selection techniques for the dataset FK 180 in the HO scenario.

Method	Hyperparameters	Accuracy	Precision	Recall	Specificity	F1-score
Without reduction	—	94,243	32,371	17,57	98,143	22,019
Pearson correlation	$n = 10$	94,916	41,064	25,63	98,341	30,779
	$n = 5$	95,014	42,594	23,88	98,517	30,023
	$n = 3$	95,58	47,728	29,585	35,614	35,614
LASSO regression	$\lambda = 0{,}001$	94,591	32,927	24,554	97,8	27,267
	$\lambda = 0{,}003$	94,412	37,386	14,016	98,728	20,023
	$\lambda = 0{,}005$	95,397	56,402	35,705	98,407	41,596

Table A.14: Results of the feature selection techniques for the dataset FK 360 in the HO scenario.

Method	Hyperparameters	Accuracy	Precision	Recall	Specificity	F1-score
Without reduction	—	95,325	63,744	34,777	98,671	43,623
Pearson correlation	$n = 10$	95,388	62,775	25,659	99,198	33,745
	$n = 5$	95,208	67,548	24,383	99,147	33,824
	$n = 3$	95,529	57,861	33,329	98,771	40,659
LASSO regression	$\lambda = 0{,}001$	95,035	62,942	29,706	98,692	38,546
	$\lambda = 0{,}003$	95,606	69,845	39,022	98,872	46,829
	$\lambda = 0{,}005$	95,458	70,103	40,007	98,844	49,33

Table A.15: Results of the feature selection techniques for the dataset FK 720 in the HO scenario.

Method	Hyperparameters	Accuracy	Precision	Recall	Specificity	F1-score
Without reduction	—	92,89	43,512	29,218	97,332	34,312
Pearson correlation	$n = 10$	93,589	40,035	32,483	97,22	34,674
	$n = 5$	93,581	44,795	41,926	96,747	43,031
	$n = 3$	93,521	47,809	38,674	97,184	41,592
LASSO regression	$\lambda = 0{,}001$	93,288	42,38	35,635	96,925	38,637
	$\lambda = 0{,}003$	93,715	43,137	38,642	97,036	39,909
	$\lambda = 0{,}005$	93,51	48,021	43,539	96,805	45,04

A.2.4 Feature selection in the RK datasets

Table A.16: Results of the feature selection techniques for the dataset RK 180 in the HO scenario.

Method	Hyperparameters	Accuracy	Precision	Recall	Specificity	F1-score
Without reduction	—	96,491	87,087	66,153	99,113	74,655
Pearson correlation	$n = 10$	95,767	89,316	54,565	99,408	67,432
	$n = 5$	95,066	88,306	47,739	99,395	61,127
	$n = 3$	95,117	88,631	47,101	99,412	59,932
LASSO regression	$\lambda = 0{,}001$	95,967	84,869	62,414	98,982	71,574
	$\lambda = 0{,}003$	95,354	87,709	51,042	99,334	63,327
	$\lambda = 0{,}005$	94,851	87,677	45,847	99,396	58,845

Table A.17: Results of the feature selection techniques for the dataset RK 360 in the HO scenario.

Method	Hyperparameters	Accuracy	Precision	Recall	Specificity	F1-score
Without reduction	—	96,066	82,646	69,685	98,588	75,461
Pearson correlation	$n = 10$	96,18	87,081	67,429	98,998	75,72
	$n = 5$	95,943	84,711	67,75	98,749	74,861
	$n = 3$	95,981	85,066	67,473	98,787	74,901
LASSO regression	$\lambda = 0{,}001$	96,216	85,222	69,96	98,786	76,517
	$\lambda = 0{,}003$	96,383	86,242	70,678	98,888	77,332
	$\lambda = 0{,}005$	96,15	82,269	71,558	98,514	75,806

Table A.18: Results of the feature selection techniques for the dataset RK 720 in the HO scenario.

Method	Hyperparameters	Accuracy	Precision	Recall	Specificity	F1-score
Without reduction	—	95,345	83,086	67,543	98,448	74,399
Pearson correlation	$n = 10$	95,538	86,509	67,088	98,799	75,112
	$n = 5$	95,868	87,675	68,516	98,908	76,565
	$n = 3$	95,303	85,869	65,196	98,763	73,807
LASSO regression	$\lambda = 0{,}001$	95,455	84,741	66,836	98,661	74,412
	$\lambda = 0{,}003$	95,862	86,492	70,011	98,748	77,105
	$\lambda = 0{,}005$	95,681	84,425	70,015	98,528	76,136